

FEDERAL UNIVERSITY OF TECHNOLOGY — PARANÁ

LENON DINIZ SEIXAS

VEHICLE INDUSTRY BIG DATA ANALYSIS USING CLUSTERING APPROACHES

PONTA GROSSA



2022

LENON DINIZ SEIXAS ✉ 

VEHICLE INDUSTRY BIG DATA ANALYSIS USING CLUSTERING APPROACHES

Análise de dados volumosos da indústria de veículos usando abordagem de agrupamento

Dissertation presented as a requirement for obtaining the title of Master in Electrical Engineering from the Federal University of Technology — Paraná (UTFPR).
Concentration Area: Control and instrumentation.

Advisor: Prof. Ph.D. Fernanda Cristina Corrêa ✉ 
Co-Advisor: Prof. Ph.D. Marcella Scoczynski Ribeiro Martins ✉ 

PONTA GROSSA

2022



4.0 International

This license allows reusers to distribute, remix, adapt, and build upon the material in any medium or format for noncommercial purposes only, and only so long as attribution is given to the creator. If you remix, adapt, or build upon the material, you must license the modified material under identical terms.



LENON DINIZ SEIXAS

ANÁLISE DE DADOS VOLUMOSOS DA INDÚSTRIA DE VEÍCULOS USANDO ABORDAGEM DE AGRUPAMENTO.

Trabalho de pesquisa de mestrado apresentado como requisito para obtenção do título de Mestre Em Engenharia Elétrica da Universidade Tecnológica Federal do Paraná (UTFPR). Área de concentração: Controle E Processamento De Energia.

Data de aprovação: 16 de Novembro de 2022

Dra. Fernanda Cristina Correa, Doutorado - Universidade Tecnológica Federal do Paraná

Dra. Marcella Scoczynski Ribeiro Martins, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Marcio Rodrigues Da Cunha Reis, Doutorado - Instituto Federal de Goiás

Myriam Regattieri De Biase Da Silva Delgado, - Universidade Tecnológica Federal do Paraná

Documento gerado pelo Sistema Acadêmico da UTFPR a partir dos dados da Ata de Defesa em 16/11/2022.

I dedicate this work to my wife, family
and friends, for the moments of
absence.

ACKNOWLEDGMENTS

This work could not have been completed without the help of several people and institutions, to which I am grateful. Of course, these paragraphs will not cover all the people who were part of this important phase of my life. Therefore, I apologize in advance to those who are not present between these words, but they can be sure that they are part of my thoughts and my gratitude.

To my family and girlfriend, for their love, encouragement and total support throughout my life.

To my supervisors, who showed me the paths to be followed and for their trust. To all the professors and colleagues of the department, who helped directly and indirectly in the realization and/or conclusion of this work.

To everyone else who somehow contributed to my personal and professional growth.

And finally, the Higher Education Personnel - CAPES (funding code 001), UTFPR support foundation - FUNTEF, Foundation Scholarship Programme Araucária and Volvo do Brasil Veículos Ltda for the financial support and opportunity.

“Até onde as leis da matemática se referem à realidade, elas não são certas; e até onde estão certas, não se referem à realidade.”
(EINSTEIN, 1921, tradução)^a.

^a “As far as the laws of mathematics refer to reality, they are not certain; and as far as they are certain, they do not refer to reality.” (EINSTEIN, 1921)

RESUMO

Trabalhar com dados se tornou algo fundamental e imprescindível no mundo moderno. Considerando uma economia e indústria mundial globalizada, a análise e visualização de dados oferece informações esclarecedoras para tomadas de decisão e planejamento estratégico. Para extrair o máximo valor possível de um conjunto de dados, a ciência de dados oferece diversos métodos estatísticos e científicos, abrangendo toda a preparação, limpeza, agregação e manipulação de dados. O Aprendizado de máquina (ML) e a Inteligência Artificial (AI) vêm juntos para aprender e explorar os dados, descobrindo coisas que não podem ser vistas apenas com a experiência do analista. O setor automotivo, que possui grande influência na economia e indústria mundial, sofre os impactos de ser focado em tecnologia e em prazos relativamente curtos. A transformação digital tem então um efeito disruptivo, considerado como o fenômeno mais importante nos 140 anos do setor, fazendo com que empresas de automóveis ofereçam produtos personalizados e otimizados para as necessidades do cliente. Para fazer isso, é necessário trabalhar extensivamente com dados e ciência de dados. Então, este trabalho traz um estudo para investigar os métodos de *clustering* de um conjunto de dados *Big Data* de uma companhia de automóveis, realizando uma revisão da literatura, tratando, normalizando e agrupando usando os métodos pesquisados, e, por fim, comparando e analisando os resultados. Foi utilizado o método *Knowledge Discovery and Data mining* para realizar o processo de mineração, comparando o desempenho dos algoritmos *K-Means*, *Fuzzy C-Means* (FCM) e Mapas Auto-Organizáveis (SOM) por meio de algumas métricas: soma dos quadrados dentro de clusters (SSW), soma dos quadrados entre *clusters* (SSB), índice silhueta (SI) e validação cruzada *K-Fold* com pontuação de homogeneidade. Para o parâmetro de inclinação os algoritmos de ML trouxeram uma melhor resposta em geral quando comparado ao método de classificação por regras chamado GTA, que não é um algoritmo de aprendizado de máquina, quando analisando as métricas apresentadas. Dentre os algoritmos de ML implementados, *K-Means* e *Fuzzy C-Means*, *K-Means* é ligeiramente superior para as métricas SSW e SSB, porém o *Fuzzy C-Means* é melhor nas métricas SI e validação cruzada. Quando é analisado o conjunto dos resultados obtidos, os algoritmos de ML tendem a distribuir mais igualmente a população entre os *clusters* do que a classificação sem aprendizado e a métrica SI comprova isso como uma boa decisão. Os métodos trazidos para este trabalho apresentaram resultados satisfatórios sobre o conjunto de dados, e mostram como a aplicação de ML pode trazer benefícios à mineração de dados. Com isso, conseguimos responder à pergunta "Como os dados históricos de uso podem ajudar uma fabricante de caminhões a melhorar o desenvolvimento de produtos e o consumo de combustível?". O *K-Means* é uma boa opção para técnica de agrupamento, enquanto o FCM também se mostrou uma boa técnica, trabalhando bem principalmente com situações de sobreposição. O FCM traz também uma interpretação extra da porcentagem de associação do cluster que pode ajudar os usuários finais a entender ainda mais os dados. Para trabalhos futuros, também podem ser implementados *K-Medoids* como método alternativo que considera um indivíduo como o centro do cluster. Este trabalho pode ser estendido para outros tipos de conjuntos de dados.

Palavras-chave: big data; machine learning; indústria 4.0; aprendizado não supervisionado; agrupamento; veículos.

ABSTRACT

Working with data has become something fundamental and essential in the modern world. Considering a globalized world economy and industry, data analysis and visualization offer enlightening information for decision making and strategic planning. Data science provides diverse statistical and scientific methods to extract the most value possible from a data set, covering all the preparation, cleaning, aggregation, and manipulation of data. Machine Learning (ML) and Artificial Intelligence (AI) come along with it to learn and explore the data, uncovering things that can not be seen with only the analyst experience. The automotive sector, which significantly influences the world economy and industry, suffers from focusing on technology and short-term focus. Digital transformation has a disruptive effect, considered essential in the sector's 140 years, causing car companies to offer customized products optimized for customer needs. To work extensively with data and data science becomes fundamental. So, this work brings a study to explore clustering methods in a Big Data dataset of a car company, performing a literature review; treating, normalizing and grouping using the researched methods; and, finally, comparing and analyzing the results. The Knowledge Discovery and Data mining method was used to perform the mining process, comparing the performance of the K-Means, Fuzzy C-Means (FCM) and Self-Organizing Maps (SOM) algorithm through some metrics: sum of squares within clusters (SSW), sum of squares between clusters (SSB), silhouette index (SI) and K-Fold cross-validation with homogeneity score. When evaluating the vehicle's distribution of the results obtained, the ML algorithms tend to distribute more evenly among the clusters than the classification without learning, and the SI metric proves this as a good decision. The methods brought to this work showed satisfactory results on the dataset, and demonstrate how the application of ML can bring benefits to data mining. With this, we managed to answer the question "How can historical usage data help a truck manufacturer improve product development and fuel consumption?". K-Means is a good and main clustering technique, while FCM has also proved to be a good technique, working mainly with overlapping situations. FCM also brings an extra interpretation of cluster membership percentage that can help end users understand the data even more. For future works, it can be also implemented K-Medoids as an alternative method that considers an individual as the center of the cluster. This work can also be extended to other types of vehicle's data set.

Keywords: big data; machine learning; industry 4.0; unsupervised learning; clustering; vehicles.

LIST OF ALGORITHMS

| | |
|--|-----------|
| Algorithm 1 – K-Means | 35 |
| Algorithm 2 – Fuzzy C-Means Algorithm | 37 |
| Algorithm 3 – Self-Organizing Map Algorithm | 40 |

LIST OF ILLUSTRATIONS

| | |
|---|-----|
| Figure 1 – Graphical illustration of high and low bias and variance | 29 |
| Figure 2 – Basic structure of SOM neural network | 39 |
| Figure 3 – Illustration of KDD process | 49 |
| Figure 4 – The whole data path | 50 |
| Figure 5 – Example of a vehicle’s slope vector | 53 |
| Figure 6 – Pearson correlation of slope data set | 55 |
| Figure 7 – Pearson correlation of slope data set zoom | 56 |
| Figure 8 – Example of a vehicle’s slope vector post dimensionality reduction | 57 |
| Figure 9 – Example of a vehicle’s speed vector | 58 |
| Figure 10 – Example of a vehicle’s GCW vector | 60 |
| Figure 11 – SI Slope Elbow Curve | 66 |
| Figure 12 – k-Fold Homogeneity Slope Elbow Curve | 67 |
| Figure 13 – Slope Detailed metrics comparison | 69 |
| Figure 14 – Speed detailed metrics comparison | 83 |
| Figure 15 – GCW detailed metrics comparison | 96 |
| Graph 1 – Silhouette comparison between original 32 dimension and trans- formed 16 dimension data sets | 62 |
| Graph 2 – SSW Slope Elbow Curve | 63 |
| Graph 3 – SSB Slope Elbow Curve | 65 |
| Graph 4 – GTA clusters average | 71 |
| Graph 5 – Slope clusters average from FCM with 4 clusters | 72 |
| Graph 6 – Example of membership grades of a vehicle in Slope data set . | 73 |
| Graph 7 – Clusters average from K-Means with 5 clusters | 73 |
| Graph 8 – Slope clusters average from SOM with 5 clusters | 74 |
| Graph 9 – Slope clusters data set distribution | 76 |
| Graph 10 – SSW Speed Elbow Curve | 78 |
| Graph 11 – SSB Speed Elbow Curve | 80 |
| Graph 12 – SI Speed Elbow Curve | 81 |
| Graph 13 – k-Fold Homogeneity Speed Elbow Curve | 82 |
| Graph 14 – Speed clusters centroids for FCM with 8 clusters | 85 |
| Graph 15 – Example of membership grades of a vehicle in Speed data set . | 86 |
| Graph 16 – Speed clusters average from K-Means with 7 clusters | 86 |
| Graph 17 – Speed clusters average from SOM with 7 clusters | 87 |
| Graph 18 – Speed clusters data set distribution | 90 |
| Graph 19 – SSW GCW Elbow Curve | 91 |
| Graph 20 – SSB GCW Elbow Curve | 93 |
| Graph 21 – SI GCW Elbow Curve | 94 |
| Graph 22 – k-Fold Homogeneity GCW Elbow Curve | 94 |
| Graph 23 – GCW clusters centroids for FCM with 6 clusters | 98 |
| Graph 24 – Example of membership grades of a vehicle in GCW data set . | 98 |
| Graph 25 – GCW clusters average from K-Means with 6 clusters | 99 |
| Graph 26 – GCW clusters average from SOM with 6 clusters | 100 |
| Graph 27 – GCW clusters data set distribution | 102 |
| Graph 28 – Example of classification | 103 |

LIST OF TABLES

| | |
|---|-----------|
| Table 1 – Slope dimensions | 52 |
| Table 2 – Speed dimensions | 57 |
| Table 3 – GCW dimensions | 59 |
| Table 4 – Slope Coefficient Variation table | 69 |
| Table 5 – Average silhouette of Slope clusters | 70 |
| Table 6 – Speed Coefficient Variation table | 83 |
| Table 7 – Average silhouette of Speed clusters | 84 |
| Table 8 – GCW Coefficient Variation table | 96 |
| Table 9 – Average silhouette of GCW clusters | 97 |

LIST OF ABBREVIATIONS AND INITIALS

| | |
|--------|---|
| AI | Artificial Intelligence |
| AIC | Akaike Information Criterion |
| ANN | Artificial Neural Networks |
| BIC | Bayesian Information Criterion |
| CH | Calinski and Harabasz Index |
| CPU | Central Processing Unit |
| CV | Coefficient of Variation |
| DB | Davies-Bouldin Index |
| DBSCAN | Density-based Spatial Clustering of Applications with Noise |
| DI | Dunn's index |
| DL | Deep Learning |
| DNg | Generalized Dunn's Index |
| DNs | Modified Dunn's index |
| FCM | Fuzzy C-Means |
| FS | Fuzzy Silhouette |
| GCW | Gross Combination Weight |
| HS | Homogeneity Score |
| IBM | International Business Machines Corporation |
| IEEE | Institute of Electrical and Electronics Engineers |
| IoT | Internet of Things |
| KDD | Knowledge Discovery and Data mining |
| ML | Machine Learning |
| NIST | National Institute of Standards and Technology |
| PBM | Pakhira, Bandyopadhyay and Maulik |
| PCC | Pearson correlation coefficient |
| SI | Silhouette Index |
| SOM | Self-Organizing Maps |
| SSB | Sum of Squares Between clusters |
| SSW | Sum of Squares Within clusters |
| SW | Silhouette Width |
| UTFPR | Universidade Tecnológica Federal do Paraná |

SUMMARY

| | | |
|----------------|---------------------------------------|-----------|
| 1 | INTRODUCTION | 14 |
| 1.1 | Automotive Industry | 17 |
| 1.2 | Justification | 19 |
| 1.3 | Objectives | 20 |
| 1.3.1 | General Objectives | 20 |
| 1.3.2 | Specific Objectives | 20 |
| 1.4 | Work Organization | 21 |
| 2 | LITERATURE REVIEW | 22 |
| 2.1 | Big Data | 22 |
| 2.2 | Machine Learning | 25 |
| 2.3 | Automotive applications | 28 |
| 2.4 | Clustering Algorithms | 32 |
| 2.4.1 | K-Means | 34 |
| 2.4.2 | Fuzzy C-Means | 36 |
| 2.4.3 | Self-Organizing Map | 38 |
| 2.5 | Evaluation Metrics | 40 |
| 2.5.1 | Sum of Squares Within Clusters (SSW) | 42 |
| 2.5.2 | Sum of Squares Between Clusters (SSB) | 43 |
| 2.5.3 | Silhouette Index | 43 |
| 2.5.4 | k-Fold Cross-Validation | 45 |
| <u>2.5.4.1</u> | <u>Homogeneity Score (HS)</u> | <u>46</u> |
| 2.5.5 | Coefficient of Variation | 47 |
| 3 | MATERIAL AND METHODS | 48 |
| 3.1 | Data preprocessing | 49 |
| 3.2 | The addressed datasets | 51 |
| 3.2.1 | Slope | 51 |
| <u>3.2.1.1</u> | <u>Correlation analysis</u> | <u>53</u> |
| 3.2.2 | Speed | 55 |
| 3.2.3 | Gross Combination Weight (GCW) | 58 |
| 4 | RESULTS AND ANALYSIS | 61 |
| 4.1 | Slope | 61 |
| 4.1.1 | Performance evaluation | 62 |
| <u>4.1.1.1</u> | <u>SSW</u> | <u>62</u> |
| <u>4.1.1.2</u> | <u>SSB</u> | <u>64</u> |
| <u>4.1.1.3</u> | <u>Silhouette</u> | <u>64</u> |
| <u>4.1.1.4</u> | <u>Homogeneity</u> | <u>67</u> |
| <u>4.1.1.5</u> | <u>Coefficient of variation</u> | <u>68</u> |
| 4.1.2 | Summary of results | 68 |
| 4.1.3 | Centroids of the clusters | 70 |
| 4.1.4 | Vehicle distribution in clusters | 75 |
| 4.2 | Speed | 77 |
| 4.2.1 | Performance evaluation | 77 |
| <u>4.2.1.1</u> | <u>SSW</u> | <u>77</u> |
| <u>4.2.1.2</u> | <u>SSB</u> | <u>79</u> |

| | | |
|------------|---|------------|
| 4.2.1.3 | <u>Silhouette</u> | 79 |
| 4.2.1.4 | <u>Homogeneity</u> | 81 |
| 4.2.1.5 | <u>Coefficient of variation</u> | 81 |
| 4.2.2 | Summary of results | 83 |
| 4.2.3 | Centroids of the clusters | 84 |
| 4.2.4 | Vehicle distribution in clusters | 88 |
| 4.3 | Gross Combination Weight (GCW) | 89 |
| 4.3.1 | Performance evaluation | 89 |
| 4.3.1.1 | <u>SSW</u> | 89 |
| 4.3.1.2 | <u>SSB</u> | 92 |
| 4.3.1.3 | <u>Silhouette</u> | 92 |
| 4.3.1.4 | <u>Homogeneity</u> | 94 |
| 4.3.1.5 | <u>Coefficient of variation</u> | 95 |
| 4.3.2 | Summary of results | 95 |
| 4.3.3 | Centroids of the clusters | 97 |
| 4.3.4 | Vehicle distribution in clusters | 100 |
| 4.4 | Example of classification | 101 |
| 5 | CONCLUSION | 105 |
| | REFERENCES | 110 |

1 INTRODUCTION

The analysis and visualization of data, in general, become a fundamental and differential component in the modern world concerning the new globalized models of the world economy and industry, offering comprehensive information for decision-making and strategic planning in the most diverse sectors (APERGIS; FILIPPIDIS; ECONOMIDOU, 2007; WANG, L. *et al.*, 2021; KATZ; BIEM, 2021; OHKUMA *et al.*, 2018; MA *et al.*, 2020).

However, the historical roots of graphic analysis and visual representations, which are intertwined with statistical and cartographic analysis, are seen in fields of science until the 19th century, perhaps connecting with statistical thinking and data obtained for planning and commerce, extending to the present day with technological progress in data collection, data storage, image processing, and advances in mathematics and statistics (FRIENDLY, 2008). Friendly (2008) further divides the history of data visualization into milestones that would illustrate the periods and difficulties encountered: Initial maps and diagrams (the period before the 17th century); Theory and measurement (17th century); New graphic models (18th century); Early modern period (first half of 19th century); Golden Age (second half of 19th century); Modern Dark Ages (first half of 20th century); Rebirth of Data Visualization (second half to the third quarter of 20th century); High-definition, dynamic and interactive data visualization (third quarter of 20th century to the present). The development of technologies provides current evolution for capturing, storing, and processing data that culminate today in Big Data & Analytics.

Precisely, big data is data that has significant Volume, Velocity, and Variety, called "The Three V's of Big Data", which after was extended to five V's, so including Veracity and Value. These are larger and more complex data sets that traditional data processing software cannot manage, but they provide information to solve problems that were impossible to solve before. In a big data set, it will be necessary to process volumes of low-density unstructured data, which may still have unknown values, that use tens or even hundreds of terabytes or petabytes. Of these data, those with the highest acquisition speed are usually transmitted to memory instead of being written to disk. Some products operate in real (or near) time, requiring evaluation and action at the same speed. In addition, handling varied data is necessary, which usually does

not fit perfectly in a relational database. Unstructured and semi-structured data, such as text, audio, and video, require additional pre-processing to add value and support metadata. In other words, big data can provide more complete and faster answers, with more data reliability (BRASIL, 2021a).

Working with big data is transforming businesses, serving as a cross-functional capacity for aligning strategies and making decisions according to demands in a space that companies invest heavily in seeking opportunities to apply data analytics and overcome competition (JOHNSON; FRIEND; LEE, 2017). The growing interest in the subject increasingly makes data one of the most valuable organizational resources. Several studies empirically demonstrate the value that big data and business analytics bring to organizational agility, their impact on innovation and product development, and how it benefits competitive performance (CÔRTE-REAL; RUIVO; OLIVEIRA, 2020; LEHRER *et al.*, 2018; ASHRAFI *et al.*, 2019). Mikalef *et al.* (2020) in their study, they assess the increased interest of the scientific community in the subject in the last decade, denoting an exponential growth of at least 1400% from 2010 to 2018 in the number of annual publications. Gandomi and Haider (2015) remark as well the frequency distribution of documents containing the term "big data" rising from 2010 on.

Nonetheless, it is necessary to develop the organizational capacity to identify areas within the business where it is possible to derive the proper value from big data. Finding where to benefit from the insights and viewings that data itself brings is a great challenge to overcome, so then being able to strategically plan and execute data analytics projects, combining and pooling resources needed to turn data into transformative action (GUPTA; GEORGE, 2016; VIDGEN; SHAW; GRANT, 2017). Wessel (2016) points out that big data allows companies to easily adapt to new environments and more quickly bring disruptive innovations – a term that describes standards-breaking technologies in general – through three principles: low cost, affordability, and a structured business model.

Datasets and database developments have their origins around the 1960s and 1970s, with the first data centers. As of 2005, the amount of data from users of online services and social networks has grown remarkably, also popularizing code structures to store and analyze large and differentiated data sets, such as NoSQL – non-relational data structures – and the platform Hadoop, which is capable of processing high workloads, being highly scalable and flawless (HADOOP, 2021). Such structures become essential

for the growth of big data, making data storage more manageable and cheaper.

Around this, in Hannover Messe 2011, an event by the German government to talk about the industry, the term Industry 4.0 appeared, coining the vision of a new industrial revolution that uses concepts of Big Data & Analytics, Cloud computing, Internet of Things (IoT) and Cyber-physical Systems, implying in definitions such as additive and hybrid manufacturing, virtual simulation, intelligent robots, horizontal and vertical systems' integration, cybersecurity and augmented reality (PFEIFFER, 2017; RIBEIRO; ABREU, 2020).

The new era of the industry was triggered by general social, economic, and political changes, in particular the need to: shorten development and innovation periods, as the high innovation capability becomes more essential; individualize demand and products, as buyers can define now conditions on the trade; flexible product development, due to the new framework requirements; decentralize, fastening decision-making to cope with the specified conditions needed; manage resource efficiency on economic and ecological aspects, maintaining an intense focus on sustainability in industrial contexts. Not only that, but there are spread identified comprehensive approaches in a technology-push that influence industrial environments and daily routine. Further mechanization and automation of processes, digitalization and networking on manufacturing and manufacturing-supporting, and miniaturization of components are some of the developments that have the potential to turn around industrial practices (LASI *et al.*, 2014) comprehensively. Overall, this brings on an industrial revolution that relies more and more on data consistency and its value.

To claim as much value as possible from the data, the area of data science sees several spotlights opening up for diverse statistical and scientific methods to shine. Data science covers all the preparation, cleaning, aggregation, and manipulation of data for further analysis, and its procedures can be divided into Extraction of data, Manipulation, Visualization, and Maintenance (BRASIL, 2021b). However, most standard tasks from data scientists can be classified as clustering (or segmentation), anomaly (or outlier) detection, association rule mining, and predictions (including classifications and regressions) (KELLEHER; TIERNEY, 2018).

To extract useful value from data sets, algorithms based on Machine Learning (ML) and Artificial Intelligence (AI) are the most common ones. Machine learning and artificial intelligence algorithms have open fields to explore the data, learn with it, and

aid in understanding and seeing things that can not be uncovered with only the analyst's experience over the matter.

1.1 Automotive Industry

Vehicles companies have a significant impact on industry and economics. As once transformed world economy with institutionalized economic models, such as Fordism and Toyotism, today impacts development of new business models, social and environmental changes (HILALI, 2015; JANOSKI; LEPADATU, 2013; NIEUWENHUIS; WELLS, 2015).

In the past thirty years, the automotive industry faced enormous challenges. The volume production became more unprofitable, the segmented niche markets increased, and the sustainability of the products and methods of production suffered from regulatory and social pressure. However, still, it is one of the world's largest manufacturing sectors (NIEUWENHUIS; WELLS, 2003, 2015).

The automotive industry is a sector that remarkably suffers from being technology-focused and relatively short-term focused (NIEUWENHUIS; WELLS, 2015). Its product life-cycle faces new challenges arising from digital transformation, software-driven innovation, new product functionality, supply chains, and partnering. These challenges are all linked to the evolution from mechatronic products to innovative connected products and cyber-physical systems (DENGER; ZAMAZAL, 2020).

The digital transformation generates significant benefits for entrepreneurs, consumers, and society. No different in the automotive industry, it results in a disruptive effect that authors consider the most important phenomenon in the 140-years of industry history (LLOPIS-ALBERT; RUBIO; VALERO, 2021).

Digital transformation strategies reflect the changes brought by new technologies in an organization, seen in companies having to transform traditional and robust business models to adapt to trends like car-sharing platforms, telematics services, autonomous driving, mobility as service, etc. (RIASANOW; GALIC; BÖHM, 2017; CHANIAS; HESS, 2016; KOTARBA, 2018).

It has been a constant in the recent three decades the automotive industry trying to offer personalized products and various vehicle models efficiently, that can share diverse characteristics yet differ from others, but overall in an optimized way for the

clients' needs (BERSCH; AKKERMAN; KOLISCH, 2021). To do so, they have been working and improving several data science methods in fields like predictive maintenance, new product development, industrial processes, and many others (MALACA *et al.*, 2019; WANG, X. *et al.*, 2020; THEISSLER *et al.*, 2021).

The digitization of the vehicle industry will enable the evolution of business-to-business approaches to a business-to-consumer model, with new ways of engaging customers and partnerships with suppliers, interacting through data technologies. That is, the shift from selling a product to now offering value with a focus on the customer experience (HOFFMANN; ZAYER; STREMPER, 2019). Mining, collecting and using historical data information is key to making this happen.

To offer customized product is necessary to understand which type of behavior matches the client's needs. Given the environment and conditions of usage, understanding the realistic behavior of vehicles is the most important stage of this process. To do that, it is needed first to collect vehicles historical usage data long enough, and then analyse this Big Data data set through data science methods to extract the most value as possible from it. For this work, we have used clustering algorithms to help understand behaviors of vehicles in these data sets. By having a complete and clear information, companies can sell the product specification which has historically been proven to be the best one for the customer need.

Classifying vehicles into clusters allows for better comparisons, and even more when different types of data sets are used to do this. Crossing information of slope (the environment), speed (the response of the vehicle in the environment) and carried weight (vehicle's type of usage), when it comes to trucks, it is possible to have full knowledge for the comparison. This can be used, for example, to track how drivers perform under the same conditions and which type of driving saves more fuel or requires more maintenance. According to Hao, Yang, and Zhou (2019), different driving behaviors can cause 7% to 25% of difference in fuel consumption, once the influence of the driving behavior on that is up to 30%. Through this historical usage and being able to differentiate behaviors, companies can return customers information about their vehicle's, offering services and improvements to their products, which is aggregating value with focus on the experience.

1.2 Justification

Applying scientific research to industry or business, in general, is not new at all. Taylorism by itself transformed management models in the 20th century by proposing that managers should become scientific, study the organization of work and extract operational efficiency (WARING, 2016). On the same page, nowadays, design research serves as the spinal dorsal that makes a new paradigm for companies to succeed (HILALI, 2015). Design research is an interdisciplinary area with social science orientation aspects that can be divided into research for design, research into design, and design as research. Briefly, it takes place where one wants to change something or create something new, whatever it can be, applying science and research-based knowledge on how to obtain extraordinary and innovative results (DOWNTON, 2003; BÆRENHOLDT, 2010). Researching is the first step of innovation, and innovation is what makes the success of a business.

According to Industria (2018), digital technologies in vehicles represent at least 50% of the total vehicle value. Driver connectivity, location-based services, and the type of driver based on tastes and preferences have contributed to accelerating the process of digitalization of the automotive sector, which, allied to the integration of software and hardware, increased cars' functionalities and complexity (LLOPIS-ALBERT; RUBIO; VALERO, 2021). That aligns with the interests of cars companies in big data and data science.

With developments in maintenance modeling fuelled by data-based approaches, many opportunities are enabled. Predictive maintenance, for instance, is a crucial approach to secure functional safety over the product life cycle while still limiting maintenance costs, and ML is an ideal candidate for it since modern vehicles have a massive amount of operating data. That is a field that has seen growing interest in the academic scenario, and more publicly available data would boost the research activities in the area (THESSLER *et al.*, 2021).

Many technologies track vehicle performance and usage. This generates large data sets with gold information to the automobile industry growth. Insights that could not be seen before Industry 4.0 now can be exploited and studied for product and features development. However, each parameter extracted from these big data sets should be carefully studied and treated to draw every essential information, removing any rubbish

that may come along.

This work brought a study over three vehicle parameters data sets. With these data sets we sought to answer the question: How can historical usage data help a truck manufacturer improve product development and fuel consumption? The goal is to find the best clustering strategy for each one to extract the most value and understanding possible from it, helping end-users comprehend trucks' behavior perfectly. The first parameter refers to slope conditions tracked on the roadway, which tells about the vehicle's inclinations; the second parameter is the measurement of speed throughout the entire driving life; the third parameter is the weight carried by the vehicle. The data sets used throughout the work comprehend several years of extensive trucks usage in the Latin America environment, provided for personal use by a major car company following General Data Protection Law (GDPL).

Classification methods based on data engineers' experience were compared with the most up-to-date methods of clustering, facing the data set with different approaches, which helped give different sights of the same set.

The objective for each parameter is to group the data set in an optimized number of clusters that can explain vehicles' behaviors and differentiate them. Having different clusters, the company can see how clients' fleets are performing, so being able to offer better vehicle specifications for each use. Also, knowing how the product is performing along the life cycle gives extra knowledge for product development and fuel consumption reduction.

1.3 Objectives

1.3.1 General Objectives

To investigate the clustering methods of a car company's Big Data data set, understand each application and market behavior, and extract maximum value from it.

1.3.2 Specific Objectives

- To review the literature on the subject;
- To model the data for the clustering;

- Find the best clustering strategy for the problem.

1.4 Work Organization

In the first chapter, an introduction was made, presenting a global view of the research, including a brief history, importance and justification for the choice of the theme, delimitation of the subject, formulation of hypotheses and research objectives, and structure of the work.

The second chapter brings the literature review, committing to the explanations and definitions around the subjects presented in the work, resuming the importance of the content and its impacts. Also, the chapter presents an overview of state-of-art techniques and concepts around the topic, exemplifying general applications and repercussions. This chapter is divided into three sections: Big data, Machine Learning, and Automotive applications.

Chapter three presents the implemented models, methodologies employed, error analysis, and comments. The chapter is divided into Data preprocessing, clustering algorithms, Evaluation metrics, and Data sets.

The fourth chapter brings the results and analysis for each data set we worked with, these being the title of each section division: Slope, Speed, Gross Combination Weight (GCW). Performance evaluation with four metrics compares the proposed ML methods with a non-ML method. Also, clusters of centroids and population distribution into clusters are analysed.

In the last chapter, chapter five, the work is summarized, presenting the conclusions and insights obtained with the method developed and the research made. Future works are also presented, and an expectation for the subject's future.

2 LITERATURE REVIEW

This chapter will review the literature, exploring each theme brought on by this work in depth. Concepts and definitions around the subjects are presented in this section, alongside the context of where each one is involved and required. Furthermore, an overview of the applications of the matter is shown, resuming the importance and deliberate approaches found in scientific literature.

The references chosen to support this work were selected after a review of the most popular and renowned academic research databases, such as IEEE Xplore, ScienceDirect, Web of Science, etc.; and worldwide universities research libraries, sometimes with the ease of Google Scholar search engine. Priority was given to publications with the most relevance, impact, or citations over the theme, at first, with no date filtering. Then, the most impactful works in the last three years have been selected, often searching for different approaches or angles of view among themselves. On the other hand, some references were selected individually by either being from a significant company or source of information about the subject, or by presenting clearer and more profound aspects of information.

2.1 Big Data

Big Data and Analytics is a relatively new term, not yet with a uniform definition, enabled by recent advances in technologies that support high-velocity data capture, storage, and analysis. At first, before its trend and even before being labelled, Laney (2001) proposed a three-fold definition encompassing the three Vs.: Volume, Velocity, Variety; remarking upon the increasing size of data, the rate that it is produced, and the increasing range of formats or representations it has. Despite the definition being entirely anecdotal, Beyer and Laney (2012) expanded alongside IBM's and NIST's work to include Veracity as the fourth V, comprising trust and uncertainty regarding data and the outcome of its analysis (WARD; BARKER, 2013; GANDOMI; HAIDER, 2015). The National Institute of Standards and Technology, when defining Big Data, also highlights the need for a "scalable architecture for efficient storage, manipulation, and analysis" (NIST, 2015).

De Mauro, Greco, and Grimaldi (2016) notice in the review that definitions of

Big Data describe their focus according to four themes: Attributes of data, just as the "3 Vs." definition; Technological needs, such as the Microsoft definition that it describes processes in which serious computing power is applied to massive and often highly complex sets of information; Thresholds, which are definitions that classify as Big Data that exceed the processing of conventional databases and alternative approaches are needed to do so; and Social impact, that highlights the effect of Big Data on the society, being so cultural, technological and scholar phenomenon. However, still, the authors propose a definition referring to the nature of the information asset itself: "Big Data is the Information asset characterized by such High Volume, Velocity, and Variety to require specific Technology and Analytical Methods for its transformation into value" (DE MAURO; GRECO; GRIMALDI, 2016).

The NIST (2015) document emphasizes the need for a system architecture that can scale and achieve the needed performance and cost-efficiency, denoting vertical scaling and horizontal scaling methods for it. The first implies increasing the system parameters of processing speed, memory, and storage for more extraordinary performance. However, this approach is limited by physical capabilities, as this requires ever more sophisticated hardware and software, whose progress has been tracked and described by the Moore's Law (MOORE, 1965). The other method, horizontal scaling, uses distributed individual resources integrated to behave as a single system, thus laying down the Big Data paradigm.

The Big Data paradigm distributes data systems across horizontally coupled, independent resources to achieve the scalability needed for the efficient processing of extensive datasets.

This new paradigm leads to several conceptual definitions that suggest Big Data exists when the scale of the data causes the management of the data to be a significant driver in the design of the system architecture. (NIST, 2015).

Not explicitly referring to the horizontal scaling, but this Big Data paradigm fundamentally states a shift in data system architectures from monolithic systems with vertical scaling into a parallelized system, horizontally scaled in a specific manner, that uses a loosely coupled set of resources in parallel.

Supercomputers built with large arrays of off-the-shelf CPUs appeared at first in the late 1990s, trending what became massively parallel processing, which is the multitude of individual processors working in parallel to execute a program (NIST, 2015). Parallel computation can speed up the execution, but one must be aware that increasing the number of processors decreases its efficiency, which is called the cost of parallel

computation (ROOSTA, 2012). As well, the program structure and memory organization of a multi-computer system must fulfill a set of conditions that describe whether two successive portions of a given program can perform in parallel and produce the same results (BERNSTEIN, 1966).

Another disruptive innovation from the 21st century, and a tool very needed in this case, is that Big Data invariably resorts to non-relational structure models, referred to as NoSQL, said logical data models which do not follow relational algebra for storage and data manipulation. Relational models are beneficial in reliability, flexibility, robustness, and scalability. However, to answer modern applications' needs, where that data is enormous and primarily unstructured, there is the non-relational databases' genuine usability (JATANA *et al.*, 2012).

In a high-level overview, a typical data architecture is composed of data sources, where the data is generated; data storage, where they are stored and processed; and applications, where the data is shared (KELLEHER; TIERNEY, 2018).

Among different tools, methods, and problems, big data engineers should know how to extract the due power of each concept and apply it correctly to their problems. By definition, according to NIST (2015), big data engineering is to include advanced techniques that use and harness independent resources to build scalable data systems whenever new architecture is required for efficient storage, manipulation, and analysis. That, potentially related to new technologies, challenges, and innovation problems, is what makes Big Data one of the pillars of the Industry 4.0 (PILLONI, 2018).

Industry 4.0 extends itself even to health domains, where IoT, Cloud Computing, and Big Data technologies revolutionize health and its ecosystem towards what is to be called Healthcare 4.0. Big Data has immense importance on the matter, followed by the increase of data that comes from the advanced technologies, just as the stream processing systems monitor people's health status in real-time that generates large amounts of structured and unstructured streamed data, thanks as well to the personal devices and wireless sensors progress. Moreover, big data techniques open up new technologies to be exploited on medical tests, images, and descriptions from clinicians, collected through records that may assume several forms and domination, such as Electronic Health Records and Electronic Medical Records. Overall, that progress in data technologies provides insights to reduce inefficiency in clinical operations, public health, research, and development, yet offering massive potential for prognostic interventions,

novel therapies, shaping lifestyle and behavior, and improving cost efficiencies and sustainability on infrastructure (ACETO; PERSICO; PESCAPÉ, 2020; KAMBATLA *et al.*, 2014).

Big Data development has changed the method of decision-making from a static process into one that is dynamic; once indeed, the analysis of the relationships between many events derived from information data has been replacing the pursuit of traditional and logical connections (DE MAURO; GRECO; GRIMALDI, 2016).

Özemre and Kabadurmus (2020) use Big Data Analytics methodology, employing machine learning algorithms, to forecast export volumes and conduct strategic market analysis on international trades. The method facilitates the strategic decision-making process, providing insights into global markets in a highly competitive business environment.

Big Data, new computing methods and data sources have changed economics and made it a more applied field. Areas like labor and public economics shifted their focus from theory toward estimating quantities (CURRIE; KLEVEN; ZWIERS, 2020). This leads to a more precise planning and action from the administration sector.

Large firms have produced more data, which can disproportionately benefit them, principally financially speaking. Data analysis improves investors' forecasting ability and reduces equity uncertainty, which lowers a company's cost of capital. That is, when investors can process more data, the investment costs of large companies fall further, allowing large companies to grow even more (BEGENAU; FARBOODI; VELDKAMP, 2018).

2.2 Machine Learning

Data-intensive science, shortened to data science, consists of three primary activities: capture, processing, and analysis. It refers to the data analysis guidance as an empirical science, learning from the data itself. Hey, Tansley, and Tolle (2009) proclaim data science in its purest form as the fourth paradigm of science, followed by experiment, theory, and computational sciences. Data science, then, can be defined as the extraction of litigable knowledge directly from the data through a research process or hypothesis formulation and hypothesis testing, performing the scientific process directly on the data (NIST, 2015).

Machine Learning (ML) is a field of computer science research and focuses on developing algorithms to match extract valuable patterns from an input data set (KELLEHER; TIERNEY, 2018). It is designed to emulate human intelligence by learning from the surrounding environment, with an ability to learn from the current context and generalize into unseen tasks. It is considered the working horse of the big data era (EL NAQA; MURPHY, 2015).

According to Bonaccorso (2017), the main goal of machine learning is to study, engineer, and improve mathematical models that can be trained with context-related data to make decisions and infer the future without the complete knowledge of all influencing elements. Using a statistical learning approach to determine the suitable probability distributions to choose the most likely successful action.

ML algorithms have been applied successfully in fields ranging from pattern recognition, computer vision, spacecraft engineering, finance, entertainment, computation biology, biomedical, medical applications, and many others (EL NAQA; MURPHY, 2015).

ML can be divided into supervised learning, unsupervised learning, and reinforcement learning. Also, deep learning can be considered a branch of machine learning.

The supervised learning follows the concept of having guidance, a teacher or supervisor, whose objective is providing the agent a precise measure of its error. This can be obtained by a training set of couples: input and expected output. After each iteration – considering that the algorithm is flexible enough and the data elements are coherent – the overall accuracy increases, lowering the difference between the predicted and expected value to zero. However, the system needs to be trained to work also with samples never seen before, being able to generalize yet avoiding overfitting. Supervised learning applications can include a predictive analysis based on regression or categorical classification, spam detection, pattern detection, natural language processing, sentiment analysis, automatic image classification, automatic sequence processing (BONACCORSO, 2017).

On the counterpart, the unsupervised approach is based on the absence of a supervisor (absolute error measures). This approach is practical when it is necessary to learn how a set of elements can be grouped or clustered according to their similarity. That clustering should consider the presence of outliers, treat them to increase the

internal coherence (density) within the cluster, and maximize the separation among clusters. Common unsupervised applications are object segmentation, similarity detection, automatic labelling (BONACCORSO, 2017).

Another approach also involves using both labelled and unlabelled data, therefore called semi-supervised. That technique can be adopted when it is necessary to categorize an extensive data set with a few labelled examples or when there is needed some constraints to a clustering algorithm, like assigning some elements to a specific cluster or excluding from others (BONACCORSO, 2017).

Reinforcement learning is based on the feedback provided by the environment, even if there are no actual supervisors. In this case, the agent does not have precise measurements of its error, yet a more qualitative feedback, usually called reward (or penalty, if negative). That approach is particularly efficient when the environment is not entirely deterministic, often being very dynamic, and when there is no possibility of having precise error measure (BONACCORSO, 2017).

Deep Learning (DL) is a branch of machine learning based on Artificial Neural Networks (ANNs). DL algorithms model high-level abstractions of input data using a graphical representation that comprises several processing layers; for instance, the detection and recognition of multiple objects are improved with that technology. It supports object classification as well and enables recognition and prediction of actions (FALCINI; LAMI; COSTANZA, 2017). Deep learning can show better performance than other approaches, even without a context-based model, suggesting that sometimes it is better to have a less precise decision made with uncertainty than a more precise one determined by a complex model which is not so fast, according to Bonaccorso (2017). DL applications include image classification, real-time visual tracking, autonomous car driving, logistic optimization, bioinformatics, and speech recognition.

Even though there are many ML algorithms, each one is designed for a particular data-mining task, so knowing the task is essential to define which set of them are recommended for doing it. However, it does not tell exactly which one to use. The data scientist should study the data set and the problem to identify which is the best one to approach it (KELLEHER; TIERNEY, 2018). Finding out if the data has a minimal cluster structure is vital to completely understand the contents of a data set, when clustering is considered (ALHONIEMI *et al.*, 1999).

A good ML algorithm should be able to generalize, which is the ability to perform

well on previously unobserved data. For this, Goodfellow, Bengio, and Courville (2016) assume that the samples in each data set are independent of each other and that the train set and the test set are identically distributed, drawn from the same probability distribution of each other. Within the algorithm, the perfect capacity should be weighted to fit a variety of functions without causing underfitting, which happens when the model is not able to perform with a sufficiently low error value on the training, and without occurring overfitting, which happens when there is a large gap between training error and test error.

It all comes to how well the model can learn and, therefore, predict. Prediction estimates the value for a given instance based on the values of input attributes of that instance. For example, predicting the target attribute for new instances that are not in the training data set are solved by supervised ML algorithms that generate prediction models (KELLEHER; TIERNEY, 2018).

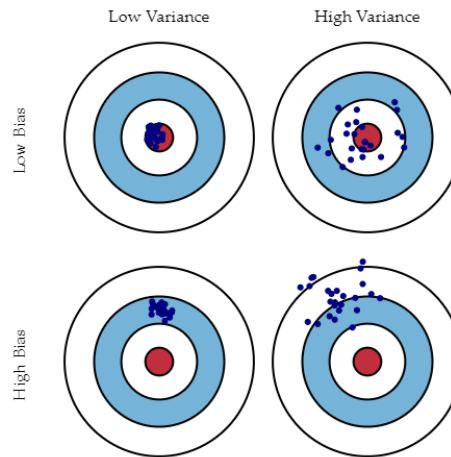
In prediction models, prediction errors can be due to bias error or variance error. Understanding these types of errors can help diagnose model results and avoid overfitting or underfitting. The error due to bias is the error taken as the difference between the expected average prediction of the model and the correct value trying to be predicted, that is, how far off, in general, the models' predictions are from the correct value. The error due to variance is the one taken as the variability of a model prediction for a given data point, that is, how much the predictions for a given point can vary between different iterations. In Figure 1 the bias and variance results are illustrated on a bulls-eye diagram, where at the center is the perfect model with correct predictions (FORTMANN-ROE, 2015).

2.3 Automotive applications

This section will discuss applications of methods and strategies related to machine learning, artificial intelligence, and data science overall on vehicles and the automotive sector found in the literature.

Bersch, Akkerman, and Kolisch (2021) develop a mathematical linear programming model describing the decision problem on the timing of introduction of new products to the market based on the resource-constrained project scheduling problem. With the tool, it can be decided the start of production date for vehicle models, variants, engines,

Figure 1 – Graphical illustration of high and low bias and variance



Source: Fortmann-Roe (2015)

and the assignment of engines to the given variants, taking into consideration different conflicting objectives, new and existing products that rely on shared resources, and also the use of platforms that create interactions between different vehicles through shared modules. The model can analyze trade-offs and efficiently evaluate courses of action in this multi-criteria approach, using real data from a major European company for computational studies.

Countries and enterprises worldwide are investing in automated driving and intelligent vehicles, seeing this ever-increasing urban mobility and modern logistics sector demand. According to Li, Cheng, Guo, *et al.* (2018), advanced AI techniques can solve problems such as traffic congestion, traffic accidents for human errors, road safety, and environmental pollution problems.

Also, for autonomous driving, clustering is highly used to exploit the advantages of multidimensional object size estimation and object classification for automotive radar sensors technologies, usually used as a preprocessing step for classification of the measured data, sometimes in a form of multi-stage clustering (SCHEINER *et al.*, 2019; SCHUBERT *et al.*, 2015; STOLZ *et al.*, 2018).

Theissler *et al.* (2021) analyzed papers from application and ML perspectives and concluded that the majority of papers have relied on supervised methods that require labelled data; that combining multiple data sources can improve accuracy; and that the use of deep learning methods will increase, but it needs efficient and interpretable methods, as well as a large amount of labelled data.

There is already an awareness of the need to integrate deep-learning-based

development with traditional development approaches in the automotive software engineering community. It is growing at the technical, methodological, and cultural levels of companies, already being considered a mature and viable technology (FALCINI; LAMI; COSTANZA, 2017). From image analysis to natural language processing, deep learning algorithms power many innovative products, such as safety or parking assist, self-driving cars, and autonomous emergency braking. Areas like virtual sensing for vehicle dynamics, vehicle health monitoring, automated driving, and data-driven product development are expected to attract the most attention from these algorithms (SINGH; ARAT, 2019).

Unsupervised learning, however, has a more established place, being heavily applied in clustering tasks necessary in many moments in the industry. In the work of Kargari and Sepehri (2012), clustering was used for automotive spare-parts distribution to reduce transportation costs, achieving a cost reduction of 32% with the proposed method, that used K-means in a 3-year data considering three factors in the similarity function: euclidean distance, lot size, order concurrency.

Altintas and Trick (2014) present a study of a data mining and classification analysis of forecasting patterns in a supply chain, where auto manufacturers provide forecasts for future orders and the supplier uses them to plan production in advance. With clustering and pattern recognition analysis, the authors could provide a framework to analyze the forecast performance of the customers.

Yi *et al.* (2019) have used polynomial regression mixture clustering in individual drivers trajectories for learning in-depth driving behaviors, which could discover more than manually defined maneuver due to the ability in accounting for both spatial and temporal information, providing promising intention prediction performance with also being adaptive to different drivers. Also in order to avoid manually designed metrics, Wei Wang *et al.* (2021) proposed to employ network representation learning to achieve accurate vehicle trajectory clustering, where, with learned vehicle vectors, vehicle trajectories are clustered, achieving better performance than baseline methods.

Analyzing driver behavior characteristics is also an aspect of investigation for automotive control, once the driver is the controller and evaluator of the quality of the vehicle path-following. It can be made based on certain pattern recognition provided by simulation or field test data. But, foremost, the driver behavior characteristics need to be classified before identified and, for that, clustering algorithms are generally used, such

as Fuzzy and K-means algorithms (LIN *et al.*, 2014). Analyzing potentially dangerous driving behaviors of commercial truck drivers, Zhou and Zhang (2019) used principal component analysis with Density-based Spatial Clustering of Applications with Noise (DBSCAN) to extract variation properties of speeding behavior, showing that at least 40% of the drivers tended to drive in a substantially dangerous way, which contributes to the development of safety education programs and countermeasures.

To evaluate whether a driving behavior is fuel-efficient, Hao, Yang, and Zhou (2019) proposed a method that uses K-means clustering combined with DBSCAN to cluster four characteristic parameters, which are related to fuel consumption, into three driving behaviors: low, medium and high fuel consumption. With that, authors could have a fuel consumption-oriented driving behavior evaluation model that can evaluate online and give an estimation of whether the driving behavior is fuel-efficient.

Wang and Wang (2020) propose a clustering algorithm based on Genetic Fuzzy C-Means (GFCM) to cluster driving behaviors for hazardous material transportation. The authors used the GFCM algorithm to cluster driving behavior data collected from real-time GPS monitoring devices into different categories, and evaluated the clustering results using various criteria such as within-cluster distance and separation between clusters. They found that GFCM was effective in identifying similar driving behaviors and reducing the impact of outliers. The authors suggest that the proposed algorithm can be used to identify hazardous driving behaviors and provide feedback to drivers to improve safety.

Qi *et al.* (2015) discuss the importance of accurately understanding driving behavior for advanced driving assistant systems. To achieve this, the authors propose using clustering and topic modeling to extract latent driving states from longitudinal driving behavior data collected by instrumented vehicles. The authors employ data mining techniques, including ensemble clustering using the kernel fuzzy C-means algorithm and a modified latent Dirichlet allocation model, to handle the large dataset and extract valuable knowledge. The authors identify three driving states, including aggressive, cautious, and moderate, and develop a quantified structure for driving style analysis. Overall, this approach can provide insight into the common and individual characteristics of driving behavior and improve the development of driving assistant systems.

The article by Lee and Jang (2019) proposes a framework consisting of four steps: data preprocessing, feature extraction, behavior identification, and behavior

evaluation, which can be used to evaluate aggressive driving behaviors using data collected from in-vehicle driving records. The authors used various machine learning algorithms, including decision tree, random forest, and Self-Organizing Map (SOM), to identify aggressive driving behaviors. They found that SOM was particularly effective in identifying complex driving behaviors that could not be easily captured by other algorithms. The authors suggest that this framework can be used to develop personalized driver training programs and improve road safety.

Investigating if trucks are used as intended by the manufacturer, since their usage might impact the longevity, efficiency and productivity, Dahl *et al.* (2020) compared customers' behaviors using logged data with vehicle configurations presets. To do so, they have used Gaussian Mixture Models to cluster and classify behaviors, and then applied Rule-based Machine Learning to examine whether the real behavior of the vehicle matches what was the intended use. With the study, authors were able to identify outliers that should be analyzed.

2.4 Clustering Algorithms

To extract the knowledge from the data, clustering was made with the algorithms presented in this section. It is the essential part of extracting the value of the dataset, and therefore, various techniques have been implemented with different approaches from each other.

According to Jain, Murty, and Flynn (1999), clustering is the unsupervised classification of patterns into groups (clusters), including observations, data items, or feature vectors. It is the partitioning of unlabeled observations into clusters so that points in a group are similar to each other and different from the ones in other groups, according to some similarity criteria (DINLER; TURAL, 2016).

Clustering problems are considered optimization problems and are solved by exact algorithms, approximation methods, or heuristics. According to Dinler and Tural (2016), clustering methods can be broadly represented by these categories: hierarchical and partitional methods.

Hierarchical clustering, which has its roots back in the 1960s and 1970s and is continuously explored, is the algorithm that builds nested clusters by agglomerating or dividing clusters successively (DINLER; TURAL, 2016). They were developed to

overcome some disadvantages of flat or partitional based clustering methods, such as the predefined number of clusters and the non-deterministic nature of the solution, so being the more deterministic and flexible mechanism for clustering (REDDY; VINZAMURI, 2018). Traditional agglomerative hierarchical clustering mentioned are single linkage (nearest neighbor), complete link, group average, McQuitty's method, median method, centroid, and Ward's method; while modern techniques can be classified as random sampling, data condensation, density-based approaches, grid-based approaches, divide and conquer, incremental learning (MURTAGH; CONTRERAS, 2012).

In partition-based clustering, approached in this work, the data is divided into different groups based on similarities and dissimilarities. Standard similarity measures are distance-based, pattern-based, and density-based. In distance-based similarity measures, the relative position of a data element inside a cluster is calculated by a distance function to the center of the cluster, called centroid. To improve the quality of the clusters, the position of the centroid is changed during different iterations, trying to minimize the intra-cluster distance and maximize the inter-cluster distance as an objective function, that is, maximize similarities and minimize dissimilarities (ALAM *et al.*, 2014).

For discovering the underlying structure of unlabeled data objects, prior knowledge about the data can be used in the form of labels or constraints to extract value, as ignoring it may result in drawing irrelevant information for the user (DINLER; TURAL, 2016). For this work, we have used the vehicle's specification as information to help on what to expect and investigate from clusters' outputs, such as knowing what weight the vehicle is built to carry.

For this particular work, partitional methods are the focus. From hierarchical clustering, Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) was implemented on the initial testing stage of the work and did not perform well within the range of number of clusters that we are interested in, that is from 2 to 10 clusters, having density problems on recognizing few number of clusters. So, we have decided to guide the work towards partitional methods, which have returned good results with reasonable number of clusters. According to Pinto and Engel (2015), Gaussian Mixture Models have time complexity $O(NKd^3)$, where N is the number of samples, K is the number of Gaussian components (clusters) and D is the number of dimensions. The high time complexity makes the algorithm prohibitive for high-dimensional tasks

and thus of limited use.

The algorithms chosen were: K-means, for this being the most important partitioning method; Fuzzy C-means, that stands out as one of most widely used soft-clustering method; and Self-Organizing Maps, that is a neural network technique able to give a spatial grouping with different base-method than the other ones (LEE; YUN, *et al.*, 2019; MELIN *et al.*, 2020). The hyper-parameters for these algorithms were adjusted empirically.

2.4.1 K-Means

K-means clustering was proposed by Macqueen (1967) and is the superior technique and one of the foundations of the partitioning approaches (JAIN; MURTY; FLYNN, 1999). The algorithm divides the data set creating convex clusters, which are clusters that, basically, all straight lines between data points within it lie within the cluster.

The data is divided into pre-defined K clusters in K-means with a distance measure calculation. The clusters centers are as far as possible, and each data point within a cluster is most similar to each other as possible. The algorithm starts by assigning each data to a particularly close cluster centroid; then, new centers are calculated considering its points. This process repeats until the centroids remain unchanged in successive iterations or a stop condition is reached, the number of iterations, time, or limitation imposed. However, it is essential to note that the first cluster centroid is positioned randomly and then updated successively after the distance calculation to each data point has been calculated. Partitioning approaches are efficient, but the randomness of the initialization, along with the need to specify the number of clusters in advance, affects the quality of the solution that so depends on the domain knowledge (ALAM *et al.*, 2014). These are significant factors that can impact the performance of the algorithm (REDDY; VINZAMURI, 2018).

K-means is well known for converging fast to local optimum, having its results depending even more on the initialization process (the positioning of the centers). To overcome the high dependency, the algorithm should be initialized with different sets of initial centers for multiple iterations, randomly or not – i.e. using initialization heuristic with searching for a good set of initial centers (CELEBI; KINGRAVI, 2015). Al-Shboul and Myaeng (2009), for instance, use a Genetic Algorithm – a bio-inspired evolutionary

algorithm – to solve that dependency problem by finding the best initialization parameter. Another alternative to avoid poor local solutions that might have few points or empty clusters is adding constraints of minimum points in it. That can be applied to data sets that have 10 or more dimensions, and it is desired 20 or more clusters (BRADLEY; BENNETT; DEMIRIZ, 2000).

The algorithm, for a given dataset $X \subset \mathbb{R}^d$, aims to minimize the objective function Φ , called Sum of Squared Error or inertia, shown in Equation 1, where $X(z) = \{x \in X : z = \operatorname{argmin}_{\hat{z} \in Z} \|x - \hat{z}\|^2\}$, and produce a set of cluster centers $Z = \{z_1, \dots, z_K\}$. After initializing, each center $z \in Z$ is updated following Equation 2, and then each $X(z)$ assignment is updated using the proximity or similarity method defined (MALLE, 2021).

$$\Phi = \sum_{z \in Z} \sum_{x \in X(z)} \|x - z\|^2 \quad (1)$$

$$\forall z \in Z : \{z := \frac{1}{|X(z)|} \sum_{x \in X(z)} x\} \quad (2)$$

The algorithm convergence towards a minimum is proofed by the fact that $c \in \mathbb{R}^d$ minimizing $\sum_{x \in C} \|x - c\|^2$ is the center of cluster C , for any subset that $C \subset X$. When the algorithm is uniformly randomly initialized, there is no guarantee of closeness to the global optimum, given the objective function (MALLE, 2021).

To deal with the initialization problem of falling into local minima, K-means was executed 20 times with different centroid seeds. The best output in terms of the objective function result, which is the distance between each data point to its centroid, was picked as the solution of the run.

The K-means algorithm can be summarized in the basic steps shown in the pseudocode 1. A single iteration of K-means has a time complexity equal to $O(dNK)$, where N is the number of samples, K is the number of clusters, and d is the number of dimensions (DINLER; TURAL, 2016).

Algorithm 1 – K-Means

Require: K // Number of clusters defined
Ensure: $X = \{x_1, x_2, \dots, x_N\}$ // Set of elements
1: Assign initial position for clusters center z_1, z_2, \dots, z_K
2: **while** Convergence criteria is not met **do**
3: Assign x_i to the closest center of Z
4: Calculate new mean for each cluster center
5: **end while**

Source: Mohd et al. (2012).

2.4.2 Fuzzy C-Means

Hard clustering is the clustering method that assigns each data point to one, and only one, of the clusters, assuming that boundaries between the clusters are well-defined. However, real data sets most likely do not have very defined boundaries. They might be fuzzy, requiring a more nuanced analysis of the object's affinity to the clusters (GATH; GEVA, 1989).

The Fuzzy C-Means (FCM) algorithm was first introduced by Dunn (1973), who highlights that it can deviate less easily to uninteresting locally optimal partitions. That algorithm was improved principally by the work of James Bezdek (1981).

Fuzzy sets are defined by indicator functions, in which case they are called membership functions. On hard clustering, it can be said that the data is assigned to the cluster with a degree of membership equal to one, as presented on the indicator of Equation 3, which presents the fuzzy partition matrix. On Fuzzy C-Means, that indicator can have continuous values in between the interval $[0,1]$; that is, each data point can belong to more than one cluster with a given membership degree (GATH; GEVA, 1989; LI; CHENG; LIN, 2008).

$$I_{Z_j}(x) = \begin{cases} 1 & \text{if } x \in Z_j \\ 0 & \text{if } x \notin Z_j \end{cases} \quad (3)$$

According to Gath and Geva (1989), there are three major difficulties during fuzzy clustering of real data:

- The clusters number can not always be pre-defined, having to find a cluster validity criterion to determine the optimal number;
- Initial guesses have to be made, as the character and location of centroids are not necessarily known a priori;
- There is much variability in cluster shapes, sizes, and variations in densities in each cluster.

The algorithm, given a dataset $X \subset \mathbb{R}^d$, with a number of centers $K \in \mathbb{N}$ and a hyperparameter $m > 1$, $m \in \mathbb{R}$, tries to minimize the objective function 4 producing a set of centers $Z = \{z_1, \dots, z_K\}$ with corresponding membership functions μ_1, \dots, μ_K (MALLE, 2021).

$$\Phi(\mu_1, \dots, \mu_K, z_1, \dots, z_K) = \sum_{x \in X} \sum_{j=1}^K \mu_j(x)^m \|x - z_j\|^2 \quad (4)$$

Knowing that there is the condition $\sum_j^K \mu_j = 1$ and applying a Lagrange multiplier, it is found that repeating the redefinitions 5, that update each $z_j, j \in \{1, \dots, K\}$ to become the μ_j -weighted center of X , and 6, that updates each $\mu_j, j \in \{1, \dots, K\}$, provides a local minimum of Φ as an output (MALLE, 2021).

$$\forall j \in \{1, \dots, K\} : \{z_j := \frac{\sum_{x \in X} \mu_j(x)^m x}{\sum_{x \in X} \mu_j(x)^m}\} \quad (5)$$

$$\mu_j(x) := \left[\sum_{k=1}^K \left(\frac{\|x - z_j\|}{\|x - z_k\|} \right)^{\frac{2}{m-1}} \right]^{-1} \quad (6)$$

The hyperparameter m can be called fuzzifier since it controls how fuzzy the clustering will be. From that, it can be assumed the asymptotic results that, for $m \rightarrow \infty$, the fuzziest state, all μ_j are equal, and the representatives coincide at the center of X ; and that, for $m \rightarrow 1$, the FCM returns into a K-Means, where the membership functions give 3. According to Bezdek, Ehrlich, and Full (1984), the useful range of m seems to be close to $[1, 30]$ and, for most data, $1.5 \leq m \leq 3.0$ gives good results. For this study it was set $m = 2$.

The Fuzzy-c-means algorithm can be summarized in the basic steps shown in the Algorithm 2. A single iteration of FCM has a time complexity equal to $O(dNK^2)$, where N is the number of samples, K is the number of clusters, and d is the number of dimensions (KUMAR; SIROHI, 2010; KOLEN; HUTCHESON, 2002).

Algorithm 2 – Fuzzy C-Means Algorithm

Require: K // Number of clusters defined

Ensure: $X = x_1, x_2, \dots, x_N$ // Set of elements

- 1: Assign initial position for clusters center z_1, z_2, \dots, z_K
 - 2: **while** Convergence criteria is not met **do**
 - 3: Update membership matrix with Equation 6
 - 4: Calculate new cluster centers with Equation 5
 - 5: Calculate the new objective function with 4
 - 6: **end while**
-

Source: Alia, Mandava, and Aziz (2011).

2.4.3 Self-Organizing Map

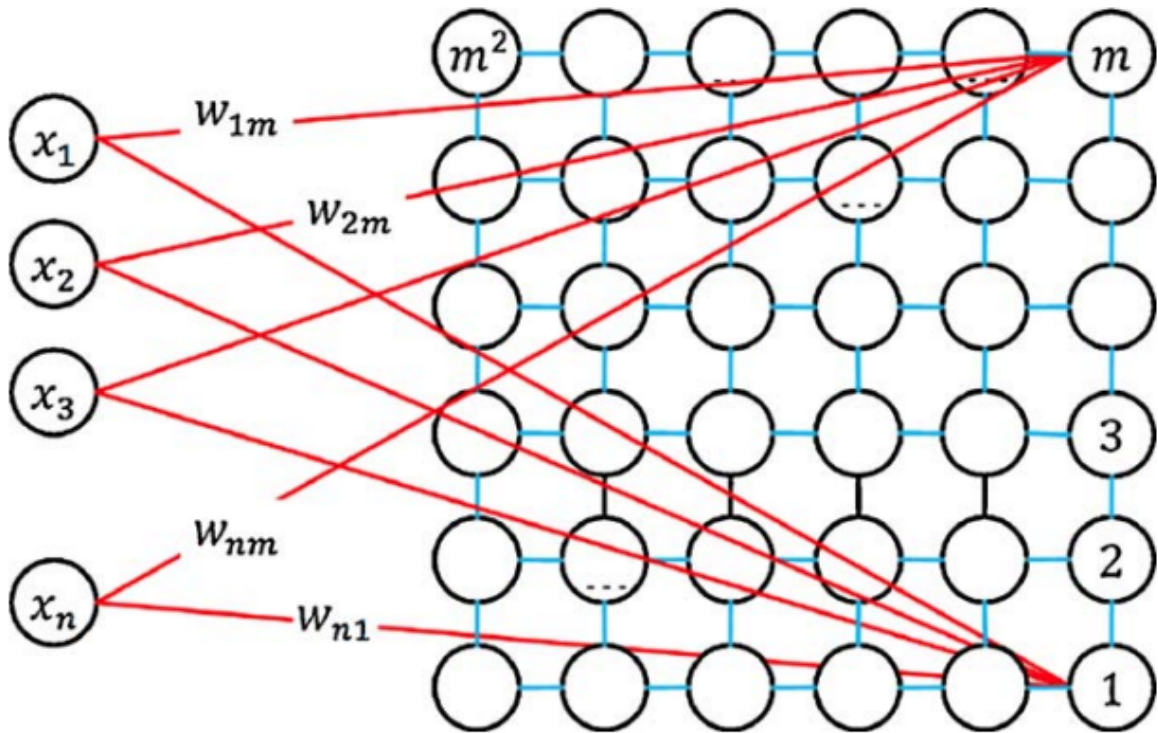
The Self-Organizing Maps (SOM) is a neural network model introduced in the 1980s by Kohonen, that is adjusted using unsupervised learning (KOHONEN, 2001). Mostly, the SOM is applied for pattern recognition, but it can also be applied as unsupervised learning and dimensionality reduction (BAÇÃO; LOBO; PAINHO, 2005).

SOM, or also Kohonen neural network, is a good method to analyze and visualize high dimensional, multivariable data, and also when there is a significant amount of unknown values, being able to learn and cluster the data, recognizing different patterns (JUNTUNEN *et al.*, 2013; ALHONIEMI *et al.*, 1999). This method has advantages when it becomes of dealing with non-linearity of a system, where there is noisy, irregular or missing information, and can be easily and quickly used with visualization resources (HONG; ROSEN, 2001).

The SOM network combines an input layer and competitive layer of processing neurons, which is normally organized as a two-dimensional grid of neurons or cells arranged on a rectangular or hexagonal sheet. A regular $M = m \times n$ neuron matrix, where $m = n$, is presented in Figure 2. The m^2 neurons map a high-dimensional input vector $X(t) = [x_1(t), x_2(t), \dots, x_N(t)] \in \mathbb{R}^d$ into a two-dimensional plane, but through different weight vectors $w_i(t) = [w_1(t), w_2(t), \dots, w_W(t)] \in \mathbb{R}^d$, with size $W = M \times d$, (ZHANG; LI, 1993). Each input x_i is connected to each neuron in the matrix, while each synaptic weight w_{ij} is connected to the i th input component and associated to the j th neuron.

The algorithm progresses in two stages, the similarity matching phase and weight adaptation phase. In the beginning, weights are randomly defined, a first pattern for the input nodes is introduced, and then the euclidean distance between the input and the weights associated with the output are calculated. The neuron w^* with the shorter distance is chosen, along with its geographic neighborhood $B(w^*)$, as according to Equation 7. The input weights of the chosen node and its neighbors are updated following the Equation 8, where $\alpha(t)$ is the learning rate function that decreases exponentially over the iterations, as presented in Equation 9, being α_0 the initial learning rate set and T the number of iterations defined. The function that defines the neighborhood order $B(t)$, given an initial neighborhood topology b_0 , is presented in Equation 10 (GHASEMINEZHAD; KARAMI, 2011).

Figure 2 – Basic structure of SOM neural network



Source: Ghaseminezhad and Karami (2011).

$$D_{min}(t) = \min\{D_i(t)\} = \min \left\{ \sum (x_j(t) - w_{ij}(t))^2 \right\} \quad (7)$$

$$w_i(t+1) = w_i(t) + \alpha(t) \times (x - w_i(t)), \forall i \in B_j \quad (8)$$

$$\alpha(t) = \alpha_0 e^{-\frac{t}{3T}} \quad (9)$$

$$B(t) = b_0 e^{-\frac{t}{3T}} \quad (10)$$

The SOM algorithm can be summarized in the steps shown in the Algorithm 3. A single iteration of SOM has a time complexity equal to $O(Md)$, where M is the neuron matrix size $M = m \times n$ and d is the number of dimensions of the input vector X (MORAES *et al.*, 2012).

Algorithm 3 – Self-Organizing Map Algorithm

Require: K // Number of clusters defined
Ensure: $X = x_1, x_2, \dots, x_N$ // Set of elements

- 1: Create initial set of neuron prototypes $W = w_1, w_2, \dots, w_W$
- 2: **while** Convergence criteria is not met **do**
- 3: Select $x \in X$ randomly
- 4: Get w^* with shorter distance with Equation 7
- 5: **for all** $w \in B(w^*)$ **do**
- 6: Updates w with Equation 8
- 7: **end for**
- 8: Decrease learning rate α according to Equation 9
- 9: **end while**

Source: Günter and Bunke (2002).

2.5 Evaluation Metrics

Evaluation methods help compare the performance of different clustering algorithms and determine the optimal number of clusters on algorithms that do not have an internal estimation. It is often unclear which quality index to use in which case, since there is no unifying protocol for clustering evaluation (TOMAŠEV; RADOVANOVIĆ, 2016).

In general, the cluster number is unknown, but validity indices can be used to find the best number. In the literature many indices had been proposed, such as Silhouette Width (SW), Bayesian Information Criterion (BIC), Akaike Information Criterion (AIC), Dunn's index (DI), Davies-Bouldin Index (DB), Calinski and Harabasz Index (CH), Gap statistic, Generalized Dunn's Index (DN_g) and Modified Dunn's index (DN_s) (SINAGA; YANG, 2020).

An estimator should exhibit low bias, the expected deviation from the true value, and relatively low variance, which measures the deviation from the estimator value that any data sampling is likely to cause. Analyzing a Bernoulli distribution, the variance of the estimator decreases as a function of the number of examples in the dataset (GOODFELLOW; BENGIO; COURVILLE, 2016).

While working with high-dimensional data, some problems arise, denoted by Bellman (1961) as the "Curse of Dimensionality". The increase in dimensionality induces an increase in the containing volume that leads to sparsity. It makes the data difficult to handle and is harder to obtain reliable density estimates, requiring more data to derive statistically sound estimates. Usually, not even the big datasets for large-scale industrial applications have enough data to overcome these problems (TOMAŠEV;

RADOVANOVIĆ, 2016).

Standard clustering methods designed for low-dimensional data can be applied in conjunction with subspace methods to perform partitioning in lower-dimensional feature subspace to perform well in many dimensions, since high-dimensional data exhibit different properties than low-dimensional data. These methods include hybrid approaches such as density-based techniques, K-means, and decision trees. Hubness-based clustering has been successfully applied for high-dimensional clustering problems like document clustering (TOMAŠEV; RADOVANOVIĆ, 2016).

A lot of internal validity indices were introduced in the late '50s and '60s as clustering popularity grew, some of them as an objective function for pattern classification methods (BALL; HALL, 1965; KRZANOWSKI; LAI, 1988). Gower (1967) compares and evaluate the most renowned by the time, and even analyze ones derived by these different methods, showing that the clustering criteria of all of those are defined in terms of distance between centroids of clusters.

Determining the number of clusters K is a cluster validity key problem. To get the optimal number, K is optimized by validity criteria. According to Zhao and Fränti (2014), given a data set $X = x_1, \dots, x_N$, a clustering algorithm and a fixed range of number of clusters $[K_{min}, K_{max}]$, the optimal cluster number can be determined with the following procedure:

- Repeat the clustering process for the number of clusters predefined from K_{min} to K_{max} ;
- With the clustering results, calculate index values with validity methods;
- Select the K that is the best result according to criteria: minimum, maximum, or knee point;
- Use that with external information, that not feeds the model, to compare and validate.

The maximum and minimum values criteria are more straightforward methods. However, when using a knee, or inflection, that process can be called the Elbow Method, which essentially is the scan for the number of clusters that increase no longer returns much better results for the validity index analyzed. That inflection point resembles an elbow, so giving the method's name. Some methods help to find the knee point, such as the L-method, which checks for the closest boundary between the pair of straight lines that fit the curve (ZHAO; FRÄNTI, 2014).

In this work, three internal validity indices were used to analyze clustering methods: Sum of Squares Within Clusters (SSW), Sum of Squares Between Clusters (SSB), and Silhouette Index (SI), presented in the following sections; and one algorithm quality metric to evaluate the clustering algorithm's performance: k-Fold Cross-Validation. These are simple, direct, and practical methods, well known for determining the best number of clusters (KRZANOWSKI; LAI, 1988). The first two serve as the base for many quality indices and are more stable than min-max functions, according to Zhao and Fränti (2014). SSW and SSB can also be used to maximize clustering initialization performance (GUPTA; CHANDRA, 2019).

According to Tomašev and Radovanović (2016), other clustering quality indexes recommended as an alternative are: Dunn's index, Davies-Bouldin index, inverted Davies-Bouldin index, Isolation index, C -index, $C\sqrt{K}$ index, Calinski-Harabasz index, Fowlkes-Mallows index, Goodman-Kruskal index, G_+ index, Hubert's Γ statistic, McClain-Rao index, PBM index, Point-biserial index, RS index, Rand index, SD index, τ index.

Of these, Dunn's index, inverted Davies-Bouldin index, Hubert's statistic, PBM index, and point-biserial index are notable for having the estimated clustering quality increasing as increases the dimensionality of the data, becoming more precise. Also, for many-dimensional problems, better handling of hub points may increase overall clustering quality (TOMAŠEV; RADOVANOVIĆ, 2016).

2.5.1 Sum of Squares Within Clusters (SSW)

The first, Sum of Squares Within clusters (SSW), also known as Sum of Squared Error, is an index that measures the squared average distance of all points within a cluster to the cluster centroid, as shown in Equation 11, where N are dimensional points, $Z = \{z_1, \dots, z_K\}$ are centroids of clusters, K the number of clusters and z_i is the i th cluster. Since it is not a normalized metric, the more compact the cluster is, or the lower the SSW value, the better because it tells that the cluster has few outliers and much more similarities within the group. Also called "inertia", it can be recognized as a measure of the internal coherence of clusters and works better with convex clusters, poorly responding to elongated clusters or with irregularly shaped clumps (ZHAO; FRÄNTI, 2014).

$$SSW = \sum_{z \in Z} \sum_{x \in X(z)} \|x - z\|^2 \quad (11)$$

2.5.2 Sum of Squares Between Clusters (SSB)

The second evaluation metric chosen is the Sum of Squares Between clusters (SSB), which measures the average squared distance between all centroids by finding the Euclidean distance from any cluster centroid z_i to all the other cluster centroids. This calculation is made for all the clusters and summed, as shown in the Equation 12, where \bar{X} is the center of the entire data set, just as Equation 13 presents (ZHAO; FRÄNTI, 2014).

A considerable value indicates that clusters are spread out, as the opposite shows that they are close to each other. For this metric, a higher value is desired, which represents that the clusters have more external separation, that is, are more different from each other.

$$SSB = \sum_{i=1}^K \|z_i - \bar{X}\|^2 \quad (12)$$

$$\bar{X} = \sum_{i=1}^N \frac{x_i}{N} \quad (13)$$

2.5.3 Silhouette Index

The Silhouette Index (SI) is a clustering quality score that gives punctuation for each point and calculates the final score as an average of the point-wise quality estimates. As shown in Equation 14, each point-wise estimate for an $x_p \in z_i$ in space is derived from $a_{i,p}$ e $b_{i,p}$, presented in Equation 15, which respectively are the average distance to other points within its own cluster and the minimal average distance to points from other different cluster (TOMAŠEV; RADOVANOVIĆ, 2016).

$$SI = \frac{1}{N} \sum_{p=1}^N \frac{a_{i,p} - b_{i,p}}{\max(a_{i,p}, b_{i,p})} \quad (14)$$

$$\begin{cases} a_{i,p} = \frac{1}{|z_i|-1} \sum_{x_q \in z_i, q \neq p} \|x_q - x_p\| \\ b_{i,p} = \min_{j \in \{1 \dots K\}, i \neq j} \left(\frac{1}{|z_j|} \sum_{x_q \in z_j} \|x_q - x_p\| \right) \end{cases} \quad (15)$$

The time complexity of this index criterion is $O(dN^2)$, which is a high demand on computational cost, being a problem to scale to large data sets. To avoid problems like lack of memory or exorbitant time to calculate it, there is the Simplified Silhouette Index, which is an approximation of the standard Silhouette Index that uses intra-cluster inter-cluster distances as distances to the respective cluster centroids. It can speed up significantly the coefficient calculation, with overall time complexity of $O(dNK)$ (TOMAŠEV; RADOVANOVIĆ, 2016).

The usual construction of SI is not applicable directly to fuzzy partitions, once it requires crisp cluster boundaries to compute average distances. SI can validate a fuzzy partition after being defuzzified by setting the maximum membership degree to one and the rest to zero. This practice discards cluster overlapping and is not the best way to deal with the problem (RAWASHDEH; RALESCU, 2012).

However, Campello and Hruschka (2006) introduced an extension that integrates within silhouettes the fuzzy values into an average silhouette-based index by computing a weighted mean that weights each silhouette by the difference of the two highest fuzzy membership values of the respective associated point. This reveals those regions with high data densities, once it increases the importance of points concentrated in the vicinity of the cluster and reduces the importance of objects that lie in overlapping areas. The proposal is shown in Equation 16, where μ_{px_j} and μ_{qx_j} are the first and second-largest elements of the fuzzy partition matrix of the data point x_j , SI_{x_j} is its Silhouette Index score and α is a weighting coefficient that levels the impact of the membership calculation in the SI score. The particular case of $\alpha = 0$ results in $FS = SI$, while increasing the coefficient moves FS away from SI , by reducing the relative importance of data objects in overlapping areas, and also tends to reveal sub-clusters with higher data densities, if they exist. In the present work, it was defined $\alpha = 1$, taking into consideration the work of the index author.

$$FS = \frac{\sum_{j=1}^N (\mu_{px_j} - \mu_{qx_j})^\alpha \cdot SI_{x_j}}{\sum_{j=1}^N (\mu_{px_j} - \mu_{qx_j})^\alpha} \quad (16)$$

Overall, points around cluster centers have higher weights since they are closer to cluster centers, while the opposite happens to distant or outliers points. Because of that, when analyzing the difference between FS and SI results, we can see how much overlapping the clusters have, since a high difference means that there is more

overlapping data points. So, even not being thorough in considering overlapping regions, that Fuzzy Silhouette (FS) calculation tends to return a fairer silhouette value than the regular one for the FCM case.

2.5.4 k-Fold Cross-Validation

To evaluate the performance of fitted models, the prediction error calculation is necessary. For that, cross-validation is widely used, which in case the k-Fold cross-validation, from a computational standpoint, may be preferred (FUSHIKI, 2009). Other procedures that can be mentioned are Resubstitution Validation, Hold-Out Validation, Leave-One-Out Cross-Validation, Repeated k-Fold Cross-Validation (REFAEILZADEH; TANG; LIU, 2016).

Using k-Fold cross-validation, a data set partition is formed by dividing it into k non-overlapping equally, or nearly equally, sized subsets. Then, by taking the average test error over k trials, the test error can then be estimated. In the trial i , the i -th subset of data is used as the test set, and the rest of the data is used as the training set (GOODFELLOW; BENGIO; COURVILLE, 2016).

To track the performance of each learning algorithm for each fold, an evaluation metric of accuracy needs to be determined. This will lead to k values of the evaluation metric that need to be aggregated, for example, by averaging or using these samples in a statistical hypothesis test that shows which algorithm is superior. In data mining and machine learning, the most used is a tenfold cross-validation, $k = 10$, since making predictions using 90% of the data is a good percentage to be generalizable to the complete data and measuring the accuracy of the rest 10% is acceptable, taking into consideration that a statistically sound experimental design must provide for the algorithm performs many independent measurements that are sufficiently sizeable (REFAEILZADEH; TANG; LIU, 2016).

The k-Fold Cross Validation is also used to estimate the generalization error of a learning algorithm when the data set is too small for a simple training/testing or training/validation division to produce an accurate estimation of the generalization error (GOODFELLOW; BENGIO; COURVILLE, 2016). The cross-validation method has also been used for model selection, but a better estimate of prediction error does not necessarily provide a better criterion for model selection (FUSHIKI, 2009).

The cross-validation was first used as a clear statement on the work of Mosteller and Wallace (1963) and was further employed as a means for choosing proper model parameters by Geisser (1975) and Stone (1974). However, it was in the work of Larson (1931) that the idea was originated, using one sample for regression and a second for prediction.

2.5.4.1 Homogeneity Score (HS)

According to Rosenberg and Hirschberg (2007), to satisfy homogeneity, all clusters of a clustering result may contain only data points members of a single class. The distribution of class within each cluster must be inclined to a single class, resulting in zero entropy. By examining the conditional entropy $H(C_1|C_2)$ – presented in Equation 17 – of the group distribution with identified classes C_1 and clusters C_2 , it can be determined how close a given clustering solution is to the ideal, wherein the perfectly homogeneous case $H(C_1|C_2)$ is 0. However, the size of this value for not perfect situations depends on the size N (number of data points) of the dataset and the class sizes distribution n . That value is then normalized by the maximum reduction in entropy available, $H(C_1)$, presented in Equation 18. When the overall class distribution is equal to the class distribution within each cluster, which means that the clustering brings no new information, the $H(C_1|C_2)$ is maximal and equals $H(C_1)$. Considering a degenerate clustering solution that there is only a single class type, where $H(C) = 0$, the homogeneity is 1. When assigning each data point to a different cluster, or simply when all clusters contain only members of a single class, the $H(C_1|C_2)$ is 0, and there is a perfectly homogeneous clustering.

$$H(C_1|C_2) = - \sum_{k=1}^{|C_2|} \sum_{c=1}^{|C_1|} \frac{a_{ck}}{N} \log \frac{a_{ck}}{\sum_{c=1}^{|C_1|} a_{ck}} \quad (17)$$

$$H(C_1) = - \sum_{c=1}^{|C_1|} \frac{\sum_{k=1}^{|C_2|} a_{ck}}{n} \log \frac{\sum_{k=1}^{|C_2|} a_{ck}}{n} \quad (18)$$

To make the Homogeneity Score (HS) adheres to the convention of 1 being the desired value and 0 undesirable, the homogeneity is defined in Equation 19.

$$HS = \begin{cases} 1 & \text{if } H(C) = 0 \\ 1 - \frac{H(C|K)}{H(C)} & \text{else} \end{cases} \quad (19)$$

Since in this work is made an unsupervised approach, not having the true labels to identify class types, we have considered the data trained with 100% of it as the true class ($H(C_1)$) and the k-Fold trained data as the clustering result $H(C_2)$. The final HS is calculated as an average of the obtained homogeneity score of the k trials. With this, we can measure the consistency of each algorithm on returning coherent clustering given non-trained data. This is important since the size of the data sets are subject to constant changes while new vehicles are introduced to them. The lower are the values presented in this index, the more it will be necessary to retrain the data set as it increases.

2.5.5 Coefficient of Variation

To analyze the dispersion of the solutions in terms of the evaluation metrics proposed, the Coefficient of Variation (CV) is measured for every number of cluster K tested in the algorithms. With the CV we are able to see the extent of variability in relation to the mean of the result, where the higher the CV, the greater is the dispersion.

The CV is presented in Equation 20, where SD is the standard deviation and M is the mean of the input, which in this case are the results for each metric running 10 times for every number of cluster K . So, for example, when $K = 4$, every algorithm is executed 10 times, returning 10 values for every metric. Then, for each metric, SSW for instance, is calculated the CV of the 10 results found. Finally, these CVs results are presented in a table, where the dispersion and consistency of each algorithm result are discussed.

$$CV = \frac{SD}{M} \quad (20)$$

3 MATERIAL AND METHODS

This chapter will describe all the development of algorithms and methods, with examples and explanations. The chapter presents the methodologies employed, the models implemented, error analysis, and comments. All the algorithms were implemented and executed on Python 3.9 using available libraries (PEDREGOSA *et al.*, 2011; DIAS, 2019).

Aimed at achieving the objectives defined, the following steps have been established:

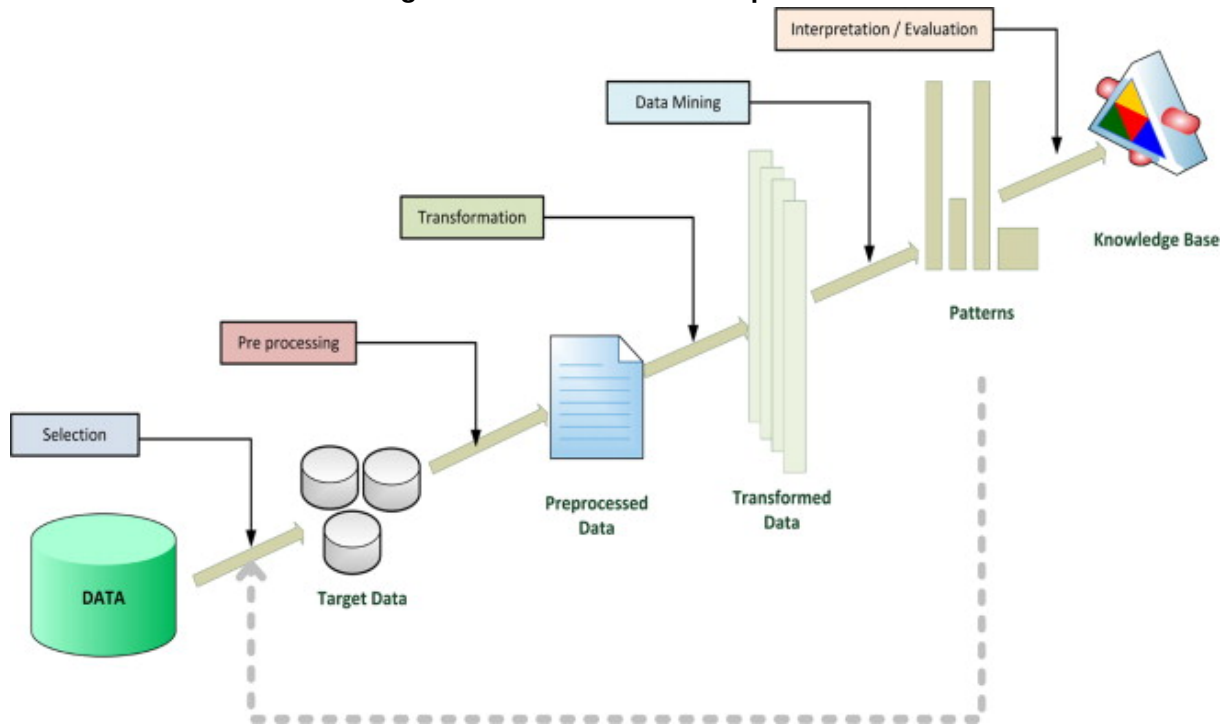
- To review the literature on the subject;
- To treat, normalize and cluster the data sets using the methods researched;
- To compare the results found using plots and figures, highlighting the best results for each parameter;
- To approve or disapprove proposed methods;
- To understand the impact of the work and what contributions it made.

Fayyad, Piatetsky-Shapiro, Smyth, *et al.* (1996) call Knowledge Discovery and Data mining (KDD) the process of automatically searching large volumes of data for previously unknown, but exciting and informative patterns, using modern information exploration techniques, but also statistics, machine learning, and pattern recognition (ALAM *et al.*, 2014). Analyzing the data from various angles and categorizing and summarizing it are the basic principles of data mining (SINGHAL; JENA, 2013).

The KDD process, presented in Figure 3, starts with data selection, specifying the scope of the data. Then it is analyzed and preprocessed to enhance its reliability, remove irrelevant data, handle missing values, and often remove outliers observations. In the third phase, it is transformed, including sampling and feature selection. The transformed data then is exploited by data mining methods and post-processed, extracting informative patterns, such as clusters, classification and association rules, sequential patterns, or prediction models. Finally, the interpretation and evaluation of the results are made.

Before starting building any ML models, it is recommended to understand the data entirely, the goals of the project, and how it will be deployed, considering any limitations that need to be addressed and what has already been done in the research field. Talking to domain experts, surveying the literature, and exploring the data can help. Looking at the data is crucial since it can give insights to help model. However, it

Figure 3 – Illustration of KDD process



Source: Alam *et al.* (2014)

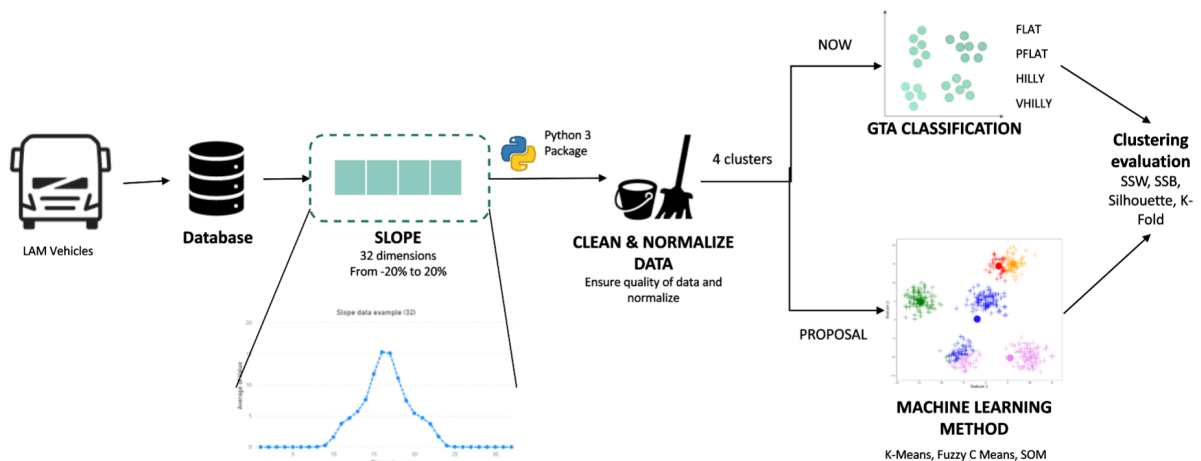
is even more critical that any not tested or untestable assumptions that eventually are made do not feed into the model. Also, the data scientist should avoid looking closely at any test data in the initial exploratory stage, because, consciously or unconsciously, these assumptions made can limit the generality of the model (LONES, 2021).

For this project, the data path can be resumed in the Figure 4. A database is fed with informations constantly measured on vehicles, where we can extract some of them, such as the slope data that each vehicle has been through. This data set is cleaned and normalized to ensure quality, to then be clustered correctly. For the slope data set exemplified, the proposal is to change a ruled-based method named GTA Classification to a machine learning method with the aid of clustering evaluation techniques.

3.1 Data preprocessing

Real-world data are primarily dirty, incomplete, and noisy. Incomplete in terms of lacking attributes, values, or attributes of interest; noisy on containing errors, outliers, and inconsistencies on names or values; and dirty for having hardware, software, or human errors, as data entry errors may occur. Missing values, impossible data combinations, of the range values are problems that can produce misleading results. Data preprocessing

Figure 4 – The whole data path



Source: Own authorship (2022).

is the way to solve those problems with cleaning, normalization, transformation, feature extraction, feature selection, etc., delivering a final and clean training set (SINGHAL; JENA, 2013). Notably, preprocessing can be very helpful in image processing since, despite the extra computational complexity cost, it can alleviate the deteriorating effects of inherent artifacts (JOSEPH; OLUGBARA, 2022).

According to Singhal and Jena (2013), data preprocessing can be divided into a few steps: Data cleaning, Data integration, Data transformation, and Data reduction. Cleaning, the first step, is removing incorrect values and checking the consistency of the data. With the dirty removed, data integration is the step to combining the data from databases, files, and different sources. Then, in Data transformation, the data is modified to fit the output system. Finally, the last step is to reduce the data to a smaller size but with the same analytical results. An additional step inside the Data reduction that can be mentioned is the data discretization, which refers to converting or partitioning continuous attributes to discretized, which is useful when creating probability mass functions.

Considering certain independent variables that have little or zero effect on the dependent variables and have no importance for the data itself, it is wise to remove them from the model as it is bound to increase the cost of data collection observation and, therefore, model application. Applying feature extraction can bring cost efficiency, paying with a decline in the accuracy of estimation and prediction (YANG *et al.*, 2018).

According to Han, Kamber, and Pei (2012b), data normalization attempts to give equal weight to attributes from a feature. It comes in handy for classification

algorithms that involve neural network or distance measurements like clustering and nearest-neighbor classification. In practice, for example, the normalization process on distance-based methods can help prevent attributes that have initially large ranges from outweighing those with smaller ranges; also, normalizing the input for neural network back-propagation algorithm for classification mining will help speed up the learning phase.

In general, normalization is an essential process of data mining and is very useful when there is no prior knowledge of the data. Gavali and Banu (2019) answer the question "why normalization should be performed" by exemplifying that this is related to activation functions as, for example, a sigmoid function that makes an input value range from 0 to 1 as output. According to Patel and Thakral (2016), a normalized data needs lesser number of iteration and offers better outcome compared to non-normalized data, in most of the cases.

Some normalization techniques are Linear normalization (also called linear scaling or Min-Max), which is used when the feature is close to a uniform distribution across a fixed range; Clipping, which is suitable to remove extreme outliers; Log Scaling or decimal scaling when the feature is conformed to power-law; Z-Score, useful when the distribution does not contain extreme outliers (DEVELOPERS, 2021; AKANBI; AMIRI; FAZELDEHKORDI, 2015; JAVAHERI; SEPEHRI; TEIMOURPOUR, 2014).

3.2 The addressed datasets

3.2.1 Slope

The first data set to be treated refers to road slope conditions for nearly, 70000 vehicles. The data set has years of storage, tracking all the road inclination that vehicles have been through. With that data clustered, it is expected to see the different environment applications of the vehicles, which, intertwined with other information, can deliver excellent knowledge of the usage and product development. Nevertheless, to work with this data, some preprocessing must be done.

The data set has 32 features, which represent the number of kilometers spent on each range of inclination, that varies from the range $[-\infty, -20\%]$, in dimension 1, to $[-1, 0\%]$, in dimension 16; and from $[0, 1\%]$, in dimension 17, to $[20\%, \infty]$, in dimension

32. That is, the negative ranges, dimension 1 to 16, are the same, in module, as the positive ranges, dimension 17 to 32, as presented in the Table 1.

Table 1 – Slope dimensions

| | Very high slope | | | | | | | High slope | | | Medium slope | | | Low slope | | |
|-----------|-----------------|----|----|----|----|----|----|------------|----|----|--------------|----|----|-----------|----|----|
| Negatives | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| Positives | 32 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 |

Source: Own authorship (2022).

The first job is to clean the data and eliminate rows with empty values. After, the data is transformed into a tabular shape, where each row is a vehicle and each column a feature. To better comprehend the data, each feature value v_i is then transformed to a percentage of the total driving distance, following the Equation 21, where d is the number of dimensions. In the end, for every vehicle vector, each of the attributes is represented as its percentage of the total travelled distance; that is, each data point is turned into a weight representation.

$$v_i = \frac{v_i}{\sum_i^D v_i} \quad (21)$$

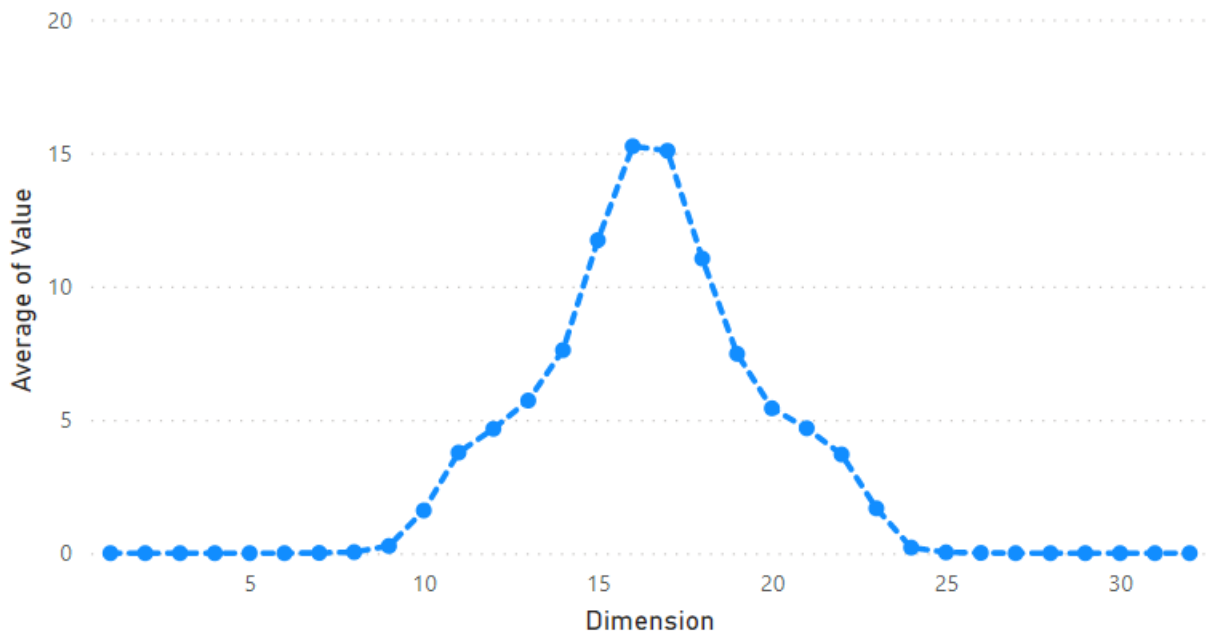
In this work, a Min-max normalization is performed. That method executes a linear transformation on the data, preserving the relationships between the original data values. The normalization is presented on Equation 22, where $\max(X_i)$ and $\min(X_i)$ is respectively the highest and lowest value for each i attribute (dimension) in the data set (HAN; KAMBER; PEI, 2012b,a).

$$x_i = \frac{x_i - \max(X_i)}{\max(X_i) - \min(X_i)} \quad (22)$$

As a result, a complete normalized data set is ready to be clusterized. In the figure 5, it is shown an example of a final normalized vehicle row, where in the X-axis are the dimensions, each one an interval of inclination, and in the Y-axis are the percentage of the total traveled time. This vehicle presented, for example, around 15% of the traveled distance in the slope dimension 15 and 16, which are the flat intervals -1% to 0% and 0% to 1%.

The data is then submitted to the clustering process. At that stage, the data is clusterized with K clusters varying from 2 to 10, so that an Elbow Curve can be used to evaluate the algorithms used, K-Means, FCM and SOM. Those and other results are presented in the next chapter.

Figure 5 – Example of a vehicle's slope vector
Slope data example (32)



Source: Own authorship (2022).

3.2.1.1 Correlation analysis

Once studying the data, we've seen that vehicles behave similarly in the same negative and positive ranges, i.e. $[-1\%, 0\%]$ and $[0\%, 1\%]$ (dimension 16 and 17). The individual data as a whole, just as shown in figure 5, may look horizontally mirrored in the center of X-axis, or, looking at the table 1, in between dimension 16 and 17, so that each relative positive and negative dimension would behave the same way.

To verify this assumption, a correlation analysis is made. If the hypothesis is proven, we can sum each related dimension values, which will reduce dimensionality from 32 to 16. It can help the performance of clustering algorithms, since reducing dimensions avoid the curse of dimensionality problem (AGGARWAL; REDDY, 2013).

For this analysis, it was used the Pearson correlation coefficient (PCC), developed by Pearson (1895), to measure the linear correlation between two sets of data. It will essentially measure the ratio between the covariance and the product of the standard deviations of two variables a and b , as presented in Equation 23, where $E(ab)$ is the cross-correlation expectation between a and b , presented in Equation 24, while σ_a and σ_b is the standard deviation of a random particle a and b (BENESTY *et al.*, 2009). In Equation 24, v_1, \dots, v_N are the possible outcomes of a random variable v , and e_1, \dots, e_N

are the respectively probability. $E(ab)$ is considered a weighted average of the v_i , given their probabilities (weights) e_i , since it satisfy $e_1, \dots, e_N = 1$ (BILLINGSLEY, 1995).

$$\rho(a,b) = \frac{E(ab)}{\sigma_a \sigma_b} \quad (23)$$

$$E(a,b) = \sum_{i=1}^N v_i e_i \quad (24)$$

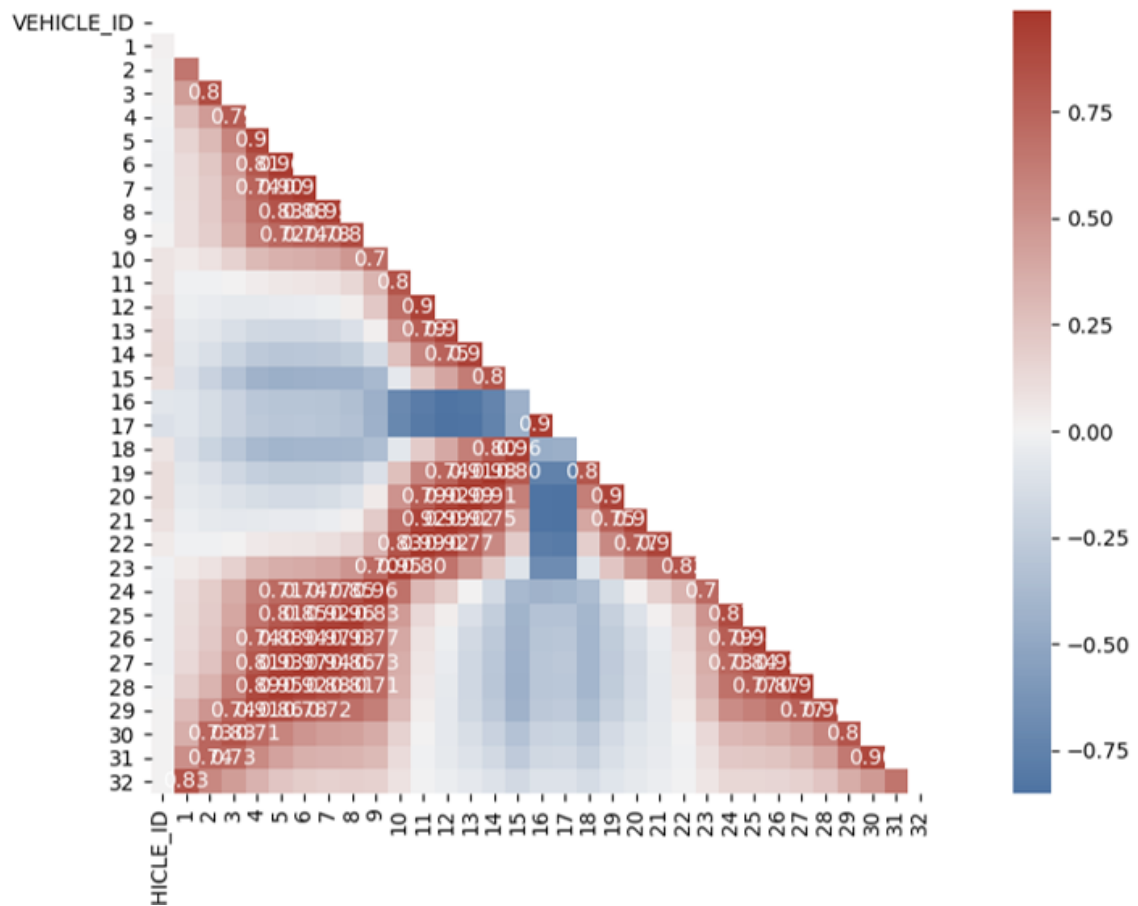
Figure 6 presents the result of the PCC applied to the slope data set. In both axis, X and Y are the dimensions that we are looking for correlation in between. In the right of the graphic, there is the gradient color legend: red for high positive correlation and blue for high negative correlation, while 0 is white for no correlation. A positive correlation indicates that the two variables tend to move in the same direction, and a negative correlation indicates that the variables tend to move in opposite directions. For this study, we have considered the grades above 0.7 as high correlation, being these labeled in the graph.

Analyzing the results, we see some areas of high correlation detected. The first are dimensions that are side-by-side, which may have obvious correlation just as no correlation happens when comparing a high inclination to a low inclination, as, for example, 1 (very high inclination) with 15 (very low inclination). Also, we see a high inverse correlation happening between dimensions 16 and 17, which are the closest to 0 % of inclination, with dimensions 10 to 15 and 18 to 23 that are medium and low, but not the lowest, inclinations. That information means that vehicles that leaves very flat terrains, decreasing the values in dimensions 16 and 17, mostly go to those pre-mid and mid-slope levels, increasing the values in the respective dimensions, and vice versa. This is important information to be analyzed further in the clusters.

The third area of high correlation is the region that we are interested in: the relative positive and negative inclinations, 1 to 16 with 17 to 32. To better analyze this region, marked in the Figure 7(a), we take a closer look as presented in the Figure 7(b). With that, we see that the relatives dimension scores, in the diagonal, have very high positive correlation, the majority above 0.95 of correlation.

With the confirmation of the hypothesis, we join the relatives dimensions, so that we have 16. The new values for each new dimension are the sum of the values of the previous dimensions. The example showed in Figure 5 would turn into the data

Figure 6 – Pearson correlation of slope data set



Source: Own authorship (2022).

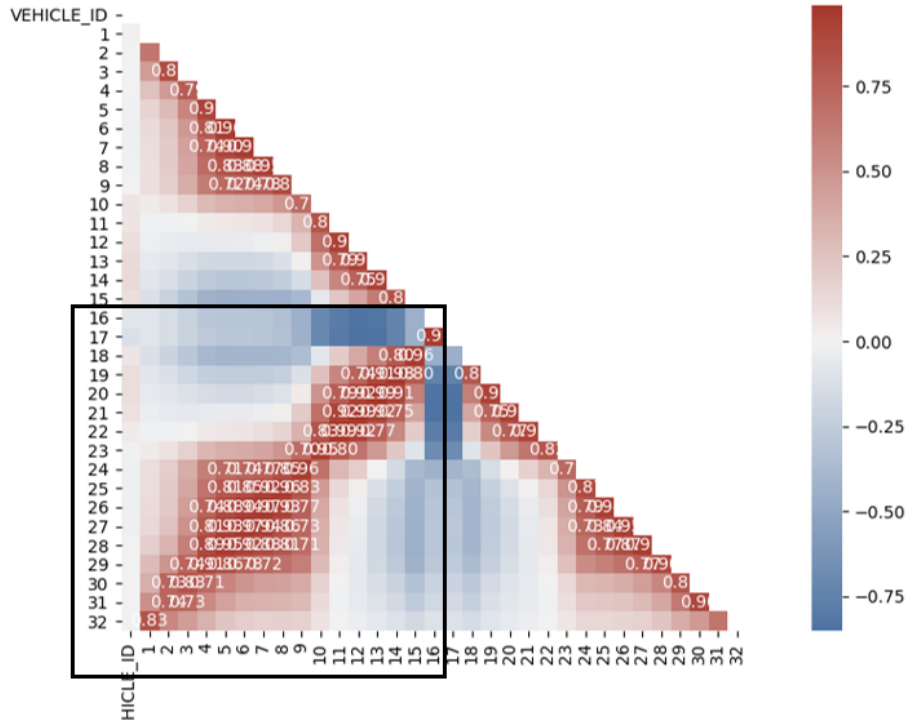
presented in Figure 8. In the next chapter, this transformed data set is then compared with the original to finally confirm the gain on reducing the number of dimensions.

3.2.2 Speed

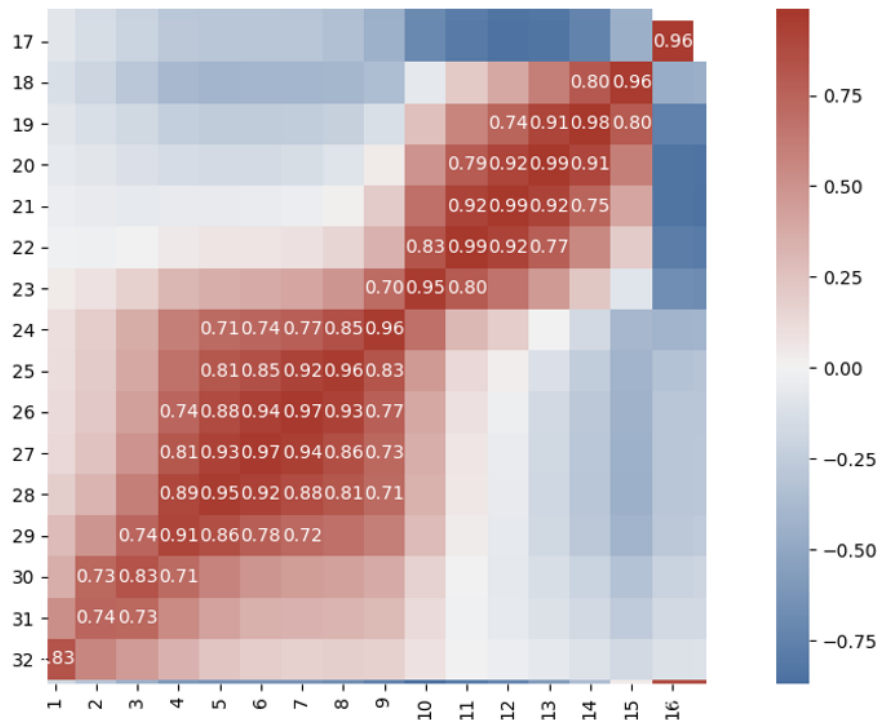
The second data set analyzed refers to speed log data. Different speed profiles may come from distinct product specifications, from weight carried, from road inclination, or just driving aspect. With that data set clustered, we are able to see closely which trucks work on same speed conditions, or also, when placed in parallel with other specifications, which truck works different from what is expected.

The data comes distributed in 20 ranges of speed, that start from [0 Km/h, 5 Km/h], in dimension 1, and end in [119 Km/h,∞], in dimension 20, as presented in Table 2. The first position is the range that the vehicle is idle or starting it, then we have the low speed ranges, medium speed ranges, high speed, and finally very high speed

Figure 7 – Pearson correlation of slope data set zoom



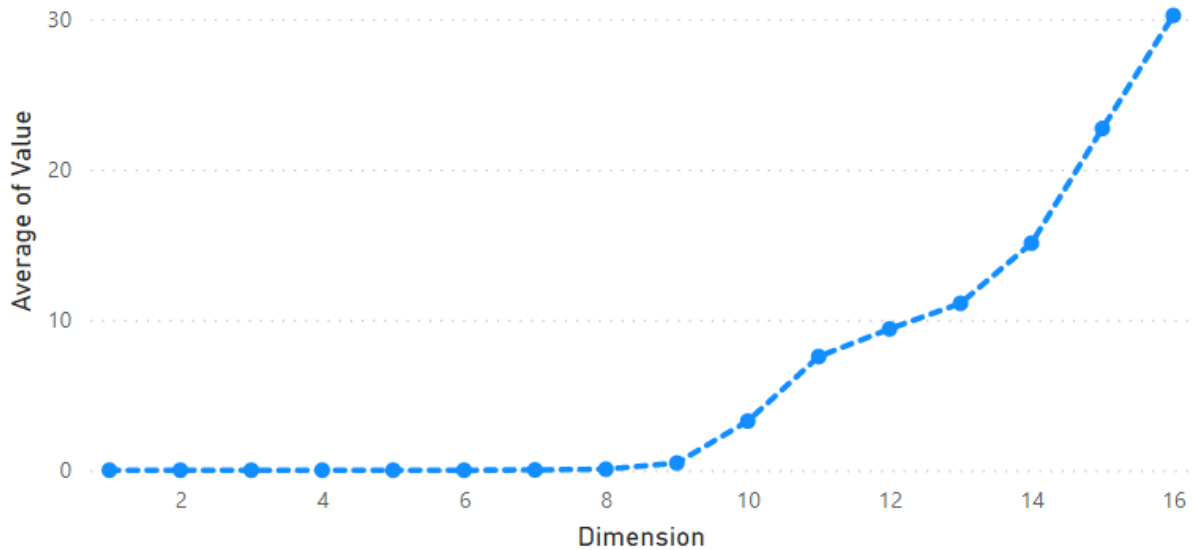
(a) Region of interest



(b) Region of interest correlation score

Source: Own authorship (2022).

Figure 8 – Example of a vehicle's slope vector post dimensionality reduction
Slope data example (16)



Source: Own authorship (2022).

dimensions. For every dimension, or range of speed, there are respective values of percentage of time spent on that speed range, considered all the vehicle log.

Table 2 – Speed dimensions

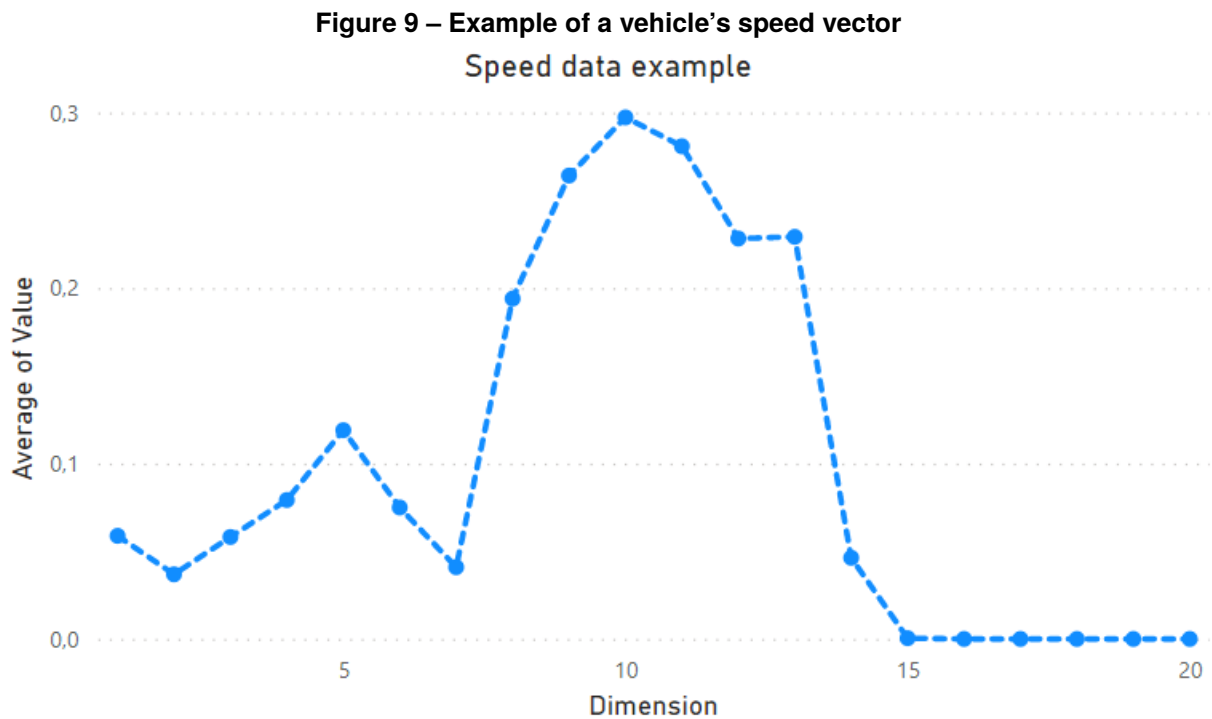
| | Idle | Low speed | | | | | | Medium speed | | | | | | High speed | | | | Very high speed | | | |
|-----------|------|-----------|---|---|---|---|---|--------------|---|----|----|----|----|------------|----|----|----|-----------------|----|----|--|
| Dimension | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | |
| | | | | | | | | | | | | | | | | | | | | | |

Source: Own authorship (2022).

The first job is to clean the data and eliminate rows with empty values. After, the data is transformed into a tabular shape, where each row is a vehicle and each column a feature. To better comprehend the data, each feature value v_i is then transformed to a percentage of the total driving distance, following the Equation 21, where D is the number of dimensions. In the end, for every vehicle vector, each of the attributes is represented as its percentage of the total travelled distance; that is, each data point is turned into a weight representation. That is the input data for the problem. But, before start clustering, a normalization stage must be done, scaling individual samples to have a unit norm to make sure the data looks like standard normally distributed data, which helps the clustering algorithm performance (HAN; KAMBER; PEI, 2012b,a). The normalization is made with Min-max, presented in Equation 22.

As a result, a complete normalized data set is ready to be clusterized. No processing of data reduction, feature extraction or selection was made since the most

complete and precise data set is needed to predict the best clustering for each individual in the group. In Figure 9, it is shown an example of a final normalized vehicle row, where in the X-axis are the dimensions, each one an interval of speed, and in the Y-axis are the percentage of traveled time in each range. This vehicle presented, for example, has the mode in the dimension 10 with around 0.3 of value (already normalized).



Source: Own authorship (2022).

At the clustering stage, the data is clusterized with K clusters varying from 2 to 10, so that an Elbow Curve can be used to evaluate the algorithms used, K-Means, FCM and SOM. Those and other results are presented in the next chapter.

3.2.3 Gross Combination Weight (GCW)

The third data set analyzed refers to the logged data of the combined gross weight carried by the vehicle. Different Gross Combination Weight (GCW) values may vary by the truck’s own weight and the weight of the load, which can be grains, solids, manufactured products, foods, etc. With that data set clustered, we are able to see closely which trucks work with the same load condition.

The data comes distributed in 29 ranges of weight, that start from $[0 T, 3.5 T]$, in dimension 1, and end in $[200 T, \infty]$, in dimension 29, as presented in Table 3. Each

dimension will have values of distance traveled in each referred GCW range, given the vehicle's travel distance. The region from 1st to 9th range has low weights such that, most of the time, it can be considered that the values in these positions refer to empty loads, that is, the weight of the truck only. Dimensions 10 to 12 are considered low weight ranges. Dimensions 13 to 17 have ranges of medium weights, while 18 to 23 are considered high weights. Dimensions 23 to 29 have been considered very high weight. This initial classification is only an initial exploration step and should not be considered as ground truth; even more, these are hard-thresholded limits that may differ for different applications, although they are the general case scenario.

Table 3 – GCW dimensions

| Empty | | | | | | | | | Low |
|-------|----|--------|-----------|----|----|----|------|----|-----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Low | | Medium | | | | | High | | |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| High | | | Very high | | | | | | - |
| 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | - |

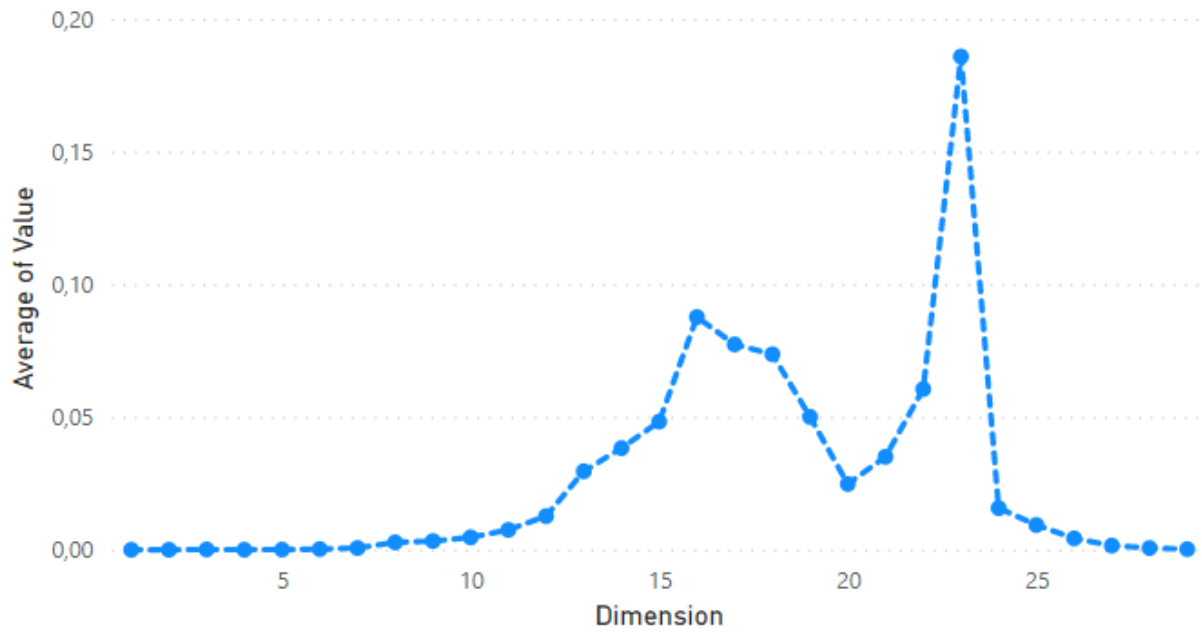
Source: Own authorship (2022).

The first job is to clean the data and eliminate rows with empty values. After, the data is transformed into a tabular shape, where each row is a vehicle and each column a feature. To better comprehend the data, each feature value v_i is transformed to a percentage of the total driving distance, following the Equation 21, where D is the number of dimensions. In the end, for every vehicle vector, each of the attributes is represented as its percentage of the total travelled distance; that is, each data point is turned into a weight representation. That is the input data for the problem. But, before start clustering, a normalization stage must be done, scaling individual samples to have a unit norm to make sure the data looks like standard normally distributed data, which helps the clustering algorithm performance (HAN; KAMBER; PEI, 2012b,a). The normalization is made with Min-max, presented in Equation 22.

As a result, a complete normalized data set is ready to be clusterized. No processing of data reduction, feature extraction or selection was made since the most complete and precise data set is needed to predict the best clustering for each individual in the group. In the figure 10, it is shown an example of a final normalized vehicle row, where in the X-axis are the dimensions, each one an interval of GCW, and in the Y-axis are the percentage of traveled distance in each range. This vehicle presented,

for example, has the mode in the dimension 23 with a value close to 0.18 (already normalized).

Figure 10 – Example of a vehicle's GCW vector
GCW data example



Source: Own authorship (2022).

At the clustering stage, the data is clusterized with K clusters varying from 2 to 10, so that an Elbow Curve can be used to evaluate the algorithms used, which are: K-Means, FCM and SOM. Those and other results are presented in the following chapter.

4 RESULTS AND ANALYSIS

This chapter presents the results of algorithms, and further analyses are made. The results are presented following the sequence: SSW Elbow Curve, SSB Elbow Curve, Silhouette, k-Fold Homogeneity, final metrics comparison, clusters average for each algorithm, and clusters average comparison. For every algorithm presented in the previous chapter, it is looked for the best number of cluster K by varying it from [2, 10] and analyzing the evaluation metrics.

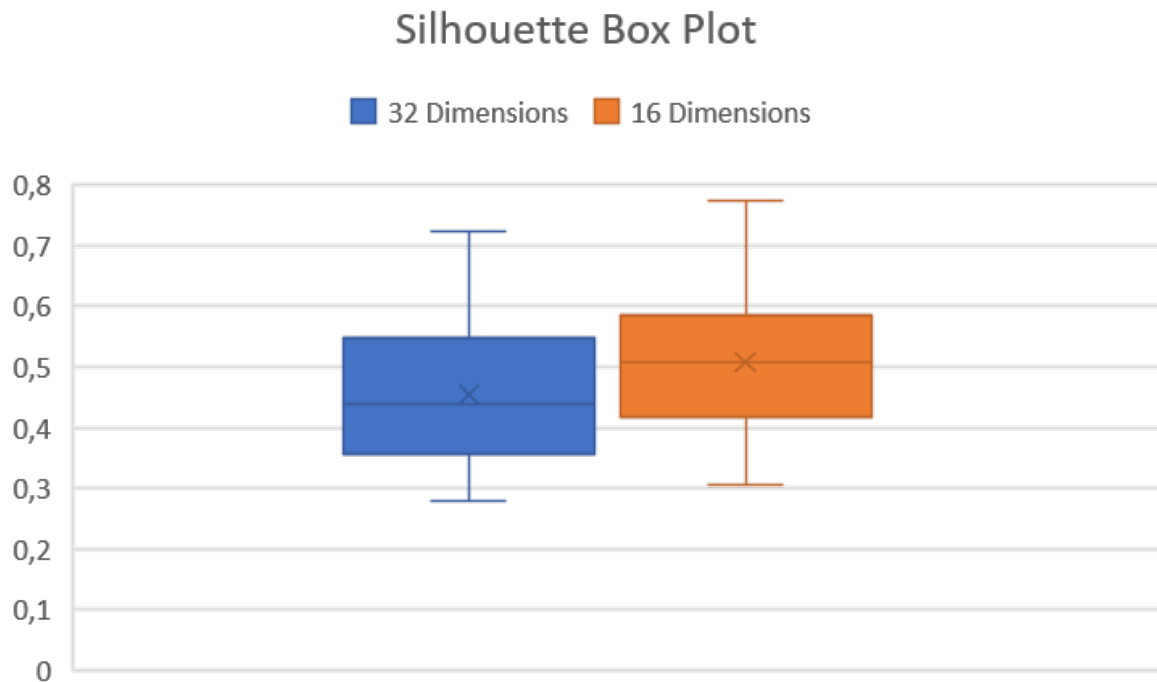
4.1 Slope

For the slope data set, the results are compared with those obtained by the classification method called GTA, which is built with extensive engineering knowledge analyzing slope conditions, that is, with no machine learning method. This classification clusters the data set into four groups: Flat, Predominantly Flat, Hilly, and Very Hilly. We then try to relate these groups from the empirical classification with the clusters found by the clustering algorithms and evaluate.

The results presented in the following sections are the ones got with the dimensionality reduction proposed, since it has returned better results. The SSW and SSB metrics can not be compared between data sets of different dimensional sizes, but the Silhouette, which is actually the most important one to use as reference when comparing validation performance, we are able to use. So, Graph 1 presents the silhouette result for all the algorithms proposed, comparing the 16 dimension and 32 dimension data sets. With the box plot, the median is represented as the center line of the box, while the mean is the cross close to it. The top of the box is the upper quartile, the bottom is the lower quartile, and outside of the box, in the top and bottom, are the respective maximum and minimum of the data. By analyzing the data with this type of graphic, we are able to see not only the mean, maximum and minimum, but also the deviation of the data. For that, the closer the quartiles are from the median, less deviation there is in the data.

Through the box plot, we can see that reducing the number of dimensions does not degenerate the result and even give a better result. The mean silhouette score increased from 0.45 to 0.51, while still increasing the maximum, minimum, and reducing

Graph 1 – Silhouette comparison between original 32 dimension and transformed 16 dimension data sets



Source: Own authorship (2022).

the quartiles sizes a bit.

4.1.1 Performance evaluation

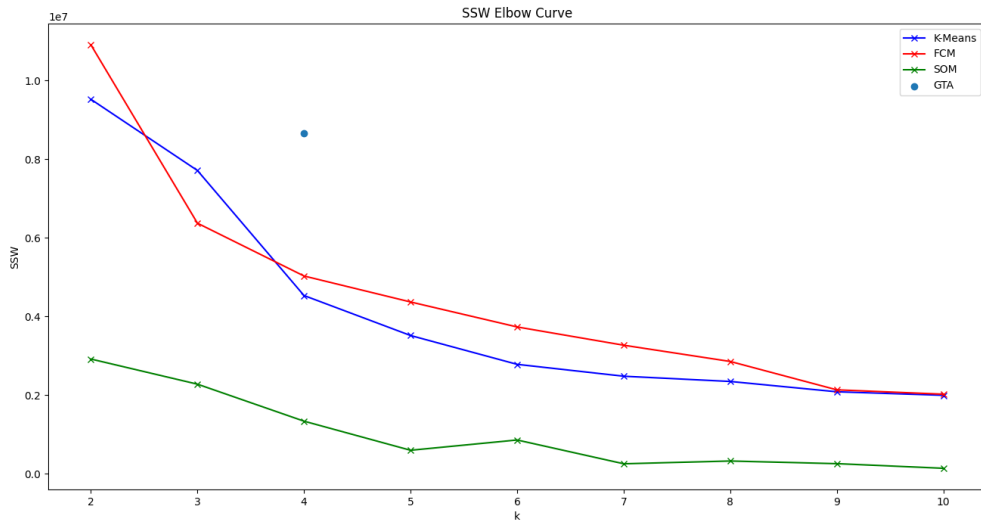
Some metrics were used to evaluate each algorithm's performance to choose the best number of clusters that describe the slope data set. To have some greater insight over each metric, they are presented as an elbow curve, which is, changing the number of clusters, in this case, from 2 clusters to 10 clusters. The results presented below are the average got after 10 iterations.

4.1.1.1 SSW

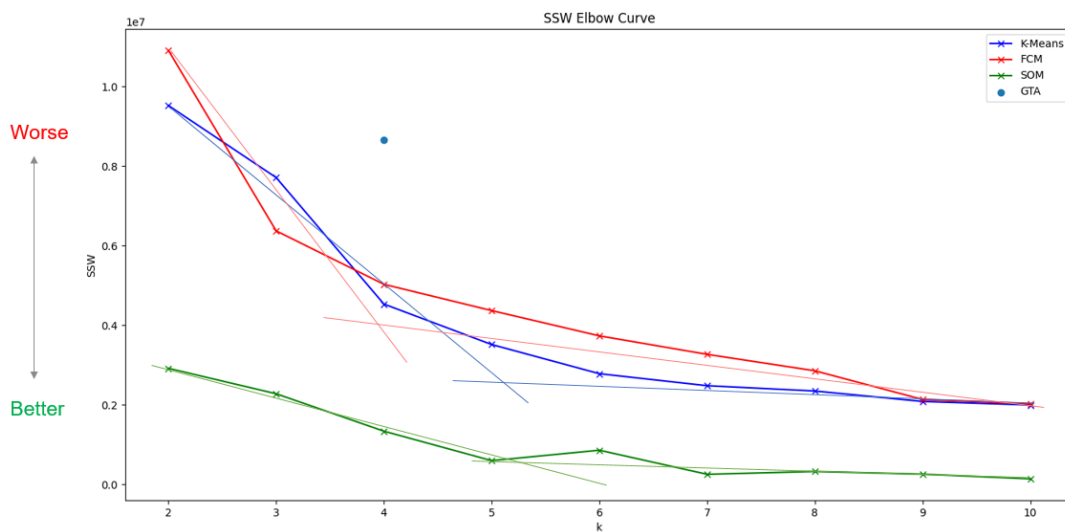
For the first metric, the SSW (Sum of Squares Within Clusters) that is presented in the Graph 2, the lowest value is supposed to be the best result, which is the tightest cluster. However, on this analysis, it is sought for the inflection point. This point means that increasing the number of clusters does not bring huge gains on the current metric further from this point. Crossing that information with the other metrics will give a better

view of the best number of clusters for the problem.

Graph 2 – SSW Slope Elbow Curve



(a) SSW Elbow Curve



(b) Adjusted SSW Elbow Point

Source: Own authorship (2022).

The SOM methodology (green line) returned the best result for every cluster number K compared with the other algorithms, with the K-Means in second place. FCM (red line) has a close result to K-Means overall, but the GTA classification (cyan dot) – which only classifies with $K = 4$ – is far from the machine learning methodologies.

Sometimes it is difficult to get a clear elbow point. To facilitate that, on Graph 2(b), two trend lines help indicate this value, where the inflection point is the cross of the two lines. The first trend line is a straight line drawn from the first point of the elbow curve, following the trend for at least 3 periods. The second trend line comes in the opposite way, from the last point in the elbow curve, following backwards the elbow curve for at least 3 periods, until it crosses the first line. With that we can locate the virtual elbow point, the point of inflection of the tendency. Through that method, for K-Means and SOM, the elbow point could be considered $K = 5$, as it is closer to that K . For FCM the elbow point is $K = 4$.

4.1.1.2 SSB

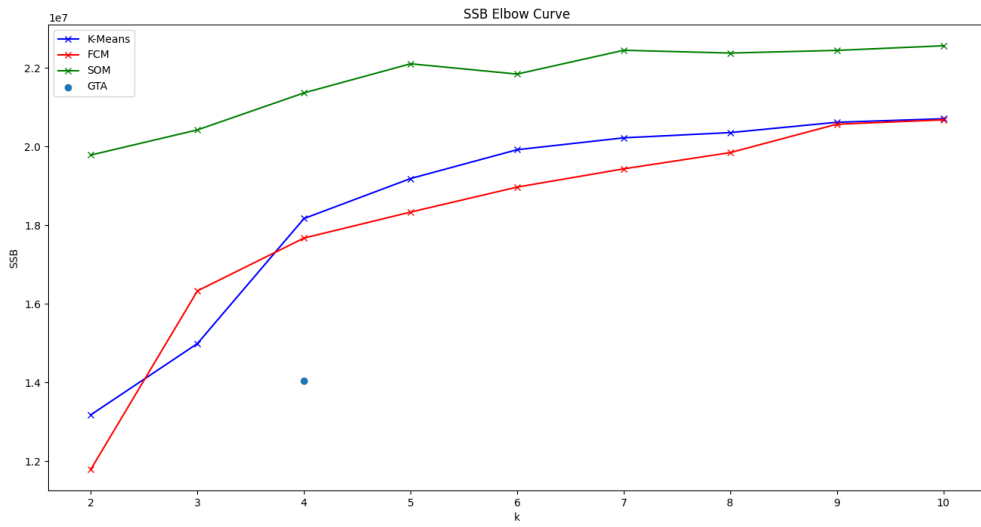
The second evaluation metric is the SSB (Sum of Squares Between Clusters), which measures how far clusters are from each other, so the highest value is the best result. A good SSB value says that the clusters are well defined. Just as in the SSW analysis, two trend lines help indicating which is the best number of clusters. The results are shown in Graph 3.

SOM again returned the best result for every cluster number K compared with the other algorithms. The FCM algorithm has often a close result to K-Means, but the GTA classification – which only classifies with $K = 4$ – is another time far from the machine learning methodologies. Analyzing the elbow point with the help of trend lines, the inflection point for K-Means and SOM can be considered $K = 5$, while for FCM the elbow point is $K = 4$.

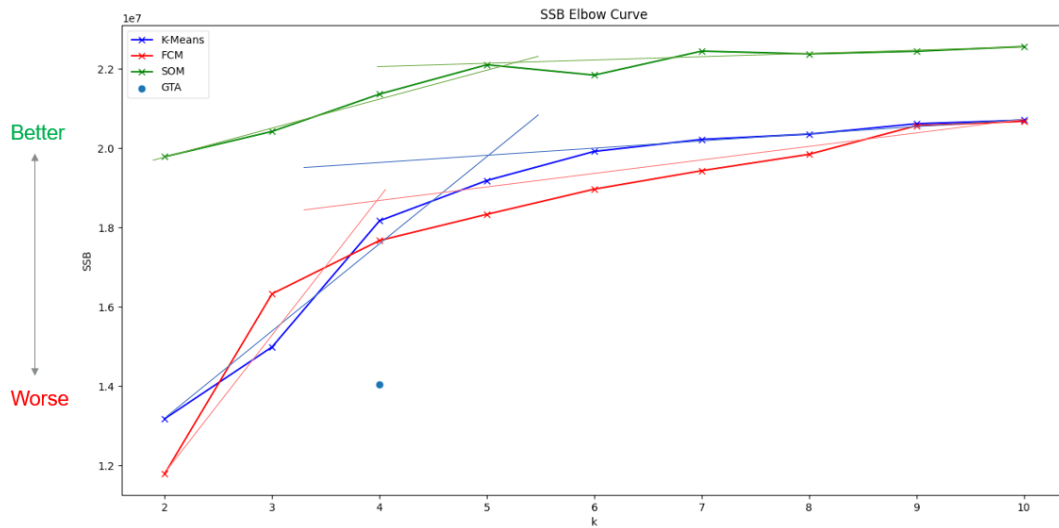
4.1.1.3 Silhouette

The third and most important metric for this study, Silhouette Index (SI), is an internal validation measurement. It is a normalized score, from -1.0 to 1.0 , which the closer to 1, the better the data is classified. Figure 11 illustrates the results considering the average SI score of the data set for each number of clusters K . For this index, SI values from 0.5 to 1.0 are considered a good result (green region on the figure), from 0.2 to 0.5 a fair result (yellow region), and from -1.0 to 0.2 a poor clustering (red region) (KAUFMAN; ROUSSEEUW, 2009).

Graph 3 – SSB Slope Elbow Curve



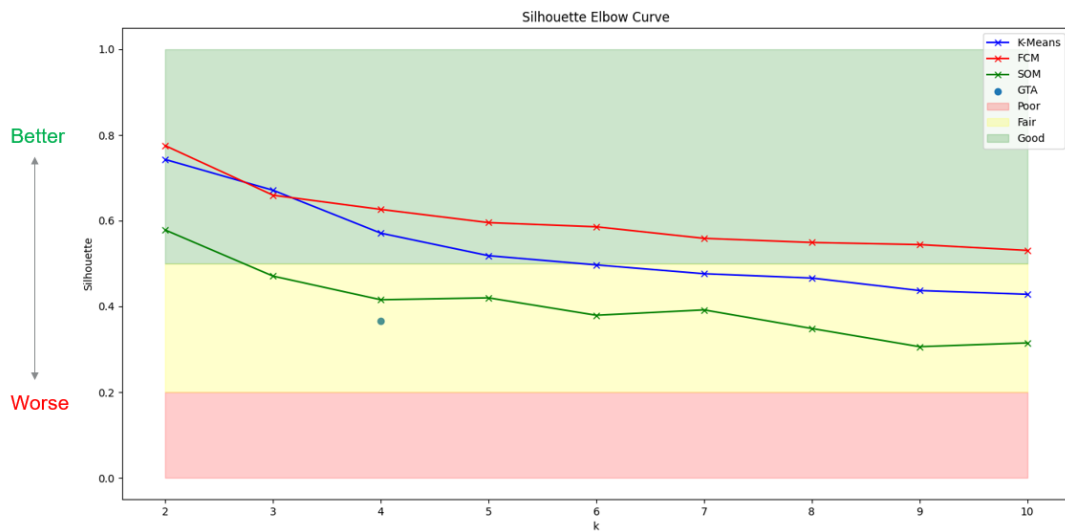
(a) SSB Elbow Curve



(b) Adjusted SSB Elbow Point

Source: Own authorship (2022).

Figure 11 – SI Slope Elbow Curve



Source: Own authorship (2022).

When comparing all the algorithms, FCM returned the best result for most cluster numbers except when $K = 3$, which in case K-Means returned a better result. FCM for every K has SI above 0.5, which is considered a good result; K-Means algorithm results from $K = 2$ to $K = 6$ are good results, higher than 0.5, but from $K = 7$ to $K = 10$ the SI value fell to a consider fair result, from 0.2 to 0.5; and for GTA, with $K = 4$, the SI value lies between 0.2 and 0.5, standing for what is considered a fair clustering. SOM brings good results only when $K = 2$ returns a fair clustering.

The Silhouette Elbow Curve is not analyzed looking for an elbow point, but now, crossing the information with the previous metrics, it is easier to pick which should be the best number of clusters.

For FCM, SSW and SSB showed that a good clustering should have at least 4 clusters. Using the SI to validate that with $K = 4$ or higher, the algorithm can return a good clustering in terms of validity; the best K for this method is $K = 4$. That way, it is ensured to have good SSB (separation) and SSW (compactness) values while choosing the number of clusters that can better classify the data set.

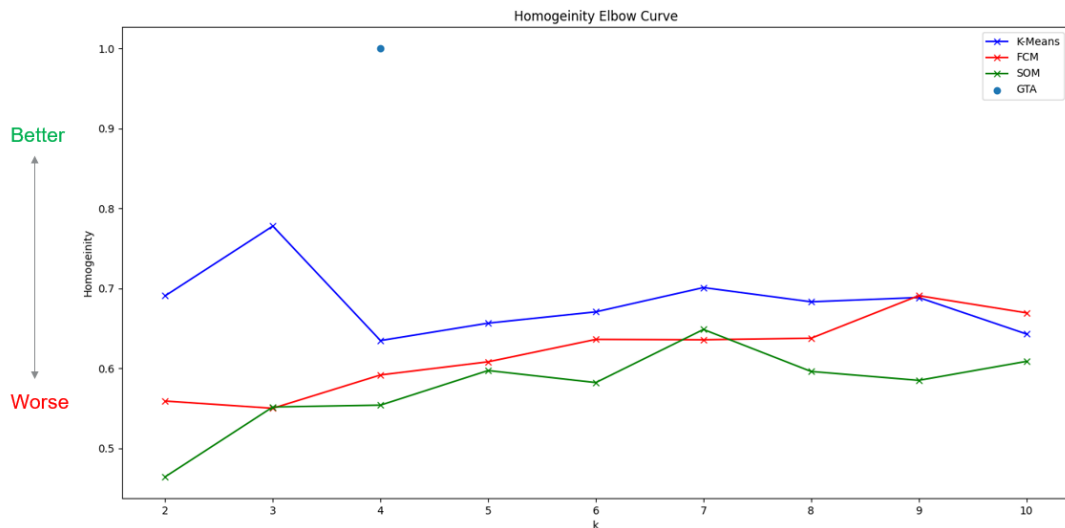
Analyzing SOM and K-Means clustering, SSW and SSB presented that a good clustering should have at least 5 clusters. Validating with SI, the best K for these methods is $K = 5$ since it returns the best Silhouette result with the least, but sufficiently better,

SSW and SSB.

4.1.1.4 Homogeneity

The last metric, k-Fold Homogeneity, is an algorithm performance consistency validation. In the same way as the SI, it is a scaled score from 0 to 1.0, where 1.0 is a perfectly homogeneous clustering. The results for each cluster are presented in figure 12.

Figure 12 – k-Fold Homogeneity Slope Elbow Curve



Source: Own authorship (2022).

The K-Means algorithm shows more consistency than the others when $K = 2$ to 8 . FCM has more homogeneity when K is 9 and 10 . For the GTA classification method, as it is a classification that follows the rules, it will return the same result for every iteration, considering that this is an unsupervised clustering. Hence, using a k-Fold method, each result is compared with another that will be the same result, so the k-Fold Homogeneity result is always 1. For that, GTA has the highest consistency when $K = 4$.

In the region that is understood to have the best K number – 4 and 5 –, K-Means has highest consistency, where, overall, the algorithms with more than 0.5 are considered to have a good consistency.

4.1.1.5 Coefficient of variation

In the Table 4, we can see how our algorithms have performed through 10 iterations each according to every metric for every number of clusters from 2 to 10. With the CV calculation, we can see the mean variation for the performance measures.

In the K-Means algorithm, the CV values are very low, except for the HS, which has values from 3.25% to 13.77%, which means that the grade for this score may vary a bit more on a different initialization, while the others will remain very close to be the same score always since the CV is close to zero. The K-Means problem of the randomness of initialization that would affect the quality of the solution could be mitigated once, looking at the SSW, SSB, and SI metrics, we had close to zero variation in the performance results. This was done by executing 20 times with different centroid seeds in each iteration and selecting the best solution in terms of distance between each data point to its centroid. The algorithm, however, suffers a bit on the consistency of the HS metric, which shows that its prediction error may vary but will not impact the performance quality.

For the FCM algorithm, the results for SSW, SSB, and SI are even closer to zero, if not it. The CV is also very low for HS, varying from 0.95% to 2.30%. Therefore, it is seen that FCM is less dependent on the initialization than K-Means and has more performance consistency.

Analyzing the SOM CV results, we see that this algorithm suffered more in performance consistency. Especially, the SSW CVs have values from 12.20% to 80.45%, while SSB have values from 0.16% to 2.83%, SI from 3.10% to 7.61% and HS from 1.73% to 16.68%. SOM algorithm has more consistency problems with compactness (SSW), mainly when it has 5 clusters or higher as the target, than the other algorithms.

In general, Table 4 presented an analysis of the consistency of algorithms for quality measurements across many startups. FCM showed more consistency than the others, less dependent on the initialization aspect.

4.1.2 Summary of results

Having a closer look at the results presented, Figure 13 compares each metric evaluation value for GTA, K-Means, FCM and SOM, where it was understood to be the best number of clusters.

Table 4 – Slope Coefficient Variation table

| K | K-Means | | | | FCM | | | | SOM | | | |
|----|---------|-------|-------|--------|-------|-------|-------|-------|--------|-------|-------|--------|
| | SSW | SSB | SI | HS | SSW | SSB | SI | HS | SSW | SSB | SI | HS |
| 2 | 0.05% | 0.03% | 0.03% | 8.95% | 0.00% | 0.00% | 0.00% | 1.58% | 19.24% | 2.83% | 4.37% | 16.68% |
| 3 | 0.01% | 0.01% | 0.02% | 10.17% | 0.00% | 0.00% | 0.00% | 1.64% | 14.82% | 1.65% | 3.34% | 8.16% |
| 4 | 0.10% | 0.03% | 0.08% | 13.77% | 0.00% | 0.00% | 0.00% | 2.30% | 12.20% | 0.76% | 4.05% | 6.14% |
| 5 | 0.08% | 0.01% | 0.97% | 11.05% | 0.03% | 0.66% | 0.00% | 1.40% | 80.45% | 2.16% | 3.83% | 2.69% |
| 6 | 0.12% | 0.02% | 0.08% | 11.54% | 0.00% | 0.00% | 0.00% | 1.07% | 43.44% | 1.70% | 5.21% | 3.02% |
| 7 | 0.05% | 0.63% | 0.02% | 11.59% | 0.00% | 0.00% | 0.00% | 0.94% | 46.16% | 0.52% | 3.10% | 2.66% |
| 8 | 0.26% | 0.03% | 0.06% | 5.39% | 0.01% | 0.20% | 0.00% | 1.12% | 54.65% | 0.78% | 5.28% | 3.07% |
| 9 | 0.06% | 0.56% | 0.04% | 9.72% | 0.05% | 0.52% | 0.00% | 1.02% | 35.09% | 0.40% | 6.04% | 1.73% |
| 10 | 0.05% | 0.47% | 0.06% | 3.25% | 0.01% | 5.95% | 0.00% | 0.95% | 27.16% | 0.16% | 7.61% | 4.37% |

Source: Own authorship (2022).

For SSW, presented on a scale of red to green from worst to best, SOM with 5 clusters has presented the best result. For SSB, presented with the same color scale, SOM with 5 clusters had the best result also. Regarding k-Fold Homogeneity, presented on a scale of white to green from worst to best, GTA classification has the highest homogeneity consistency. The lack of reproducibility by the ML methods, seen with that Cross-Validation Homogeneity Score, means that this data set should be retrained more frequently, according to the increase in the number of samples.

Moreover, for SI, following the previous color rule presented in Figure 11, the FCM algorithm had the best results, but K-means also has a "good" result, above 0.5, while GTA and SOM have considered "fair" result. The best SI score is from FCM with 4 clusters.

Figure 13 – Slope Detailed metrics comparison

| | GTA | FCM (K = 4) | K-Means (K = 5) | SOM (K = 5) |
|----------------------|----------|-------------|-----------------|-------------|
| SSW (↓) | 8,66E+06 | 5,03E+06 | 3,51E+06 | 5,94E+05 |
| SSB (↑) | 1,40E+07 | 1,77E+07 | 1,92E+07 | 2,21E+07 |
| Silhouette (1,00 ↑) | 0,37 | 0,63 | 0,52 | 0,42 |
| Homogeneity (1,00 ↑) | 1,00 | 0,59 | 0,66 | 0,60 |

Source: Own authorship (2022).

The SOM methodology has presented the most compact and distinct groups and GTA the least. First, there should be a reminder that for FCM's SSW, SSB, and k-Fold Homogeneity, it is considered only the cluster with the highest membership value, that is, the Hard C-Means way, since there is no specific calculation that can include the fuzzy aspect the same way there is for SI, presented on equation 16 of the previous chapter. It undoubtedly impacts the grade obtained for each index. Despite that disclaimer, FCM algorithm had the best clustering performance when looking at the SI results, which present the impact of the soft-clustering consideration on this data set. We can interpret

that this data set has a lot of overlapping situations.

Before analyzing the centroids of the clusters, it is necessary to look into the silhouette of each cluster for all algorithms, presented in Table 4.1.2. For GTA, the Flat cluster has the highest SI score with 0.84, but the lowest, PFlat, has -0.01, which is a lousy classification score. The FCM cluster with the highest SI score is the Flat, with 0.78, but VHilly has the lowest with 0.46, which is a "fair" score. K-Means, however, has a much more equal score between clusters: the highest value is 0.53 for K4 and K5, and the lowest is 0.51 for K2, all considered "good" classification. The highest cluster of SOM has a SI of 0.54, and the lowest is 0.25. Even though FCM has the highest SI score, the K-Means' SI score has more consistency between clusters, which may impact the classification of some samples.

Table 5 – Average silhouette of Slope clusters

| GTA | Average of SILHOUETTE | FCM | Average of SILHOUETTE |
|---------|-----------------------|----------|-----------------------|
| FLAT | 0.84 | F-FLAT | 0.78 |
| HILLY | 0.43 | F-HILLY | 0.69 |
| PFLAT | -0.01 | F-PFLAT | 0.58 |
| VHILLY | 0.36 | F-VHILLY | 0.46 |
| K-MEANS | Average of SILHOUETTE | SOM | Average of SILHOUETTE |
| K1 | 0.52 | S1 | 0.33 |
| K2 | 0.51 | S2 | 0.54 |
| K3 | 0.52 | S3 | 0.50 |
| K4 | 0.53 | S4 | 0.43 |
| K5 | 0.53 | S5 | 0.25 |

Source: Own authorship (2022).

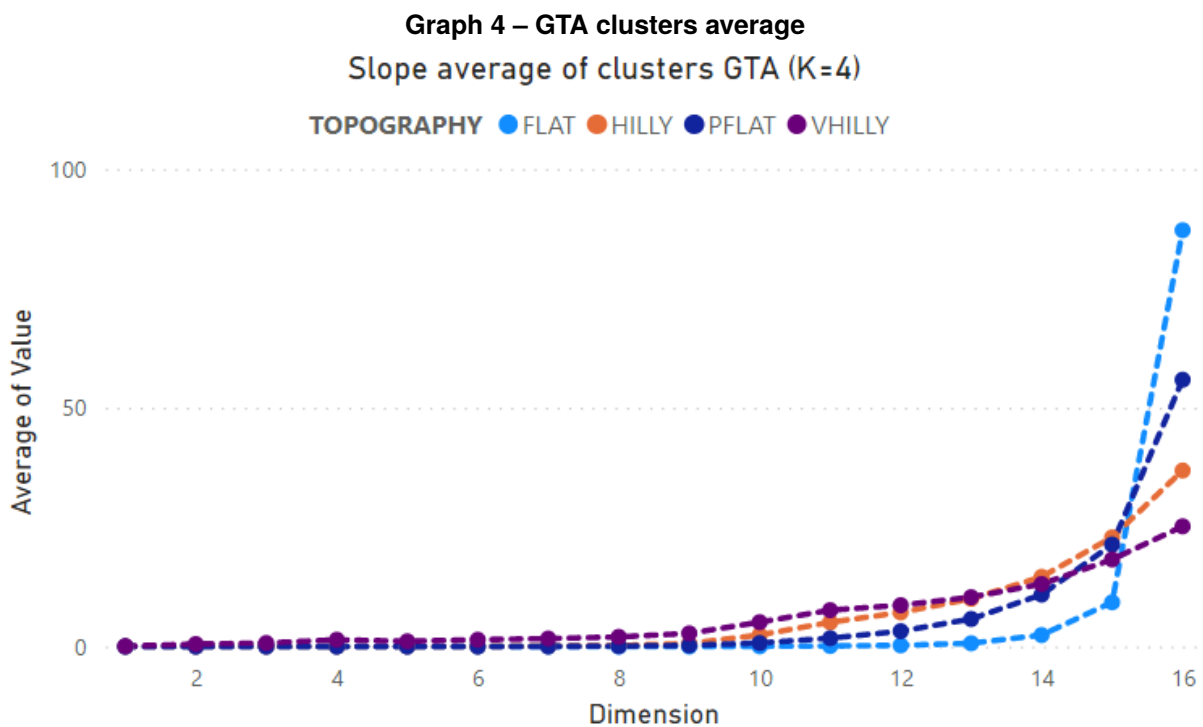
4.1.3 Centroids of the clusters

To see how the clusters' shapes are, a line graph shows the center of clusters, represented as the mean value of members within the cluster for each dimension. In this section, the results for 4 and 5 clusters will be presented since they showed to be the best numbers of clusters.

For methods that have 4 clusters, we used the same labels as from the GTA classification – Flat, Predominantly Flat (named "PFlat"), Hilly, and Very Hilly (named "VHilly") – but with a prefix "K" for K-Means, "F" for FCM and "S" for SOM to differentiate them. For methods presented with 5 clusters, it was not given any particular label, just numbered from 1 to 5.

Graph 4 presents the clusters' centroids of the GTA classification method. In the graph, the Y-axis is the percentage of time spent on the respective slope interval presented on the X-axis.

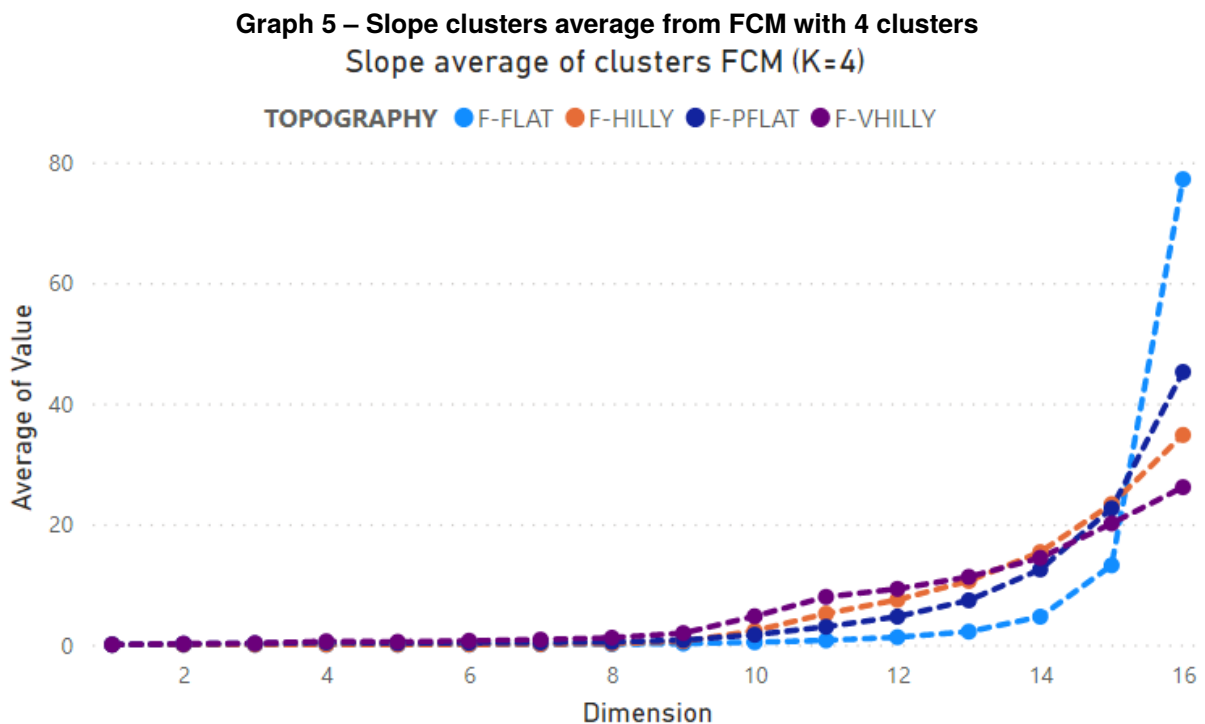
The first cluster from top to bottom is the Flat cluster, represented by the cyan color. It restricts data points that have a high presence on dimensions 14 to 16, that are low inclinations, with an average peak of above 85% of a lifetime spent on the 16th dimension, and never passed dimensions lower than the 9th, which are high inclinations. Then, in the blue color, the Predominantly Flat (PFlat) has an average peak close to 55% on dimension 16 and will not have values on dimension 3 or lower, which are very high slopes. The orange cluster is labeled Hilly, with an average peak close to 35% on dimension 16, and will eventually have few values in high and very high dimensions. The last, Very Hilly (VHilly) in purple color, with a peak around 25% on the flat dimension 16, will have lower values than the others on low and medium slopes but higher values on high and very high slopes regions.



Source: Own authorship (2022).

Graph 5 presents the average of clusters for FCM in the same color sequence and suffix as GTA, but with the prefix "F" for each label. The F-Flat cluster centroid has a dimension 16 value of 77%, lower than GTA's. For the Predominantly Flat cluster, F-PFlat, the peak on the 16th dimension is 45%, lower than GTA's. The 16th dimension

value for F-Hilly is 35%, close to GTA's Hilly. For F-VHilly, which has a flat slope peak of 26%, which is also very close to the GTA VHilly.

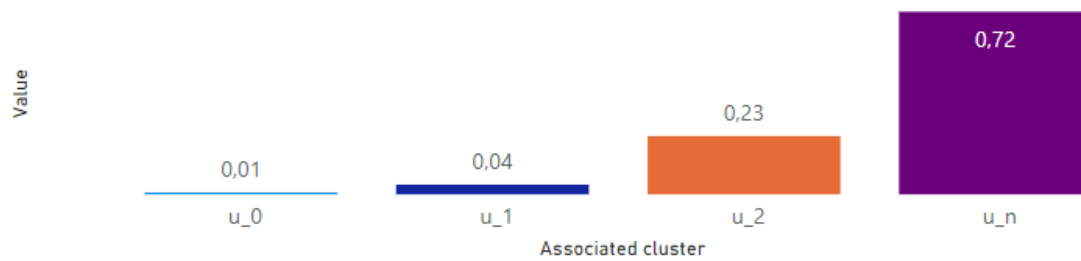


Source: Own authorship (2022).

However, Graph 5 presents the average based only on the Hard C-Means part of the method, considering only the cluster of highest membership grade for every vehicle. With the membership grades, we can have deeper information on the clustering of a sample, as presented in Graph 6, that has the membership values for the FCM clustering of the vehicle presented previously in Graph 8. We see that this vehicle has 72% on belonging to cluster F-VHilly, on purple, but also 23% of membership value in cluster Hilly (orange), 4% in PFlat (blue), and 1% in Flat (cyan). Since this data set has more than 3 dimensions and cannot be spatially plotted, with membership values, in an overlapping data set, we can detect outliers and to which clusters a data point is closest.

Graph 7 shows the average of the clusters found by K-Means, presented with the same color sequence – as GTA's clusters. The first cluster, ordering from flattest to steepest, is the K1 with a peak value on the 16th of 78%. The second cluster, K2, has a peak of 47% on dimension 16, while K3 has a value of 35% on that same slope range. K4 presented a value of 26% on that initial flat position. The new fifth cluster is a little bit different from the others, with a high presence in the very hilly region, dimensions 1 to 7, and the lowest of all clusters in the flat dimension 16 with 18%. This cluster could be

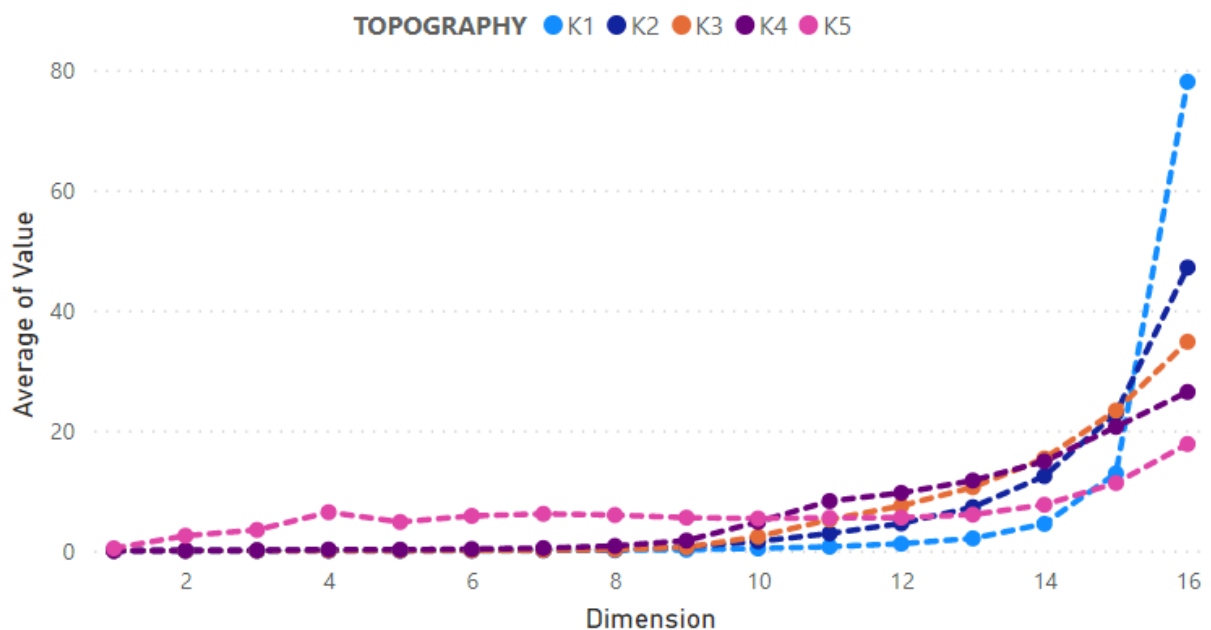
**Graph 6 – Example of membership grades of a vehicle in Slope data set
FCM membership degree**



Source: Own authorship (2022).

a cluster only of vehicles working at very high elevation jobs, such as mine extraction vehicles.

**Graph 7 – Clusters average from K-Means with 5 clusters
Slope average of clusters K-Means (K=5)**

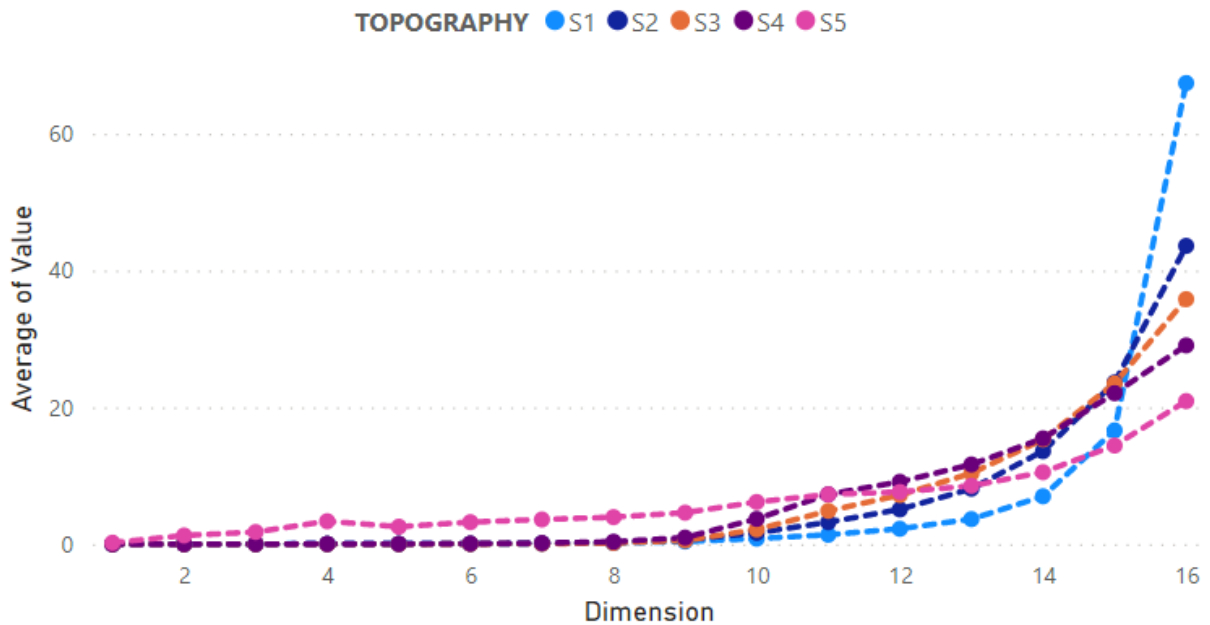


Source: Own authorship (2022).

Graph 8 shows the centroid of the clusters found by SOM, presented in the same color sequence. From flattest to steepest, the first cluster is the S1 with a peak value on the 16th of 67%. The second cluster, S2, has a peak of 43% on dimension 16, while S3 has a value of 36% on that same slope range. S4 presented a value of 29% on that initial flat position. The new fifth cluster, just as K-means', has a high presence in the very hilly region, dimensions 1 to 7, and the lowest of all clusters in the flat dimension 16 with 21%, tracking those vehicles working at very high elevation jobs only.

For all algorithms, from the first to the fourth cluster, for dimensions 13 to 1,

Graph 8 – Slope clusters average from SOM with 5 clusters
Slope average of clusters SOM (K=5)



Source: Own authorship (2022).

they reverse the order of values seen in dimension 16: now, the highest value for each dimension is from the steepest cluster, while the flattest has the lowest value. This fact complements the initial correlation analysis, where we saw a negative correlation for these points, mainly for dimensions 13 to 10, with the 16th dimension. For those algorithms with a fifth cluster, that last one did not follow this rule, but it was noticed that these clusters are a part of the others as a cluster of exceptional, very hilly situations.

Comparing similar clusters from different algorithms, the Flat for GTA is the flattest of all, with a value of 87% on dimension 16, followed by K-means' K1 with 78%, FCM's F-Flat with 77%, and SOM's S1 with 67%. For PFlat, GTA has 55%, while the others are close to each other with 43% to 47% (K2, F-PFlat, S2). Comparing the GTA Hilly with K3, S3, and F-Hilly, all of them have a value close to 35% in dimension 16, just as the VHilly with K4, S4, and F-VHilly, which have around 26%. We can see that, except for GTA's first and second flattest clusters, all of the clusters found by the algorithms are close to each other in terms of the centroids.

4.1.4 Vehicle distribution in clusters

Analyzing all the samples as a whole, we show in Graph 9 how the data set is clustered for each algorithm.

For GTA in Graph 9(a), with 45.99%, HILLY had the most significant cluster and was 2.062.16% bigger than FLAT, which had the lowest number of vehicles with 3.02%. The HILLY cluster was followed by VHILLY, PFLAT, and FLAT in size. GTA's Flat is very restrictive since only a few vehicles could be classified as Flat, while the Hilly cluster comprises more than half the population. PFlat and VHilly are clusters of very compact size.

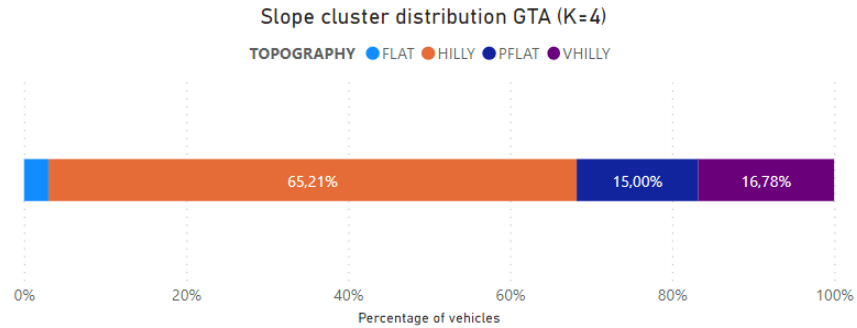
For FCM in Graph 9(b), with 40.74%, F-HILLY had the highest count of vehicles, followed by F-PFLAT, F-VHILLY, and F-FLAT. F-HILLY was 355.70% bigger than F-FLAT, which had the lowest number of samples, with 8.94% of the total. For FCM on these graphs, it was considered for each vehicle only the cluster with the highest membership degree for calculation purposes, transforming into the Hard C-Means result.

For K-Means in Graph 9(c), with 45.99%, K3 had the highest number of samples and was 1.482.98% more prominent than K5, which had the lowest count, with 2.91% of the total.

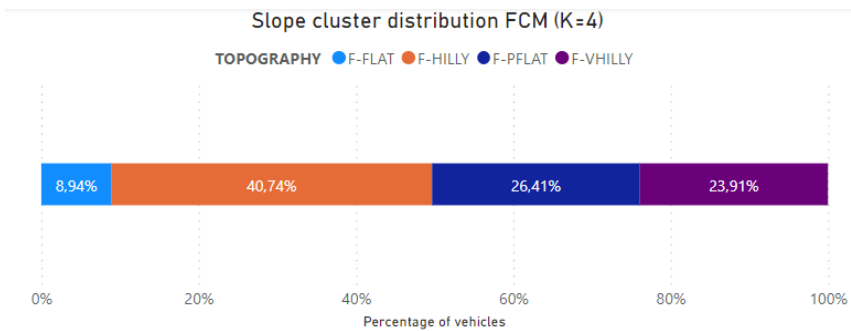
For SOM in Graph 9(d), with 30.27%, S3 had the highest number of vehicles and was 361.79% bigger than S5, which had the lowest count, with 6.56%.

Analyzing average cases and some outliers, we could see that the classifications made by the proposed methods generally make more sense with the reality than the GTA classification made by engineering experience. Reducing the size of the Hilly population and increasing the size of the others brought great benefits to the clustering and directly impacted the Silhouette grade of the clustering methods. The K5 and S5 cluster become a cluster of trucks that work mainly in mines and environments with more than 8% of inclination; Very Hilly (or K4 and S4) clusters would include only trucks that spend more time in the hilly and very hilly ranges, dimension 8 to 13, which are recognized as mountain highways; Hilly (K3 and S3) turns into an intermediate cluster from Very Hilly and Predominantly Flat; while Predominantly Flat is a cluster that describes an environment that is mainly flat but sometimes crosses mountain highways; and the Flat cluster has included mainly trucks that work inside cities or that work from one factory to another close one. Vehicles that work through mountain highways are vehicles that

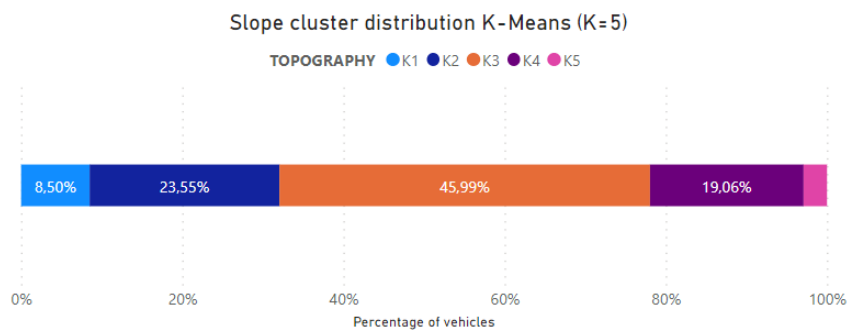
Graph 9 – Slope clusters data set distribution



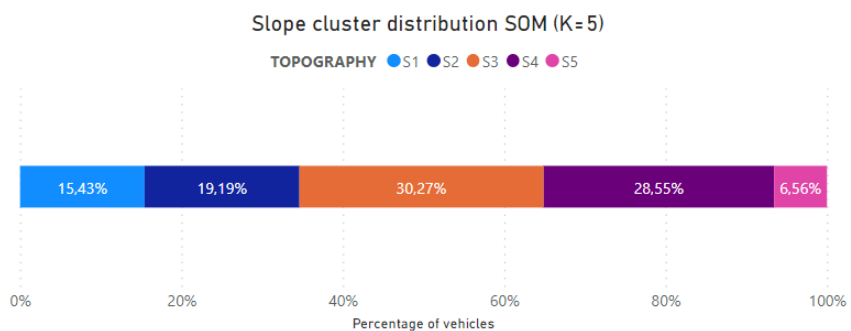
(a) Samples distribution in GTA clusters



(b) Samples distribution in FCM clusters



(c) Samples distribution in K-Means clusters



(d) Samples distribution in SOM clusters

Source: Own authorship (2022).

transport production between ports and cities. With the GTA classification, we did not have a clear definition since different applications were in the same cluster, as seen in Graph 9(a), where the Hilly cluster is massive and Flat, too restrictive.

4.2 Speed

We did not have a non-ML method for the speed data set, so all methods are compared using the elbow curve changing from $K = 2$ to $K = 10$.

4.2.1 Performance evaluation

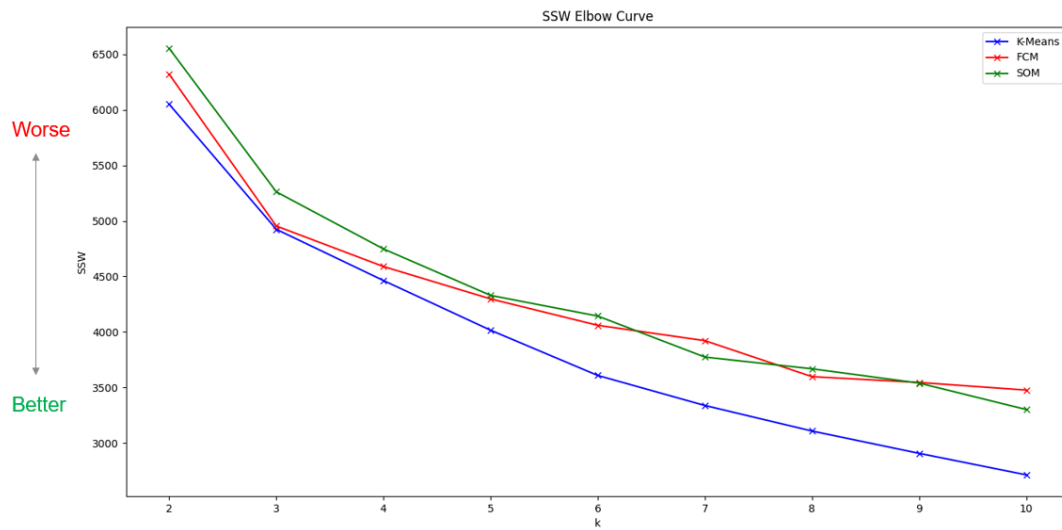
Some metrics were used to evaluate the algorithm's performance and choose the best number of clusters that describe the speed data set. SSW, SSB, SI score, and HS are presented, followed by their CVs and a summary of the evaluation metrics. The centroids and the vehicle distribution of the clusters found by the best algorithm for each method will be presented.

4.2.1.1 SSW

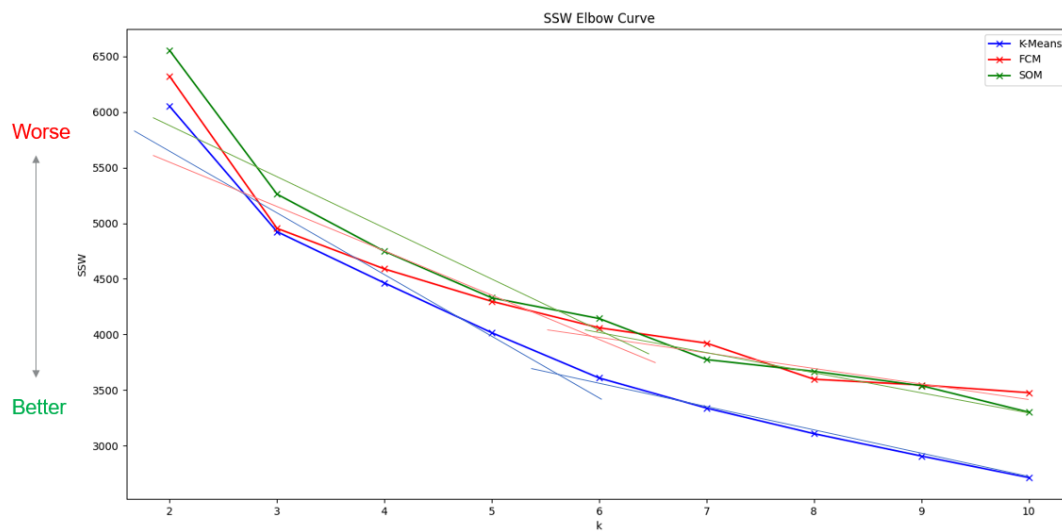
The SSW (Sum of Squares Within Clusters), presented in Graph 10, has the lowest value as the best result, representing the tightest cluster. In this analysis, it is sought for the inflection point, the point that, from it, increasing the number of clusters will no longer bring huge gains. Crossing that information with other metrics will give a better view of the best number of clusters for the problem.

The K-Means methodology (blue line) returned the best result for every cluster K compared with the other algorithms, followed mainly by the FCM. To get the elbow point, in Graph 10(b), two trend lines help indicate where the inflection point is across them. The first trend line is a straight line drawn from the first point of the elbow curve, following the trend for 5 periods. The second trend line comes oppositely, from the last point in the curve ($K = 10$), following backward the elbow curve for 5 periods until it crosses the first line. The increase from 3 to 5 periods, compared with the slope's elbow method, is because, in this case, the tendency change more smoothly and with 3 periods, the elbow point would be too far from the curve, losing the relationship with it.

Graph 10 – SSW Speed Elbow Curve



(a) SSW Elbow Curve



(b) Adjusted SSW Elbow Point

Source: Own authorship (2022).

With the virtual elbow point allocation, we see that the best number of clusters would be with at least $K = 6$.

4.2.1.2 SSB

The second evaluation metric is the SSB (Sum of Squares Between Clusters), which measures how far clusters are from each other, so the highest value is the best result. A good SSB value says that the clusters are well defined. As in the SSW analysis, two trend lines help indicate the best number of clusters. The results are shown in Graph 11.

K-Means again returned the best result for every cluster number K compared with the other algorithms, followed most of the time by FCM. Analyzing the elbow point with the help of trend lines, the inflection point for K-Means and SOM can be considered $K = 6$, while for FCM the elbow point is closer to $K = 5$.

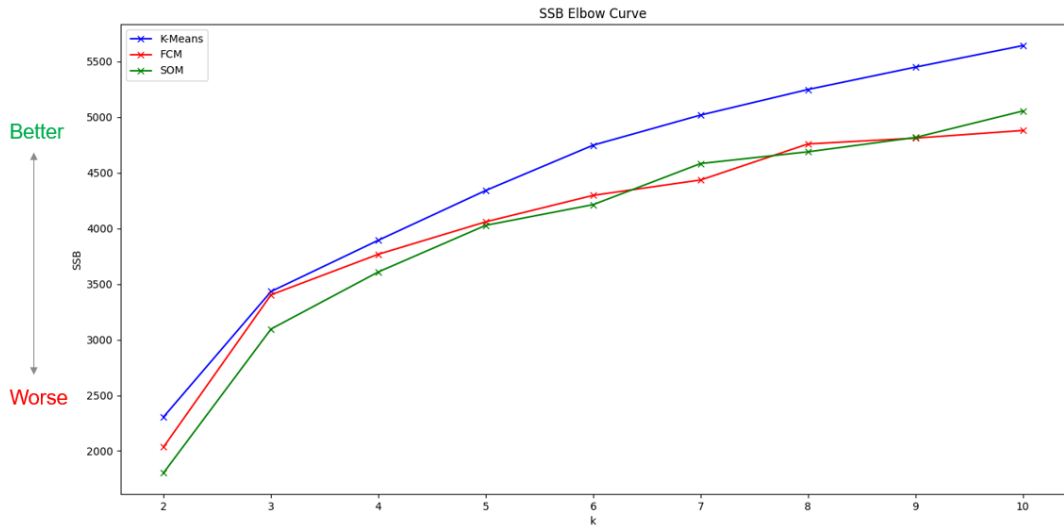
4.2.1.3 Silhouette

The third and most important metric for this study, Silhouette Index (SI), is an internal validation measurement. It is a normalized score, from -1.0 to 1.0 , which the closer to 1 , the better the data is classified. Graph 12 illustrates the results considering the average SI score of the data set for each number of clusters K . For this index, SI values from 0.5 to 1.0 are considered a good result (green region on the graph), from 0.2 to 0.5 a fair result (yellow region), and from -1.0 to 0.2 a poor clustering (red region) (KAUFMAN; ROUSSEEUW, 2009).

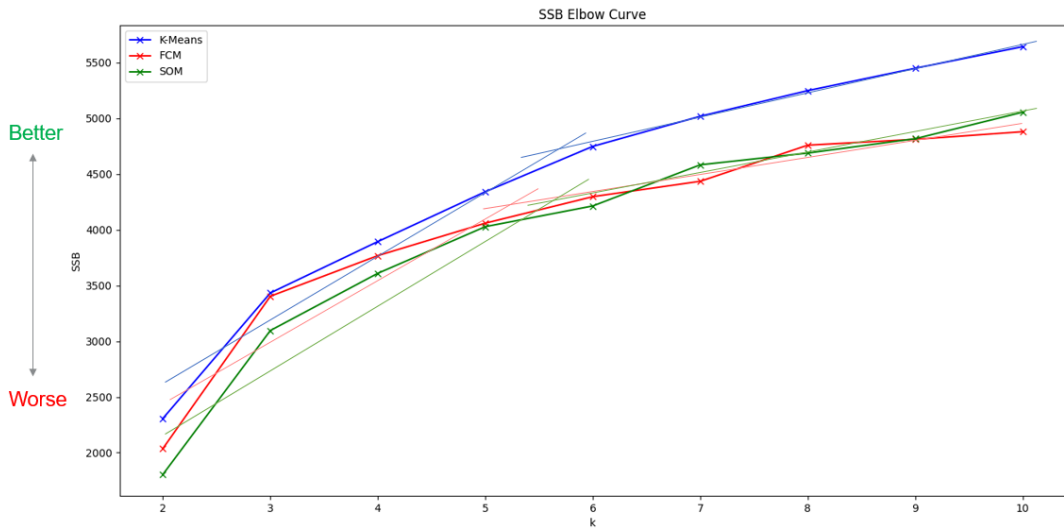
When comparing all the algorithms, FCM returned the best result for most cluster numbers except when $K = 7$, which in case K-Means returned a better result. All the clustering methods returned "fair" results according to the definition, except for FCM and K-Means when $K = 2$, which returned a "good" classification.

The Silhouette Elbow Curve is not analyzed looking for an elbow point, but now, crossing the information with the previous metrics, it is easier to pick which should be the best number of clusters. We have seen with SSW and SSB that a compact and distinct clustering result should have at least 6 clusters. Analyzing the SI results in Graph 12, the best number of clusters with K higher than 6 is 8 for FCM, 7 for K-Means, and 7 for

Graph 11 – SSB Speed Elbow Curve



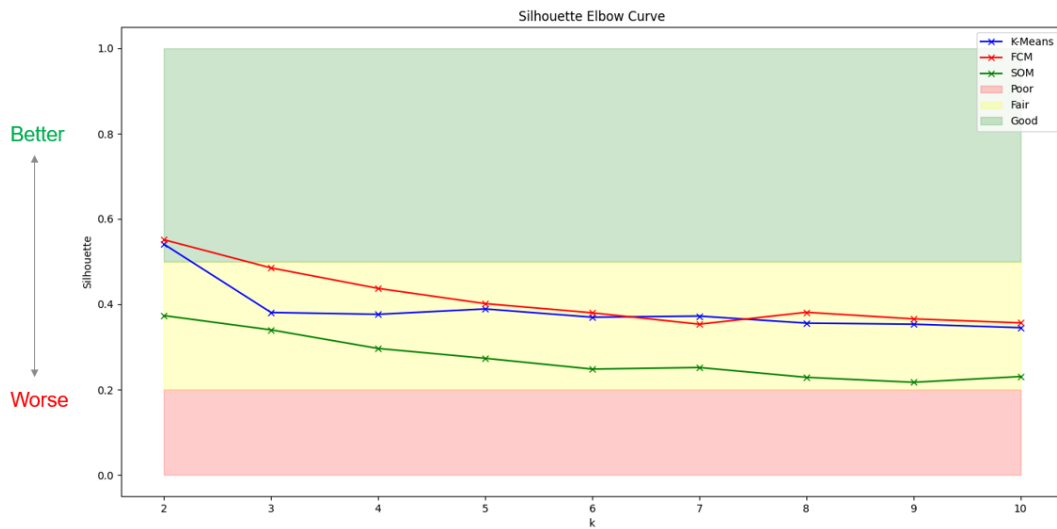
(a) SSB Elbow Curve



(b) Adjusted SSB Elbow Point

Source: Own authorship (2022).

Graph 12 – SI Speed Elbow Curve



Source: Own authorship (2022).

SOM.

4.2.1.4 Homogeneity

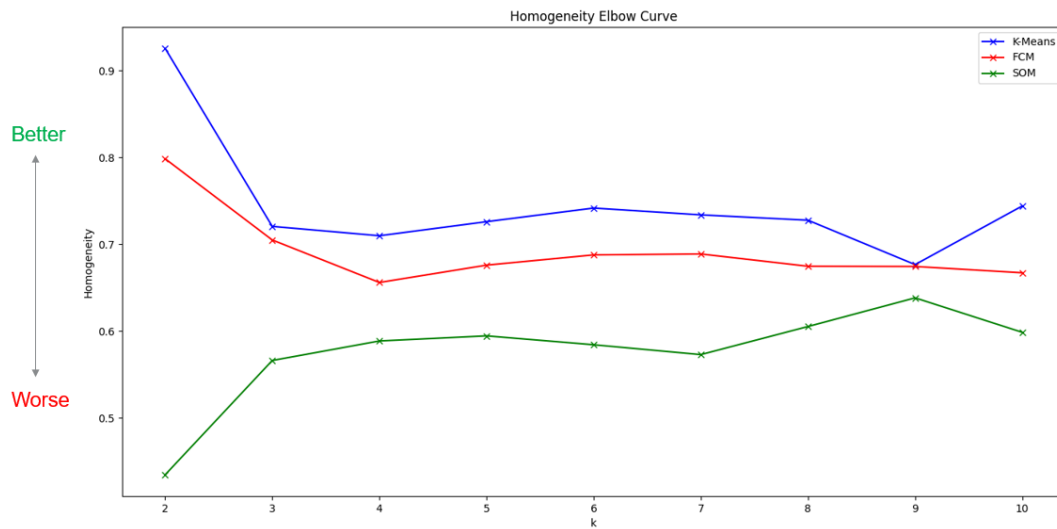
The last metric, the k-Fold Homogeneity Score, is an algorithm performance consistency validation. In the same way as the SI, it is a scaled score from 0 to 1.0, where 1.0 is a perfectly homogeneous clustering. The results for each cluster are presented in Graph 13.

The K-Means algorithm shows more consistency than the others, with an average score close to 0.83, followed by FCM with 0.75 on average. SOM had the worst results, with an average score of around 0.6. K-Means has the highest consistency, particularly for the region that is understood to have the best number of clusters.

4.2.1.5 Coefficient of variation

In the Table 4, we can see how our algorithms have performed through 10 iterations each according to every metric for every number of clusters from 2 to 10. With the CV calculation, we can see the mean variation for the performance measures.

Graph 13 – k-Fold Homogeneity Speed Elbow Curve



Source: Own authorship (2022).

In the K-Means algorithm, the CV values are very low, except for the HS, which has values from 3.25% to 12.31%, which means that the grade for this score may vary a bit more in different initializations, while the others will remain very close to be the same score always since the CV is close to zero. The K-Means problem of the randomness of initialization that would affect the quality of the solution could be mitigated once; looking to the SSW, SSB and SI metrics, we had close to zero variation in the performance results. The algorithm, however, suffers a bit on the consistency of the HS metric, which shows that its prediction error may vary but will not impact the performance quality.

For the FCM algorithm, the results for SSW, SSB and SI are also close to zero, if not. The CV is also very low for HS, varying from 0.92% to 3.53%. Therefore, it is seen that FCM is less dependent on the initialization than K-Means and has more performance consistency.

Analyzing the SOM CV results, we see that this algorithm suffered more in the consistency of the performance. Values may vary from 1.13% and 3.94% for SSW, 0.94% and 10.89% for SSB, 1.95% and 8.50% for SI, 2.21% and 39.95% for HS. Despite the high values for $K = 2$ and $K = 3$, for most of the K s the scores have less variation.

In general, Table 6 presented an analysis of the consistency of algorithms for quality measurements across many startups. FCM showed more consistency than the

others, which means less dependence on the initialization aspect.

Table 6 – Speed Coefficient Variation table

| K | K-Means | | | | FCM | | | | SOM | | | |
|----|---------|-------|-------|--------|-------|-------|-------|-------|-------|--------|-------|--------|
| | SSW | SSB | SI | HS | SSW | SSB | SI | HS | SSW | SSB | SI | HS |
| 2 | 0.00% | 0.00% | 0.01% | 3.25% | 0.00% | 0.00% | 0.00% | 3.53% | 3.00% | 10.89% | 8.50% | 39.95% |
| 3 | 0.00% | 0.00% | 0.01% | 12.31% | 0.00% | 0.00% | 0.00% | 1.49% | 3.94% | 6.69% | 5.79% | 9.59% |
| 4 | 0.00% | 0.00% | 0.00% | 10.25% | 0.00% | 0.00% | 0.06% | 1.73% | 1.22% | 1.61% | 1.95% | 4.93% |
| 5 | 0.00% | 0.00% | 0.02% | 9.11% | 0.01% | 0.01% | 0.03% | 1.32% | 1.47% | 1.58% | 2.83% | 5.70% |
| 6 | 0.00% | 0.00% | 0.03% | 4.94% | 0.20% | 0.18% | 0.09% | 0.92% | 1.13% | 1.11% | 3.73% | 4.50% |
| 7 | 0.00% | 0.00% | 0.14% | 6.68% | 0.58% | 0.51% | 1.39% | 1.45% | 1.67% | 1.37% | 4.48% | 2.90% |
| 8 | 0.00% | 0.00% | 0.06% | 7.57% | 0.42% | 0.32% | 2.87% | 2.04% | 2.31% | 1.81% | 3.99% | 4.21% |
| 9 | 0.03% | 0.01% | 1.36% | 10.00% | 0.72% | 0.53% | 1.60% | 1.66% | 1.28% | 0.94% | 2.93% | 2.75% |
| 10 | 0.00% | 0.00% | 0.01% | 7.70% | 1.13% | 0.80% | 1.72% | 1.95% | 1.78% | 1.16% | 2.70% | 2.21% |

Source: Own authorship (2022).

4.2.2 Summary of results

Having a closer look at the results presented, Figure 14 compares each metric evaluation value for K-Means, FCM, and SOM, where it was understood to be the best number of clusters.

For SSW and SSB, presented on a scale of red to green from worst to best, K-means with 7 clusters presented the best result. Regarding k-Fold Homogeneity, presented on a scale of white to green from worst to best, K-Means also has the best result. Low grades in the Cross-Validation Homogeneity Score mean that this data set should be retrained more frequently, according to the increase in the number of samples in the data set. For SI, following the previous color rule presented in Graph 12, the FCM algorithm had the best result with 0.38, but K-means had a close result with 0.37.

Figure 14 – Speed detailed metrics comparison

| | K-Means (K = 7) | FCM (K = 8) | SOM (K = 7) |
|----------------------|-----------------|-------------|-------------|
| SSW (↓) | 3,34E+03 | 3,60E+03 | 3,77E+03 |
| SSB (↑) | 5,02E+03 | 4,76E+03 | 4,58E+03 |
| Silhouette (1,00 ↑) | 0,37 | 0,38 | 0,25 |
| Homogeneity (1,00 ↑) | 0,73 | 0,67 | 0,57 |

Source: Own authorship (2022).

The K-Means methodology has presented the most compact and distinct groups, SOM the least. First, there should be a reminder that for FCM's SSW, SSB, and k-Fold Homogeneity, it is considered only the cluster with the highest membership value, that is, the Hard C-Means way, since there is no specific calculation that can include the fuzzy aspect the same way there is for SI, presented on Equation 16 of the previous chapter.

It undoubtedly impacts the grade obtained for each index. FCM algorithm had the best clustering performance looking to the SI result but was only 0.01 ahead of K-means, which had tighter and more distinct clusters.

Before analyzing the centroids of the clusters, it is necessary to look into the silhouette of each cluster for all algorithms, presented in Table 7. The FCM cluster with the highest SI score is the number 2, with 0.53, but the lowest has a 0.22 score. K-Means, however, has a much more equal score between clusters: the highest value is 0.39, and the lowest is 0.33. The highest score of SOM has a SI of 0.29, and the lowest 0.22. Even though FCM has the highest SI score, K-Means has an equal classification score between clusters.

Table 7 – Average silhouette of Speed clusters

| FCM | Average of SILHOUETTE | K-Means | Average of SILHOUETTE | SOM | Average of SILHOUETTE |
|-----|-----------------------|---------|-----------------------|-----|-----------------------|
| 0 | 0.51 | 0 | 0.39 | 0 | 0.28 |
| 1 | 0.42 | 1 | 0.33 | 1 | 0.26 |
| 2 | 0.53 | 2 | 0.36 | 2 | 0.27 |
| 3 | 0.35 | 3 | 0.37 | 3 | 0.27 |
| 4 | 0.37 | 4 | 0.39 | 4 | 0.28 |
| 5 | 0.36 | 5 | 0.28 | 5 | 0.22 |
| 6 | 0.24 | 6 | 0.39 | 6 | 0.29 |
| 7 | 0.22 | - | - | - | - |

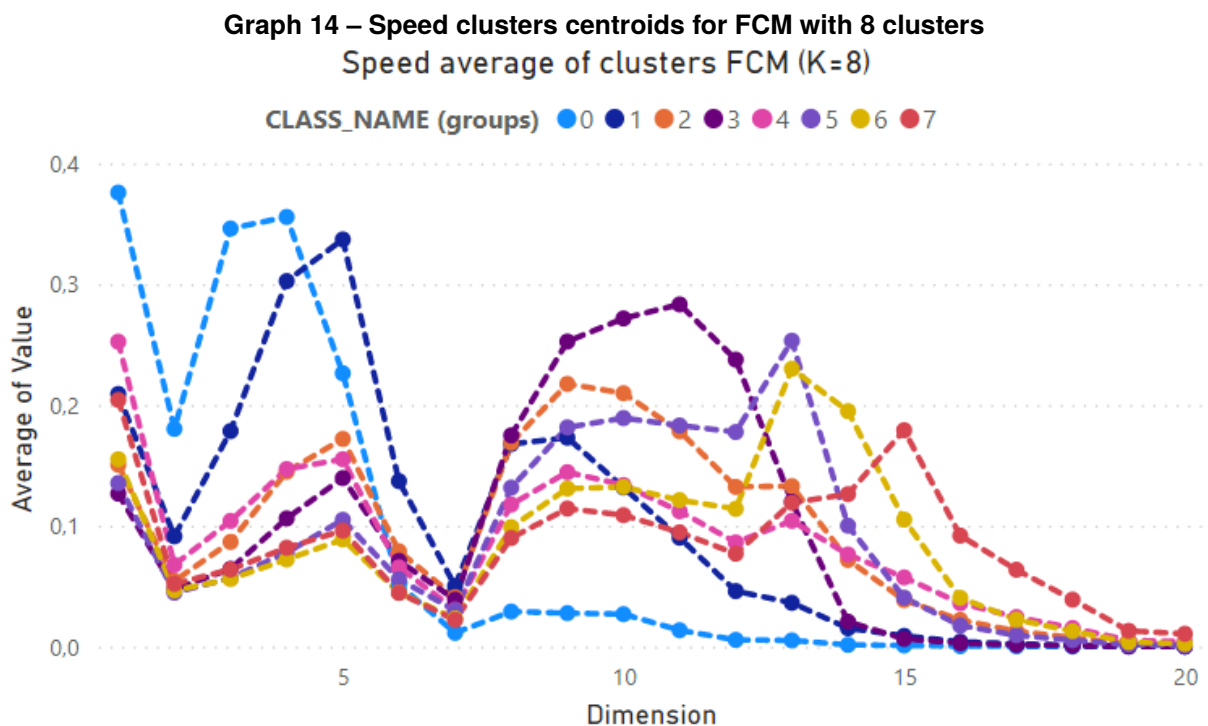
Source: Own authorship (2022).

4.2.3 Centroids of the clusters

To see the clusters' shapes, a line graph shows the center of clusters, represented as the mean value of members within the cluster for each dimension. In this section, the results of Graph 12 are presented since they showed to be the best numbers of clusters for each algorithm.

Graph 14 presents the average of clusters for FCM ordered by mode value when not considering dimension 1 (idle). With that order, we can recognize cluster 0 as the slowest and 7 as the fastest. According to what was presented in Table 2, clusters 0 and 1 are the most representative in the region that we understand as low speeds; while in the medium speed ranges, clusters 2 and 3 are the most representative; and cluster 5, 6 and 7 are the ones with the highest mode for high-speed profiles. Cluster 4, which has the mode in dimension 9, is not the most representative in any speed range, being a cluster of vehicles that travel in many speed ranges.

Being the most representative cluster of a range of speed means that vehicles that work most of the time in that profile will be classified in that respective cluster. Considering clusters that are the most representative of the same speed profile, we can see differences between them. For example, considering a profile that works most of the time in medium speed ranges but rarely in high-speed ranges, it can be classified in cluster 3, since, of the most representative clusters in the medium speeds, cluster 3 is the one that has the lowest values in high-speed profiles.

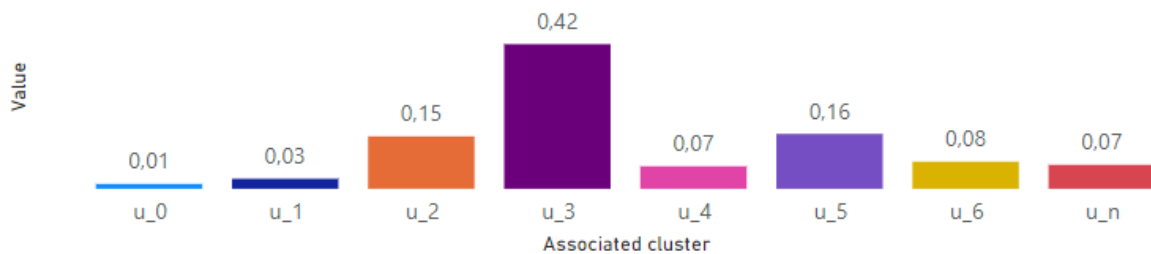


Source: Own authorship (2022).

However, Graph 14 presents the average based only on the Hard C-Means part of the method, that is, considering only the cluster of highest membership grade for every vehicle. With the membership grades, we can have deeper information on the clustering of a sample, as presented in Graph 15, that has the membership values for the FCM clustering of the vehicle presented previously in Graph 9. We see that this vehicle has 42% belonging to cluster 3, on purple, and 16% of membership value in cluster 5 and 15% in cluster 2. Since this data set has more than 3 dimensions and cannot be spatially plotted, with membership values, in an overlapping data set, we can detect outliers and to which clusters a data point is closest. In this example, we can interpret that this vehicle, when driving in low-speed ranges, may be closer to cluster 2 behaviors, such as it is closer to cluster 5 at high speeds, while most of the time, it is

part of cluster 3 at medium speed profiles.

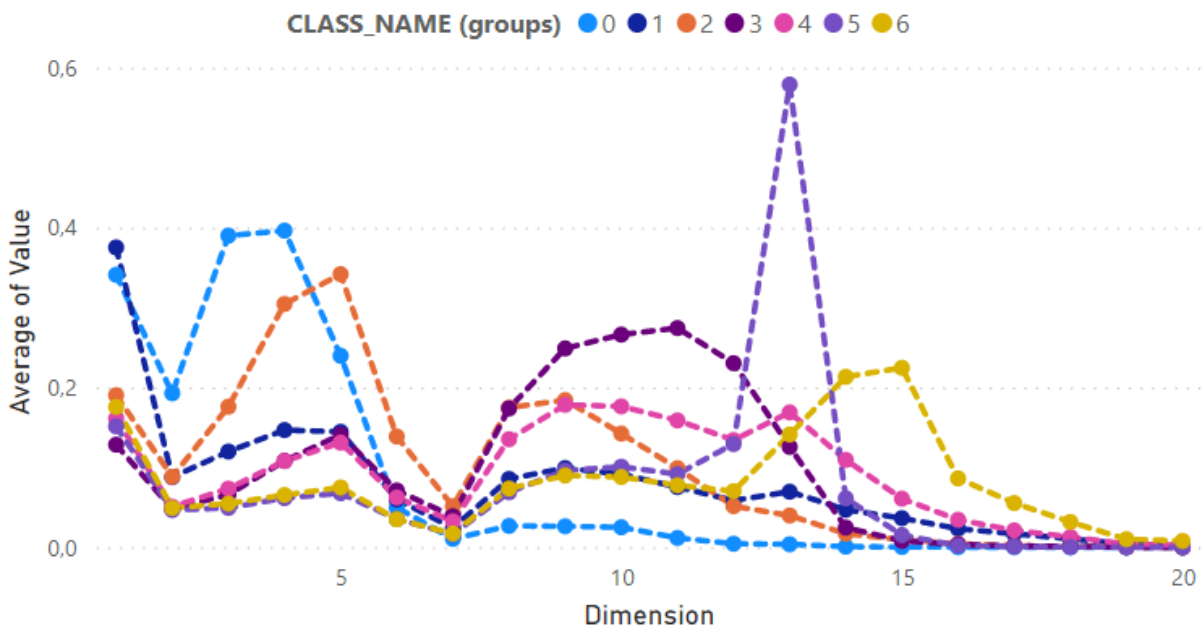
Graph 15 – Example of membership grades of a vehicle in Speed data set
FCM membership degree



Source: Own authorship (2022).

Graph 16 shows the centroids of the clusters found by K-Means, presented with the same order rule as FCM’s clusters. Clusters 0 and 2 are the most representative clusters in the low-speed ranges. The medium speed is defined mainly by clusters 3 and 4, while clusters 5 and 6 represent the most high-speed profiles. Cluster 1, despite having the mode in the low-speed ranges, is primarily an intermediate cluster between all speed ranges.

Graph 16 – Speed clusters average from K-Means with 7 clusters
Speed average of clusters K-Means (K=7)

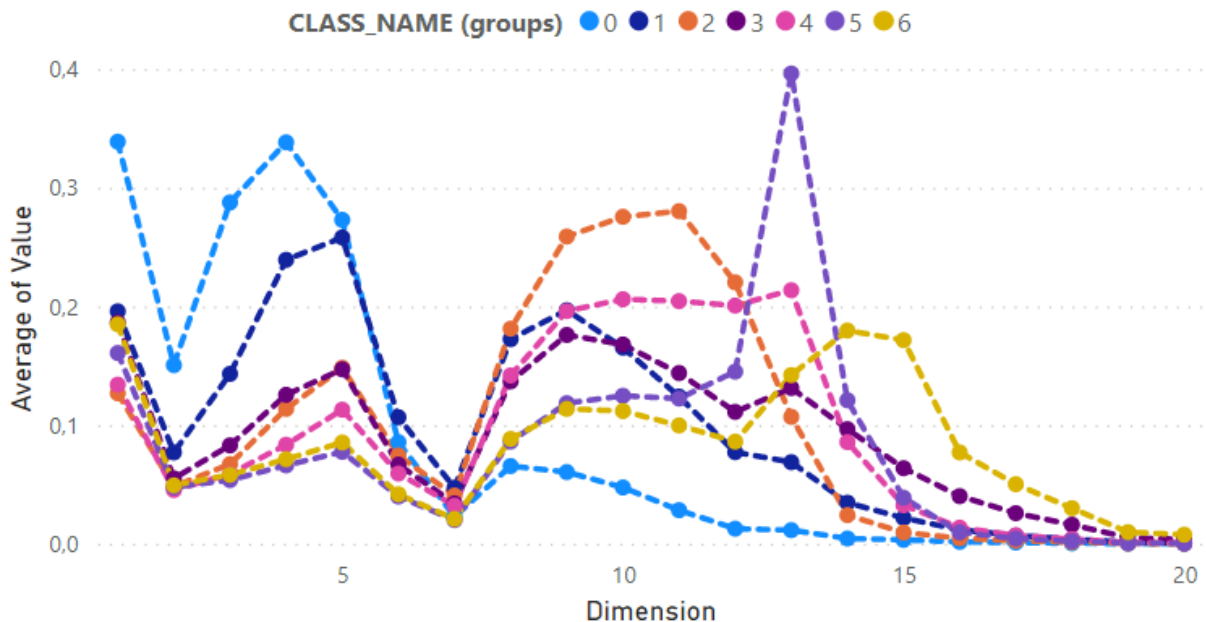


Source: Own authorship (2022).

Graph 17 shows the centroid of the clusters found by SOM, presented following the same sequence of lowest mode to highest. Clusters 0 and 1 are the ones that most

represent low-speed profiles, while clusters 2, 3, and 4 are the most defining clusters of medium speed, and clusters 5 and 6 are the most representative of high-speed ranges.

Graph 17 – Speed clusters average from SOM with 7 clusters
Speed average of clusters SOM (K=7)



Source: Own authorship (2022).

The clusters found by all the algorithms are similar to each other to a certain extent. We see a pattern of recognizing at least two clusters in low-speed ranges, two for medium speeds, two for high-speed profiles, and one not well defined in any speed range, with equal values in different speed regions. For the clusters that are well defined, the first cluster of low ranges constantly has the mode speed in dimension 4, while the second has the mode in dimension 5. In the medium-speed clusters, the first has the highest value in dimension 11 and the second cluster in dimension 9. For the high-speed profiles, the first cluster has the mode in dimension 13 and the second in dimension 14 or 15.

There were some overlapping situations in the clusters found by the algorithms, but even though we can differentiate them for some aspects of analyzing the centroids. This considerable overlap in the data set directly impacts the grades returned by the SI score since it considers the distance to its cluster and the closest one being so close that the SI score has dropped hard.

The first and slowest cluster is present in the low-speed ranges, with close to zero values in the others. This type of vehicle drives slow and is usually identifying a

cluster related to high load or high inclination jobs only, perhaps without traveling too much on highways. The second cluster of low-speed ranges, though, has higher values in middle ranges, which are often related to highways or empty load, but most of the time, a vehicle in this cluster will drive slow.

The medium-speed range clusters can be identified as vehicles on highways facing traffic, high or low inclination, with average load sizes. The second cluster most present in medium speed ranges has higher values at high speeds. It can be for more time driving empty load or the driver profile.

The high-speed clusters are made of vehicles that travel in the medium speed range, but most of the time, they are at high speed. Low slope percentages, good road conditions, or low load duty can be related to low slope percentages. This profile has to do much with the driver's behavior; once this data is sensitive, high-speed driving is significantly related to the driver's characteristics.

The intermediate cluster may be related to travels on highways with additional traffic and slope conditions or with a non-constant weight carried by the vehicle, since each time it carries different loads, the vehicle will drive with different speed limits. We see in this cluster prominent values in high speeds, which are most feasible when the vehicle is empty, but also some values in slower ranges that may represent traffic or full load conditions.

To identify what the vehicles in the cluster are mostly doing by only looking at the speed without the product specification is very difficult since there are different product types for different speed dynamics. Even so, the specification of a product will not ensure that it will work to the specification it was designed to, just as we can not be sure that a driving profile will be assumed to be a specific product type, even more than the driver profile itself is a thing. For that, it is fascinating to cross the information with other data sets to discover more about the clusters, but for now, all these analyses are initial hypotheses that we can infer with only the speed clustering information.

4.2.4 Vehicle distribution in clusters

Analyzing all the samples as a whole, we show in Graph 18 how the data set is clustered for each algorithm.

For FCM in Graph 18(a), cluster 3, with 19.49%, had the highest number of

samples, which is 175.83% higher than cluster 0, which had the lowest with 7.07%. In terms of size, cluster 3 is followed by clusters 5, 6, 4, 2, 7, and 0. For FCM on these graphs, it was considered for each vehicle only the cluster with the highest membership degree for calculation purposes, transforming into the Hard C-Means result. For FCM the largest cluster is the cluster that is most present in medium range speeds, with low values in high speed, and the smallest cluster is the slowest.

For K-Means in Graph 18(b), cluster 4 of K-Means had 41.95% of the samples and was 1.386.13% bigger than 5, which had the lowest count, with 2.82%. Cluster 3 is the second largest, followed by clusters 2, 6, 1, 0, 5.

In Graph 18(c), for SOM, cluster 3 had 20.10% of the samples and was the highest count, 183.85% higher than cluster 5, which had the lowest number, with 7.08%. Cluster 3 was followed by clusters 2, 4, 0, 1 and 5 in size.

In K-Means and SOM cluster, the cluster that has intermediate values in most of the dimensions is the largest one. For these clustering algorithms, principally K-Means, most of the population drive in different speed ranges throughout its life.

4.3 Gross Combination Weight (GCW)

For the GCW data set, we did not have a non-ML method to compare with, so all methods are compared using the elbow curve changing from $K = 2$ to $K = 10$.

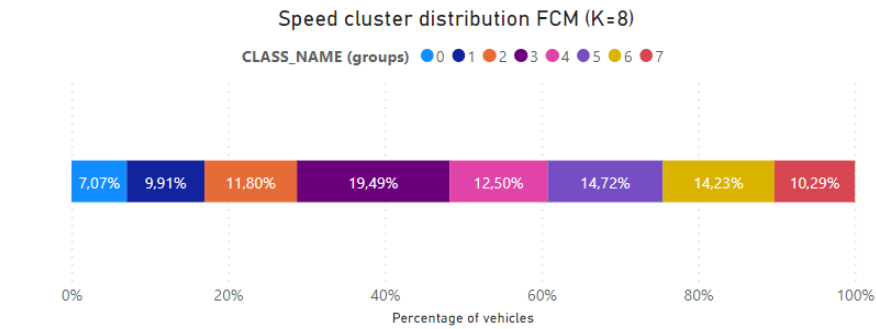
4.3.1 Performance evaluation

Some metrics were used to evaluate the algorithm's performance and choose the best number of clusters that describe the GCW data set. SSW, SSB, SI score and HS are presented, followed by its CVs and a summary of the evaluation metrics. The centroids and the vehicle distribution of the clusters found by the best algorithm for each method will be presented.

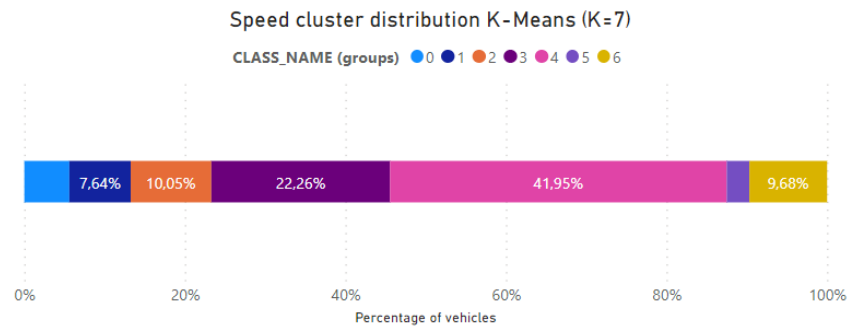
4.3.1.1 SSW

The SSW (Sum of Squares Within Clusters), presented in Graph 19, has the lowest value as the best result, representing the most compact cluster. In this analysis, it

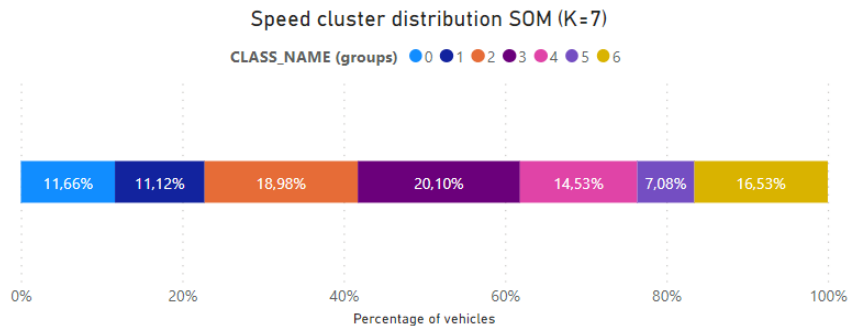
Graph 18 – Speed clusters data set distribution



(a) Samples distribution in FCM clusters



(b) Samples distribution in K-Means clusters



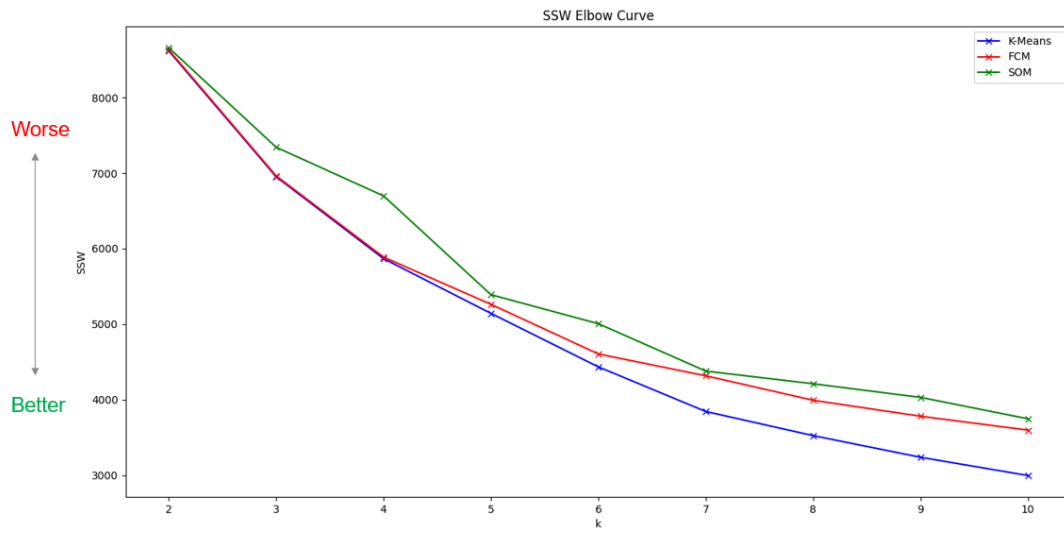
(c) Samples distribution in SOM clusters

Source: Own authorship (2022).

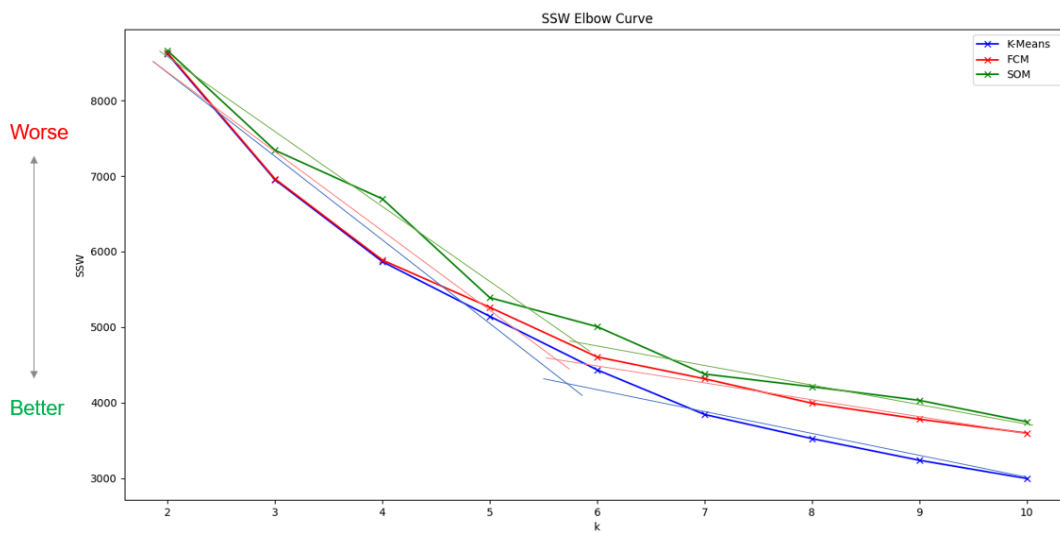
is sought for the inflection point, the point that, from it, increasing the number of clusters will no longer bring huge gains. Crossing that information with other metrics will give a better view of the best number of clusters for the problem.

The K-Means methodology (blue line) returned the best result for every cluster K , followed by FCM. To get the elbow point, in Graph 19(b), two trend lines help indicating where the inflection point is the cross them. The first trend line is a straight line drawn from the first point of the elbow curve, following the trend for 5 periods. The second trend line comes oppositely, from the last point in the curve ($K = 10$), following backward the elbow curve for 5 periods until it crosses the first line. The increase from 3 to 5 periods, compared with the slope's elbow method, is because, in this case, the tendency

Graph 19 – SSW GCW Elbow Curve



(a) SSW Elbow Curve



(b) Adjusted SSW Elbow Point

Source: Own authorship (2022).

change more smoothly, and with 3 periods, the elbow point would be too far from the curve itself, losing the relationship with it. With the virtual elbow point allocation, we see that the best number of clusters would be with at least $K = 6$.

4.3.1.2 SSB

The second evaluation metric is the SSB (Sum of Squares Between Clusters), which measures how far clusters are from each other, so the highest value is the best result. A good SSB value says that the clusters are well defined. As in the SSW analysis, two trend lines help indicate the best number of clusters. The results are shown in Graph 20.

K-Means returned the best result for every cluster number K , followed by FCM. Analyzing the elbow point with the help of trend lines, the inflection point for all the algorithms can be considered $K = 6$.

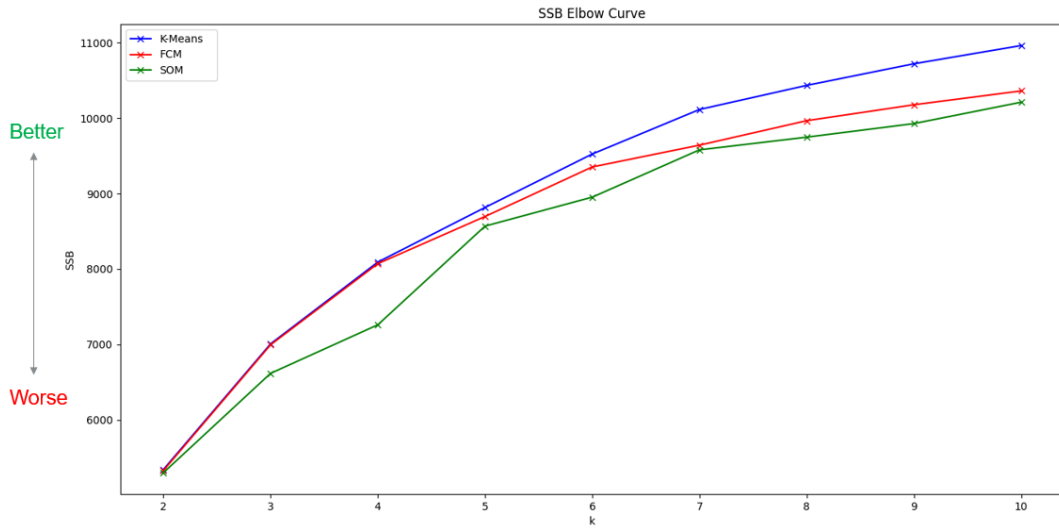
4.3.1.3 Silhouette

The third and most important metric for this study, Silhouette Index (SI), is an internal validation measurement. It is a normalized score, from -1.0 to 1.0 , which the closer to 1, the better the data is classified. Graph 21 illustrates the results considering the average SI score of the data set for each number of clusters K . For this index, SI values from 0.5 to 1.0 are considered a good result (green region on the graph), from 0.2 to 0.5 a fair result (yellow region), and from -1.0 to 0.2 a poor clustering (red region) (KAUFMAN; ROUSSEEUW, 2009).

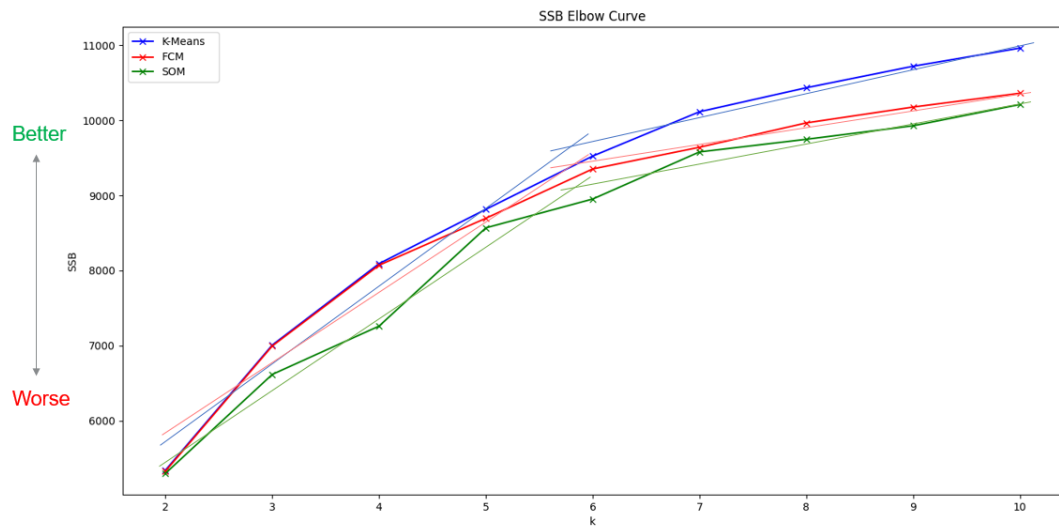
When comparing all the algorithms, FCM returned the best result until $K = 5$, in case K-Means becomes the best algorithm. FCM until 4 clusters, K-Means until 6 clusters, and SOM with 2 clusters returned "good" results. Except for FCM with 9 clusters that returned a "poor" result, all the other results are in the "fair" region.

The Silhouette Elbow Curve is not analyzed looking for an elbow point, but now, crossing the information with the previous metrics, it is easier to pick which should be the best number of clusters. We have seen with SSW and SSB that a compact and distinct clustering result should have at least 6 clusters. Analyzing the SI results in Graph 21, the best number of clusters with K higher than 6, is 6 for all algorithms.

Graph 20 – SSB GCW Elbow Curve



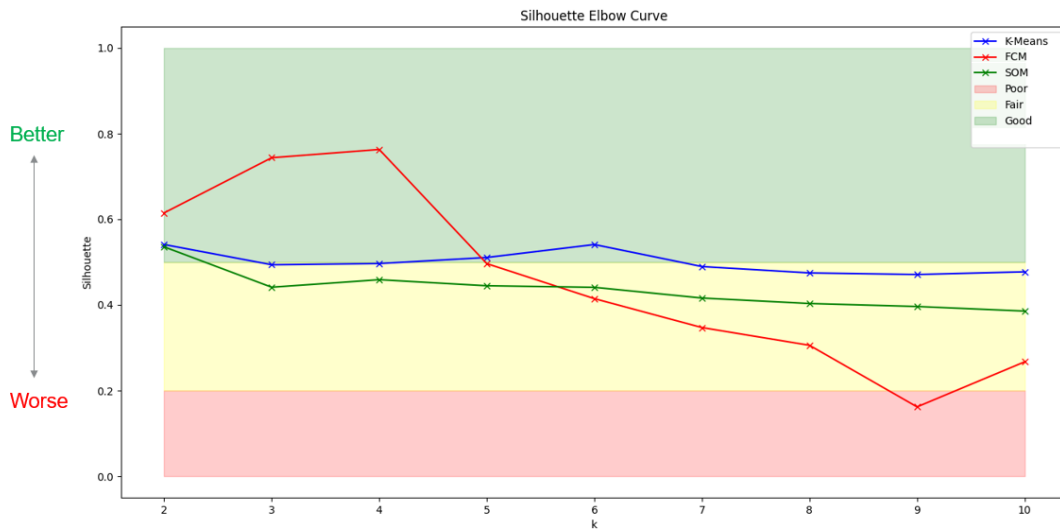
(a) SSB Elbow Curve



(b) Adjusted SSB Elbow Point

Source: Own authorship (2022).

Graph 21 – SI GCW Elbow Curve

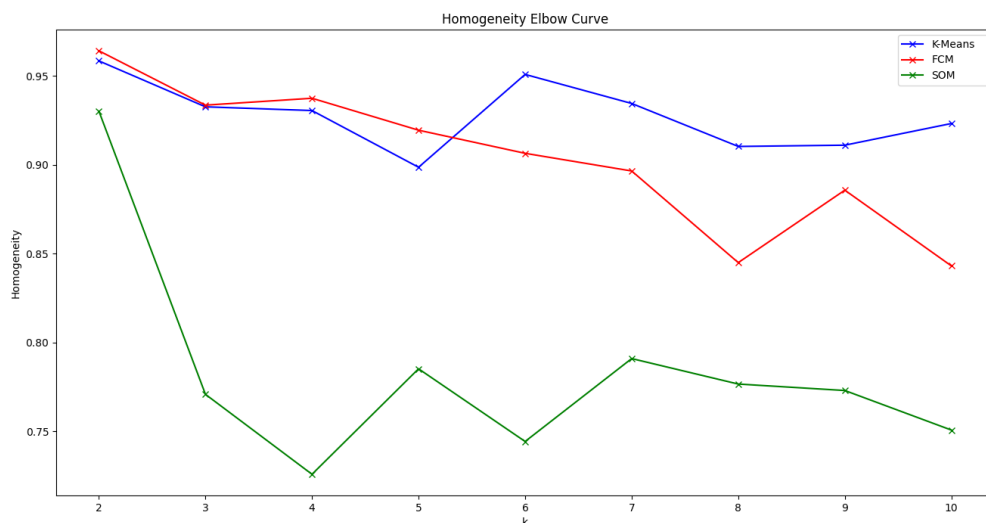


Source: Own authorship (2022).

4.3.1.4 Homogeneity

The last metric, k-Fold Homogeneity Score, is an algorithm performance consistency validation. In the same way as the SI, it is a scaled score from 0 to 1.0, where 1.0 is a perfectly homogeneous clustering. The results for each cluster are presented in Graph 22.

Graph 22 – k-Fold Homogeneity GCW Elbow Curve



Source: Own authorship (2022).

The K-Means algorithm shows more consistency than the others, except in $K = 2$ to $K = 5$, with an average score close to 0.93, followed by FCM, the best when $K = 2$ to $K = 5$, with 0.90 average. SOM had the worst results, with an average score of around 0.78. K-Means has the highest consistency, particularly for the region that is understood to have the best number of clusters.

4.3.1.5 Coefficient of variation

In the Table 4, we can see how our algorithms have performed through 10 iterations each according to every metric for every number of clusters from 2 to 10. With the CV calculation, we can see the mean variation for the performance measures.

In the K-Means algorithm, the CV values are very low; almost all zeroed, except for HS, which has values from 0.12% to 1.50%. The K-Means problem of the randomness of initialization that would affect the quality of the solution could be mitigated once; looking at the SSW, SSB, and SI metrics, we had zero variation in the performance results.

If not, the FCM algorithm results for SSW, SSB, and SI are also close to zero. For HS, the CV is also very low, varying from 0.11% to 2.06%, which means that the grade for this score may vary a bit more in different initializations, while the others will remain very close to the same score always since the CV is close to zero.

Analyzing the SOM CV results, we see that this algorithm suffered more in the consistency of the performance. Values may vary from 0.31% and 6.62% for SSW, 0.50% and 5.14% for SSB, 0.50% and 7.34% for SI, 1.71% and 7.90% for HS. This algorithm suffers a bit more than the others in the consistency of the performance.

In general, Table 8 presented an analysis of the consistency of algorithms for quality measurements across many startups. K-means showed more consistency than the others, which means less dependence on the initialization aspect, when analyzing the first three indexes, and, for k-Fold HS, we see that the algorithm is more reliable than the others when there is more non-trained data.

4.3.2 Summary of results

Having a closer look at the results presented, Figure 15 compares each metric evaluation value for K-Means, FCM, and SOM, which was understood to be the best

Table 8 – GCW Coefficient Variation table

| K | K-Means | | | | FCM | | | | SOM | | | |
|----|---------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| | SSW | SSB | SI | HS | SSW | SSB | SI | HS | SSW | SSB | SI | HS |
| 2 | 0.00% | 0.00% | 0.00% | 0.12% | 0.00% | 0.00% | 0.00% | 0.11% | 0.31% | 0.50% | 0.50% | 1.96% |
| 3 | 0.00% | 0.00% | 0.00% | 0.17% | 0.00% | 0.00% | 0.00% | 0.34% | 0.99% | 1.10% | 2.06% | 7.90% |
| 4 | 0.00% | 0.00% | 0.00% | 0.37% | 0.00% | 0.00% | 0.00% | 0.30% | 5.57% | 5.14% | 5.13% | 7.52% |
| 5 | 0.00% | 0.00% | 0.00% | 1.50% | 0.00% | 0.00% | 0.00% | 0.40% | 2.54% | 1.60% | 3.32% | 6.36% |
| 6 | 0.00% | 0.00% | 0.00% | 0.33% | 0.00% | 0.00% | 0.00% | 0.28% | 5.95% | 3.33% | 5.51% | 7.05% |
| 7 | 0.00% | 0.00% | 0.00% | 0.36% | 0.05% | 0.02% | 0.10% | 0.52% | 1.79% | 0.82% | 2.50% | 2.37% |
| 8 | 0.00% | 0.00% | 0.00% | 0.61% | 0.09% | 0.04% | 0.00% | 2.06% | 4.67% | 2.02% | 3.63% | 3.26% |
| 9 | 0.00% | 0.00% | 0.00% | 0.21% | 0.00% | 0.00% | 0.00% | 0.40% | 6.62% | 2.69% | 7.34% | 1.71% |
| 10 | 0.00% | 0.00% | 0.00% | 0.23% | 2.72% | 0.94% | 0.00% | 0.67% | 4.91% | 1.80% | 5.33% | 2.94% |

Source: Own authorship (2022).

number of clusters.

For SSW and SSB, presented on a scale of red to green from worst to best, K-means with 6 clusters presented the best result. Regarding k-Fold Homogeneity, presented on a scale of white to green from worst to best, K-Means also has the best result. Low grades in the Cross-Validation Homogeneity Score mean that this data set should be retrained more frequently, according to the increase in the number of samples in the data set. For SI, following the previous color rule presented in Graph 21, the K-means algorithm had the best result with 0.54, followed by SOM with 0.44 and FCM with 0.41.

Figure 15 – GCW detailed metrics comparison

| | K-Means (K = 6) | FCM (K = 6) | SOM (K = 6) |
|----------------------|-----------------|-------------|-------------|
| SSW (↓) | 4,43E+03 | 4,61E+03 | 5,01E+03 |
| SSB (↑) | 9,52E+03 | 9,35E+03 | 8,95E+03 |
| Silhouette (1,00 ↑) | 0,54 | 0,41 | 0,44 |
| Homogeneity (1,00 ↑) | 0,95 | 0,91 | 0,74 |

Source: Own authorship (2022).

The K-Means methodology has presented the most compact and distinct groups, SOM the least. First, there should be a reminder that for FCM's SSW, SSB, and k-Fold Homogeneity, it is considered only the cluster with the highest membership value, that is, the Hard C-Means way, since there is no specific calculation that can include the fuzzy aspect the same way there is for SI, presented on Equation 16 of the previous chapter. It undoubtedly impacts the grade obtained for each index. K-means also had the best clustering performance looking to the SI result, in the "good" region, while the others are in the "fair" region.

Before analyzing the centroids of the clusters, let us look into the silhouette of each cluster for all algorithms presented in Table 9. The FCM cluster with the highest

SI score is 1 and 5, with 0.46, while the lowest is cluster 0, with 0.36. K-Means, with the highest SI, have clusters 4 and 5 with 0.68 and 0.61, while the lowest value is for cluster 0, with 0.42. The highest score of SOM had a SI of 0.48, and the lowest 0.40. SOM showed closer scores between the clusters, but K-Means had good results for the heaviest clusters, which put the SI average of the algorithm high above SOM.

Table 9 – Average silhouette of GCW clusters

| FCM | Average of SILHOUETTE | K-Means | Average of SILHOUETTE | SOM | Average of SILHOUETTE |
|-----|-----------------------|---------|-----------------------|-----|-----------------------|
| 0 | 0.36 | 0 | 0.42 | 0 | 0.43 |
| 1 | 0.46 | 1 | 0.46 | 1 | 0.45 |
| 2 | 0.40 | 2 | 0.46 | 2 | 0.48 |
| 3 | 0.41 | 3 | 0.45 | 3 | 0.44 |
| 4 | 0.37 | 4 | 0.68 | 4 | 0.46 |
| 5 | 0.46 | 5 | 0.61 | 5 | 0.40 |

Source: Own authorship (2022).

4.3.3 Centroids of the clusters

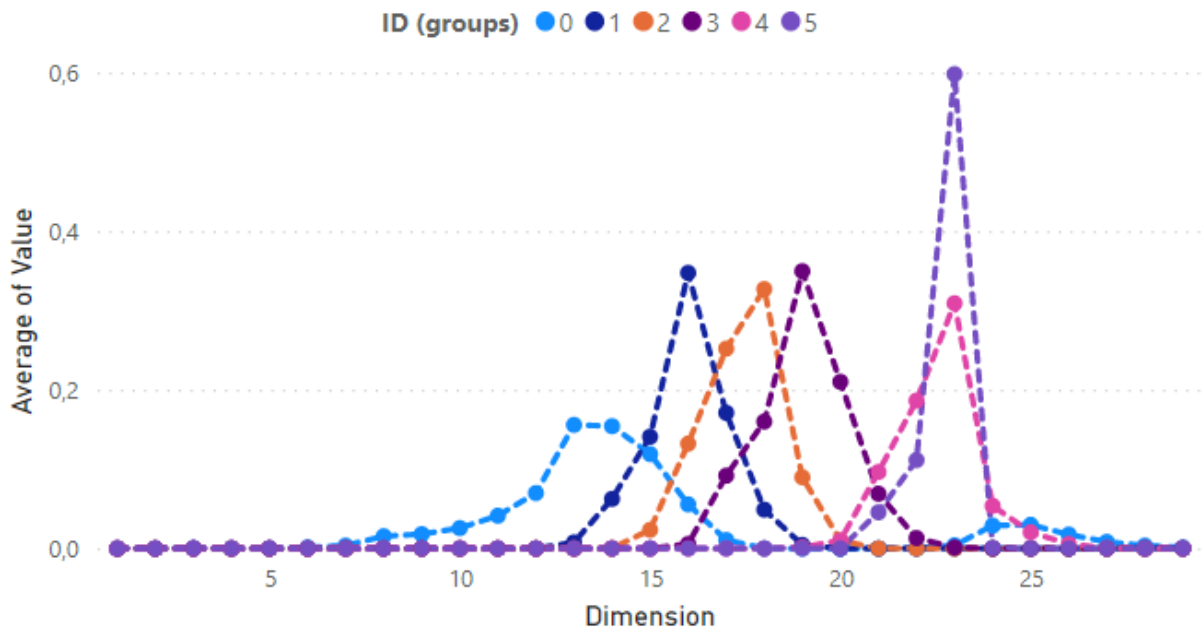
To see the clusters' shapes, a line graph shows the center of clusters, represented as the mean value of members within the cluster for each dimension. In this section, the results of Graph 21 are presented since they showed to be the best numbers of clusters for each algorithm.

Graph 23 presents the average of clusters for FCM, ordered by mode value. With that order, we can recognize cluster 0 as the lightest and cluster 5 as the heaviest. According to what was presented in Table 3, cluster 0 represents the low weight region, with the mode in dimension 13; cluster 1, the medium weight, with the mode in the 16th range; and cluster 2, 3, 4 and 5 are the ones representing high weights, with the mode in dimension 18, 19, 23 and 23, respectively. Despite not having a cluster with the highest presence in very high weights, clusters 4 and 0 have values in these ranges.

Cluster 2, which has a mode in the high-weight region, has more values in the medium than the high area, which shows that this initial separation of dimensions states a very blurry line that can only be taken as more of an explanatory rule of the dimensions, not as a classification, given the hardness of it. Cluster 2, then, can be considered a cluster with mid-high weights, just as clusters 4 and 5 are on the borderline of high and very high.

However, Graph 23 presents the average based only on the Hard C-Means

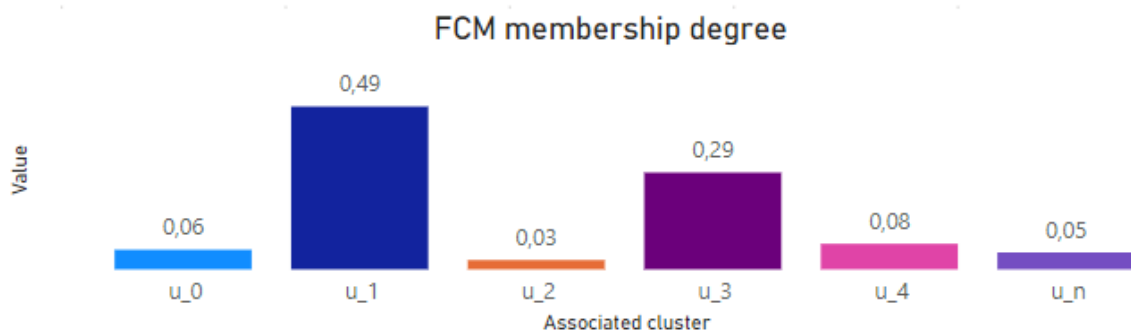
Graph 23 – GCW clusters centroids for FCM with 6 clusters
 GCW average of clusters FCM (K=6)



Source: Own authorship (2022).

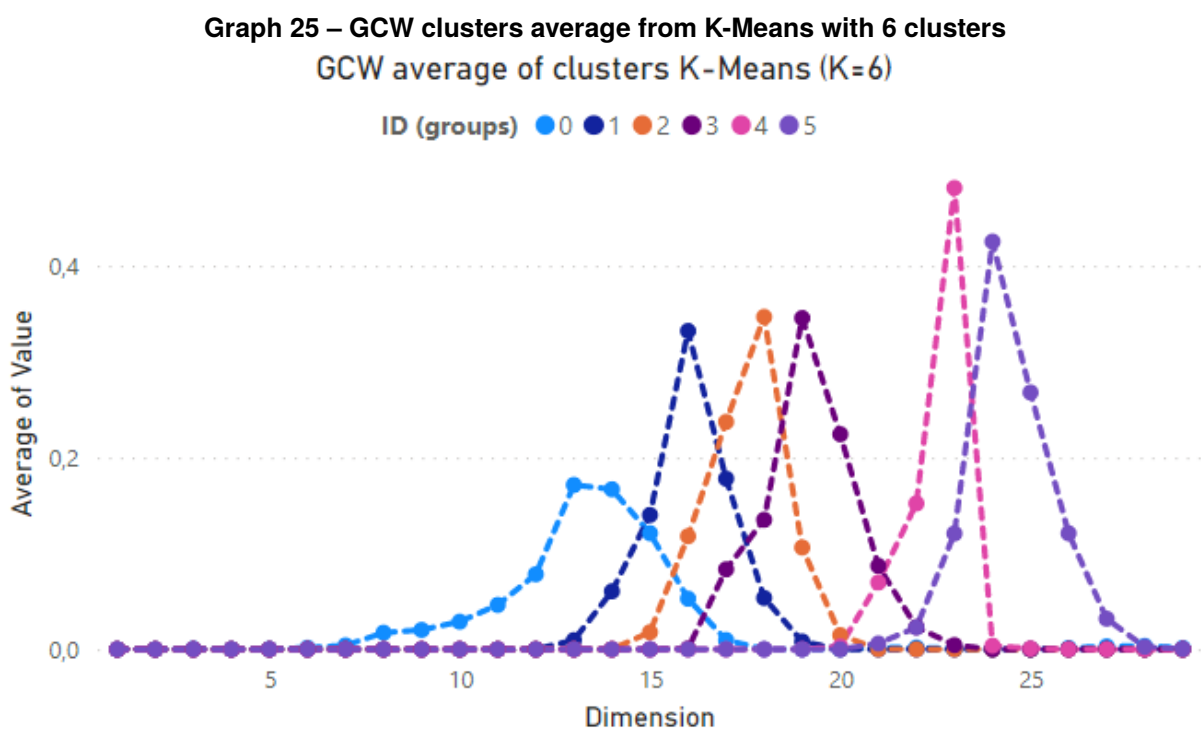
part of the method, considering only the cluster of highest membership grade for every vehicle. With the membership grades, we can have deeper information on the clustering of a sample, as presented in Graph 24, that has the membership values for the FCM clustering of the vehicle presented previously in Graph 10. We see that this vehicle has 49% on belonging to cluster 1, on blue, but also 29% of membership value in cluster 3. Since this data set has more than 3 dimensions and cannot be spatially plotted, with membership values, in an overlapping data set, we can detect outliers and to which clusters a data point is closest. In this example, this vehicle usually carries medium weight but tends to have closeness (similarities) to cluster 3, the intermediate cluster of high GCW ranges.

Graph 24 – Example of membership grades of a vehicle in GCW data set



Source: Own authorship (2022).

Graph 25 shows the centroids of the clusters found by K-Means, presented with the same order rule as FCM's clusters. Cluster 0 is the lightest cluster representing low GCWs, and cluster 1 represents medium weights, while cluster 2 is in the middle of high and medium weights, with the mode in dimension 18. Clusters 3 and 4 represent the high weights, cluster 3 the lighter high GCWs, with the mode in dimension 19, and cluster 4 the heavier, with the mode in dimension 23. Cluster 5 is the one that represents very high weights. Compared to FCM's clusters, K-means clusters 4 and 5 are better defined and separated from each other.

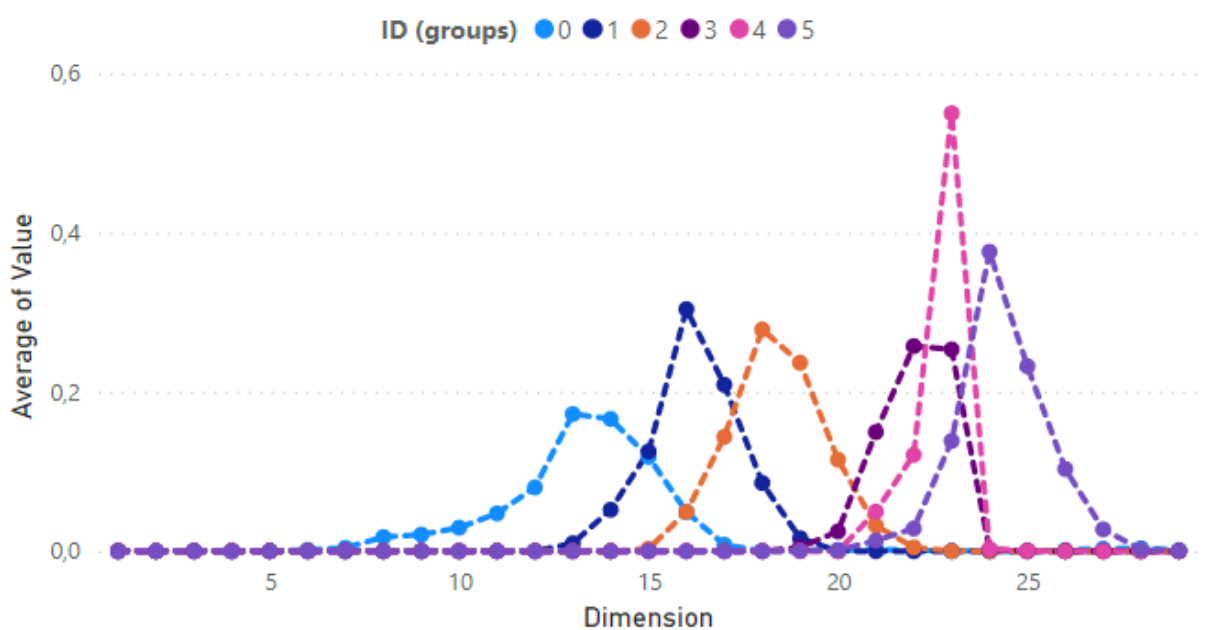


Source: Own authorship (2022).

Graph 26 shows the centroid of the clusters found by SOM, presented following the same sequence of lowest mode to highest. Cluster 0 defines the low GCW region, while cluster 1 represents medium weights, and cluster 2 is in the middle of high and medium weights. Cluster 3, which has the mode in dimension 22, represents the high ranges alongside cluster 4, which has the mode in dimension 23. Cluster 5 is the one defining very high weight ranges. In this algorithm, we see clusters 3, 4, and 5 overlapping each other a bit more, mainly for cluster 3, that in the other algorithms is not so present in dimension 23.

The clusters found by all the algorithms are similar to each other to a certain extent. We see a pattern in recognizing at least one cluster for low weight ranges, two for

Graph 26 – GCW clusters average from SOM with 6 clusters
GCW average of clusters SOM (K=6)



Source: Own authorship (2022).

medium GCWs, and one for high GCW profiles. K-means and SOM have also predicted a cluster for very high ranges, while FCM has kept one more cluster in the high region. Clusters 0, 1, and 2 are close to being the same in all algorithms; cluster 4 of SOM and K-means are the same as cluster 5 of FCM. Cluster 3 of FCM and K-means define the same region, but in SOM, that area is defined by cluster 2 only, while cluster 3 represents slightly heavier dimensions that match the area that FCM's cluster 4 represents.

There were some overlapping situations in the clusters found by the algorithms, but even though we can differentiate them for some aspects of analyzing the centroids. This overlap in middle ranges and dimension 23 directly impact the grades returned by the SI score since it considers the distance to its cluster and the closest one, and, being that short, the SI score drops hard.

4.3.4 Vehicle distribution in clusters

Analyzing all the samples as a whole, we show in Graph 27 how the data set is clustered for each algorithm.

For FCM in Graph 27(a), with 21.03%, 5 had the highest count of vehicles and was 107.11% higher than 3, which had the lowest count with 10.15%. Cluster 5 was

followed by clusters 4, 0, 1, 2, and 3. For FCM in these graphs, it was considered for each vehicle only the cluster with the highest membership degree for calculation purposes, transforming into the Hard C-Means result. For FCM the most significant cluster is the cluster that represents high GCW ranges, with the highest mode in dimension 23, and the smallest cluster is the lighter high, weight cluster. The presence of 2 significant clusters in the same area, clusters 4 and 5, around dimension 23, results in a small close cluster.

For K-Means in Graph 27(b), with 37.66%, cluster 4 is the largest and was 1.022.82% higher than 5, the smallest, with 3.35% of the data. Cluster 4 was followed by clusters 1, 0, 2, 3, and 5. The largest cluster is again the one that defines dimension 23, and the smallest is the cluster of very high GCW ranges. K-means could concentrate better on the vehicles in cluster 4 and separate well the very high in cluster 5, which is proven in Table 9, where clusters 4 and 5 had the highest silhouette score.

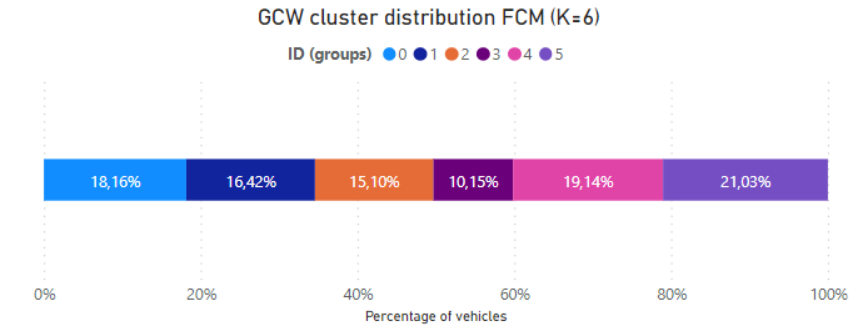
In Graph 27(c), for SOM, cluster 4, with 28.52% of the samples, had the highest number and was 624.05% higher than 5, which had the lowest, with 3.94%. Cluster 4 was followed by clusters 1, 2, 0, 3, and 5. The clusters with the most and fewest samples are defining the same region as from K-means, but poorly, since these clusters have a lower grade than the ones of K-means, according to Table 9.

4.4 Example of classification

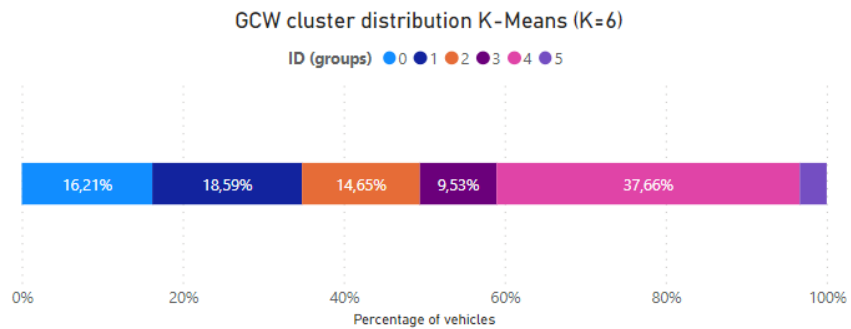
It is necessary to get the arbitrary vehicle that we have been presenting as an example since Graph 5 and analyze which clusters for all presented data set it will fall into. Graph 28 brings the cluster of the slope, speed, and GCW for this given vehicle, using the algorithms with the highest SI, after elbow analysis, for each data set.

The first data set, in Graph 28(a), presents the FCM clustering of the slope conditions this example vehicle has been through, considering the algorithm with 4 clusters. The membership values of this classification are presented in Graph 6, where this sample belongs 72% to cluster F-VHilly and 23% to cluster F-Hilly. Graph 28(b) presents the FCM clustering (considering 8 clusters) of the speed profile vehicle has accumulated throughout its life. The membership values of this classification are presented in Graph 15, where this sample belongs 42% to cluster 3, 16% to cluster 5, and 15% to cluster 2. Graph 28(c) presents the K-means clustering (considering 6 clusters) of the GCW this

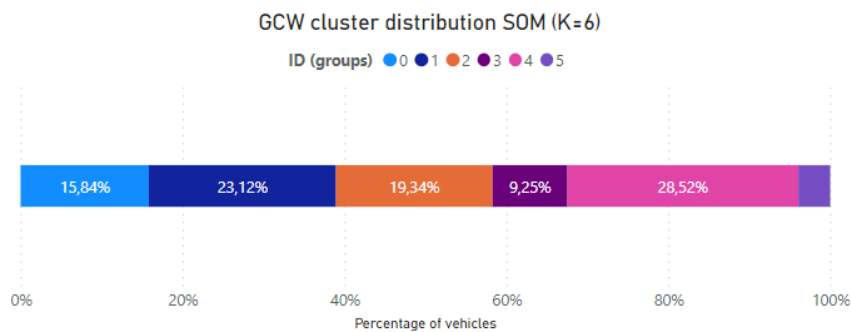
Graph 27 – GCW clusters data set distribution



(a) Samples distribution in FCM clusters



(b) Samples distribution in K-Means clusters



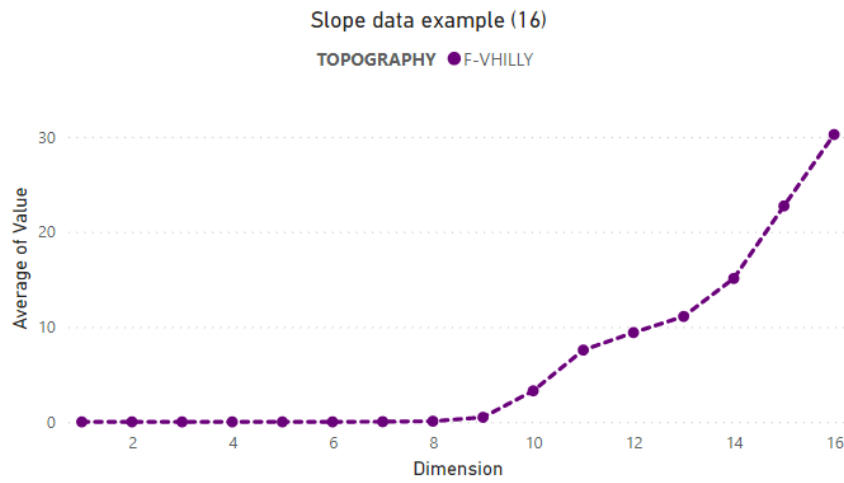
(c) Samples distribution in SOM clusters

Source: Own authorship (2022).

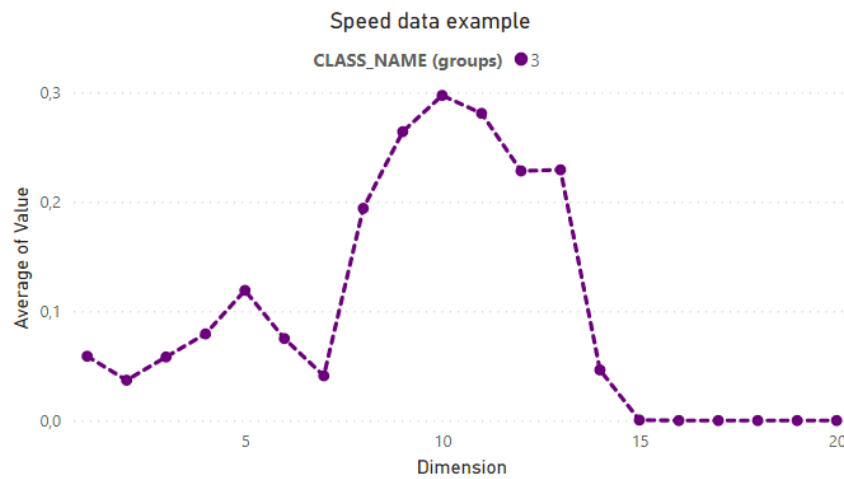
vehicle has been working with, which is part of cluster 3.

So, this vehicle, being part of the F-VHilly, is supposed to be working across mountain ranges highways, which matches the drive speed it has presented, in the medium speed cluster 3, which is continuously not fast, but not too slow, representing highways with inclinations. We can speculate that the low-speed values in dimension 5 are traffic issues and city speed limits since the vehicle does not carry such heavyweights that would enforce low-speed driving. This vehicle was clustered in the low GCW ranges, which also matches the medium speed profile since if it were a vehicle carrying much weight uphill, it would most likely have slow speed profiles. Thus, we can infer that this vehicle type of work is traveling across cities, passing through mountain ranges, and

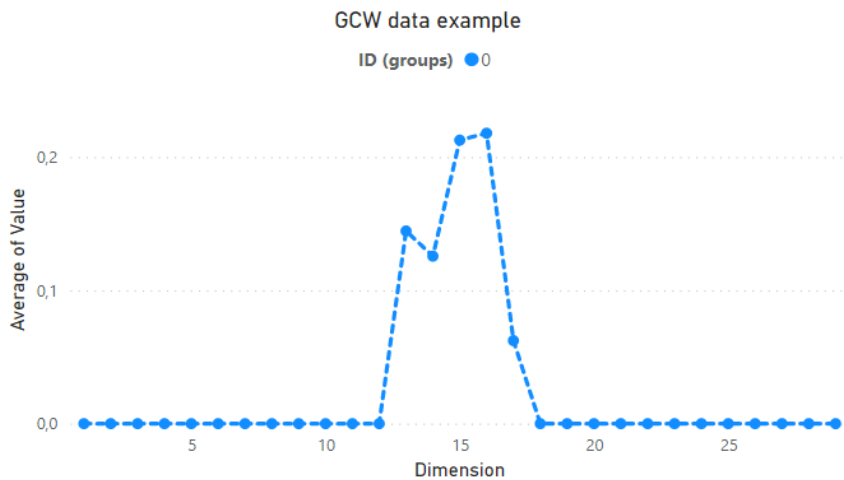
Graph 28 – Example of classification



(a) Example of slope data clustered by FCM



(b) Example of speed data clustered by FCM



(c) Example of GCW data clustered by K-means

Source: Own authorship (2022).

carrying low charges, often entering traffic areas.

Without the clustering being made, we would hardly assume these hypotheses, and if it was the case, probably not be so sure about the conclusions since we would not have references to measure the behavior with different ones. A scientific approach can give confidence in the conclusions and decisions once everything is measured and all reliable methods. By having reference profiles for each data set and being able to classify vehicles accordingly, the industry can harvest the benefits of clustering, which are, for example, offering value to customers with specialized services and products given the application since we can perfectly detect different work niches and its behaviors; product design and development that can innovate in market deficiencies, such as by knowing the usage of vehicles for different operation than it was designed to, for lacking appropriate vehicle; for commercial support, by selling appropriate vehicle given the desired application that the clusters can bring an average profile; etc.

5 CONCLUSION

Companies must be aware and prepare for changes and challenges, and that is for this time, the Big Data, Industry 4.0, and the digital transformation. Technologies have the power to disrupt unprepared companies and mutate the market share ultimately. Every corporation should be cautious that to survive strongly, it needs to become much closer to the software and services industry, and for that, working with data is essential. The earlier that is realized, the greater the chances to survive, just like stated by Kay (2008), "the best way to predict the future is to invent it".

The machine learning research goal is not to seek a universal learning algorithm or the absolute best of them. Instead, it is to understand what kinds of distributions are relevant to real-world applications and what kinds of ML algorithms perform well on essential data drawn distributions (GOODFELLOW; BENGIO; COURVILLE, 2016).

As an interesting alternative tool to extract functional patterns from data sets, machine learning can be divided into supervised learning, unsupervised learning, and reinforcement learning. The unsupervised approach can be advantageous to learn how elements in a data set can be clustered based on their similarities. This can be applied in many instances in the modern industry, considering how much data can be collected from processes and products that need to be better understood.

ML is already being used worldwide to help managers' decision-making on automated driving and intelligent vehicles developments in the automotive sector. This work was brought a study over some vehicle data that comprehend years of extensive usage on Latin America's roads, provided by a major vehicle company for personal use. The datasets have parameters that can be used in product development, predictive maintenance, and many other environments of growth and development.

To extract most of those parameters and in a way to better comprehend the data, clustering that information is needed. Classifying the data in groups, for example, can help engineers understand how the vehicle's behavior compares to others with the same characteristics. Unsupervised clustering was used to execute the task; especially three methods were implemented: K-Means, Fuzzy C-Means and Self Organizing Maps.

The first parameter refers to the records of the slope conditions that each vehicle went through throughout its life. Before the clustering process itself, that parameter must pass through a data preprocessing since real-world data have dirty, noises, or are

often incomplete. That stage cleans, transforms, and normalizes the data to be better clustered. The algorithms' results are compared with those provided by a non-ML classification, built with extensive empirical knowledge, using performance metrics.

The performance evaluation is made with Sum of the Squares Within Clusters (SSW), Sum of the Squares Between Clusters (SSB), Silhouette Index (SI), and K-Fold Cross-Validation using Homogeneity Score. These methods helped visualize the best number of clusters for each algorithm and compare the performance between them to choose the best algorithm. The first ones, SSW and SSB, were used to pre-define a minimum optimal region to then pick the best result with SI. The last metric was used to check if any of the methods were not reliable enough.

SOM with 5 clusters returned the best result in terms of compactness (SSW), but despite having also the best result in SSB, the best algorithm in terms of silhouette is FCM with 4 clusters. For the Cross-Validation Homogeneity Score, the most consistent method is the GTA classification, while K-Means is the most consistent among the ML methods most of the time.

Overall, the ML methods have presented a better clustering performance than the non-ML method, GTA classification. K-Means and FCM with 4 and 5 clusters were considered to have a "good" classification, while GTA and SOM with 5 clusters were considered to have a "fair" performance. The lack of reproducibility by the ML methods, seen with the Cross-Validation Homogeneity Score, means that the data set should be retrained more frequently, according to the increase of the number of samples.

After the clustering we could separate the data set into, at least, a cluster that retains very flat slopes, a cluster that has few values in higher inclinations, one that has more presence high inclinations, and another that is even more present in high inclinations. With these clusters well defined, we were able to speculate some environments that would be the case for each one. Introducing a new cluster for K-Means and SOM brought interesting results, mainly for K-Means that could keep a "good" silhouette, above 0.5. The new 5th cluster separates a specific application of very high jobs only, like mine extraction, from the 4th cluster, which are vehicles that work a lot through mountain highways. FCM and GTA couldn't detect this new cluster, even when increasing to $K = 5$ on FCM case.

FCM has the benefit of bringing a soft-clustering analysis, which means that a vehicle can be classified into different clusters with a degree of membership. SOM (in

this project) and K-Means, however, are hard-clustering methods, so a data point can belong to just one single cluster. This particularity of FCM gives an extra insight for the analyst to understand the vehicle behavior, since we can know that, despite being part of one cluster, the data point may be closer to a specific one than the other clusters.

Analyzing different average working cases and some outliers, we could see that the classifications made by the proposed methods, in general, make more sense with the reality than the GTA classification could get. Reducing the size of the Hilly population and increasing the size of the others, brought great benefit to the clustering and have direct impact on the Silhouette grade of these clustering methods. The non-ML method was very restrictive in the Flat cluster, while the Hilly was too much comprehensive, including a lot of different applications into one cluster. The ML methods could well read that from the data and return more reasonable and well identified clusters.

The second data analyzed refers to speed profiles that vehicles have gone through throughout their lives. The data set was cleared, normalized and then clusterized by the proposed algorithms: K-Means, Fuzzy C-Means and Self Organizing Maps. The performance evaluation is made with Sum of the Squares Within Clusters (SSW), Sum of the Squares Between Clusters (SSB), Silhouette Index (SI), and K-Fold Cross-Validation using Homogeneity Score. These methods helped visualize the best number of clusters for each algorithm and compare the performance between them to choose the most appropriated one.

K-Means with 7 clusters has returned the best result in terms of compactness (SSW), but despite having also the best result in SSB, the best algorithm in terms of silhouette is FCM with 8 clusters. All algorithms had a "fair" performance in terms of SI. For the Cross-Validation Homogeneity Score, the most consistent method is K-Means. Low grades in this metric means that the data set should be retrained more frequently as the number of samples increases. SOM had the worst performance in all metrics.

With the clustering, we were able to define at least two cluster for each speed profile region (low speed ranges, medium speed ranges and high speed ranges) and one cluster that would be an intermediate between all speed ranges.

The speed clustering alone can not give clear information about the working profile. The data is very sensible to driver behavior, weight carried, slope conditions, road conditions, etc. It would be interesting to cross the clustering with product specifications to be able to have some clearer conclusion about the clusters. However, in the other way

around, the clustering of speed profiles can help other data sets to be interpreted and understood. By clustering vehicles by speed, companies are able to analyze different usage of vehicles, that may share or not same characteristics, and further analyze part performance in different speed clusters, looking to maintenance needs and product design, or help customers to buy the vehicle specification that fits the desired speed profile knowing the average outcome of the product characteristics, when looking to the commercial purposes that information can give.

The third data analyzed refers to Gross Combination Weight (GCW) profiles that vehicles have carried throughout their lives. The data set was cleared, normalized and then clusterized by the proposed algorithms, while the performance evaluation is made with Sum of the Squares Within Clusters (SSW), Sum of the Squares Between Clusters (SSB), Silhouette Index (SI), and K-Fold Cross-Validation using Homogeneity Score. These methods helped visualize the best number of clusters for each algorithm and compare the performance between them to choose the most appropriated one.

K-Means with 6 clusters returned the best result in terms of compactness (SSW), distinctness (SSB), grouping validity (SI) and homogeneity (HS). K-Means result had a "good" performance in terms of SI, while FCM and SOM's best had a "fair" clustering. For the Cross-Validation Homogeneity Score, the most consistent method is K-Means and the least is SOM.

With the clustering, we were able to clearly define at least one cluster for low weights, two clusters for medium and one for high GCW profiles. K-means and SOM also introduced a cluster for very high GCW ranges, while FCM add one more to high region. The clusters were well defined, making evident the main working weights in the data set, especially for K-Means, which managed to well classify the high GCW area. By clustering vehicles by weight, companies are able to analyze different usage of vehicles and further analyze part performance in GCW clusters, looking to maintenance needs, product efficiency, product design, etc.

Working with ML methods can help analysts to better comprehend big data sets and clarify individual behavior. A scientific approach can give confidence in the conclusions, once everything is measured and all methods are reliable, and theoretical basis for decision making. With clustering, trucks that have similar working pattern of usage can be found and compared in a sea of data where they would be hardly put side by side. This comparison can help on offering personalized products, on fuel

consumption analysis, on understanding the maintenance needs for each behavior and, of course, for product development. With this, we managed to answer the initial question "How can historical usage data help a truck manufacturer improve product development and fuel consumption?" and help the company that supported this work to aggregate and increase value of data.

Trucks are the main means of transport for logistics in many countries. In China, for example, they are responsible for 76% of the national logistics. With a good data extraction and well-defined clusters, works like this can help drivers improve their fuel efficiency, saving up not just liters of fuel and money for companies, but reducing the impact of logistics on the environment. The best truck driver in efficiency, when compared to an average driver in a same class, can save, for example, 3285 liters a year, which is 10282.05 Kg of CO₂ considering diesel as the fuel (HAO; YANG; ZHOU, 2019) (SCALA JUNIOR, 2013).

For future works, other clustering methodologies and metrics could be implemented and used to compare with the present methods. This work can also be extended to other types of vehicle's data set, which will lead to a even more precise comparison of vehicles. Also, given the analysis that the soft-clustering FCM brought to the study, we can also implement something similar for the hard-clustering methods by calculating for each sample the distance to its own cluster and to the other clusters. This could help to see if a sample is an outlier or is close to its center, such as the membership degrees could help understand the belonging to each cluster. The adjustment of clustering methods' hyper-parameters can be optimized with optimization techniques exploring other fields of computational intelligence, such as evolutionary computing (BEZDEK, J. C., 1994).

REFERENCES

ACETO, Giuseppe; PERSICO, Valerio; PESCAPÉ, Antonio. Industry 4.0 and health: internet of things, big data, and cloud computing for healthcare 4.0. en. **Journal of Industrial Information Integration**, v. 18, p. 100129, June 2020. ISSN 2452414X. DOI: 10.1016/j.jii.2020.100129. Available from: <https://linkinghub.elsevier.com/retrieve/pii/S2452414X19300135>. Visited on: 28 Mar. 2022.

AGGARWAL, Charu C.; REDDY, Chandan K. **Data clustering: algorithms and applications**. [S. l.]: CRC Press, Aug. 2013. Google-Books-ID: JU_SBQAAQBAJ. ISBN 978-1-4665-5822-9.

AKANBI, Oluwatobi Ayodeji; AMIRI, Iraj Sadegh; FAZELDEHKORDI, Elahe. Chapter 4 - feature extraction. In: AKANBI, Oluwatobi Ayodeji; AMIRI, Iraj Sadegh; FAZELDEHKORDI, Elahe (Eds.). **A Machine-Learning Approach to Phishing Detection and Defense**. Boston: Syngress, 2015. P. 45–54. ISBN 978-0-12-802927-5. DOI: <https://doi.org/10.1016/B978-0-12-802927-5.00004-6>. Available from: <https://www.sciencedirect.com/science/article/pii/B9780128029275000046>.

ALAM, Shafiq *et al.* Research on particle swarm optimization based clustering: a systematic review of literature and techniques. **Swarm and Evolutionary Computation**, v. 17, p. 1–13, 2014. ISSN 2210-6502. DOI: <https://doi.org/10.1016/j.swevo.2014.02.001>. Available from: <https://www.sciencedirect.com/science/article/pii/S2210650214000145>.

ALHONIEMI, Esa *et al.* Process monitoring and modeling using the self-organizing map. en. **Integrated Computer-Aided Engineering**, v. 6, n. 1, p. 3–14, Jan. 1999. Publisher: IOS Press. ISSN 1069-2509. DOI: 10.3233/ICA-1999-6102. Available from: <https://content.iospress.com/articles/integrated-computer-aided-engineering/ica00029>. Visited on: 1 June 2022.

ALIA, Osama; MANDAVA, Rajeswari; AZIZ, Mohd. A hybrid harmony search algorithm for mri brain segmentation. **Evolutionary Intelligence**, v. 4, p. 31–49, Mar. 2011. DOI: 10.1007/s12065-011-0048-1.

ALTINTAS, Nihat; TRICK, Michael. A data mining approach to forecast behavior. en. **Annals of Operations Research**, v. 216, n. 1, p. 3–22, May 2014. ISSN 1572-9338. DOI: 10.1007/s10479-012-1236-9. Available from: <https://doi.org/10.1007/s10479-012-1236-9>. Visited on: 3 May 2022.

APERGIS, Nicholas; FILIPPIDIS, Ioannis; ECONOMIDOU, Claire. Financial deepening and economic growth linkages: a panel data analysis. en. **Review of World Economics**, v. 143, n. 1, p. 179–198, Apr. 2007. ISSN 1610-2878, 1610-2886. DOI: 10.1007/s10290-007-0102-3. Available from: <http://link.springer.com/10.1007/s10290-007-0102-3>. Visited on: 10 Nov. 2021.

ASHRAFI, Amir *et al.* The role of business analytics capabilities in bolstering firms' agility and performance. en. **International Journal of Information Management**, v. 47, p. 1–15, Aug. 2019. ISSN 02684012. DOI: 10.1016/j.ijinfomgt.2018.12.005. Available from:

<https://linkinghub.elsevier.com/retrieve/pii/S0268401218307734>. Visited on: 5 Nov. 2021.

BAÇÃO, Fernando; LOBO, Victor; PAINHO, Marco. Self-organizing maps as substitutes for k-means clustering. en. *In: SUNDERAM, Vaidy S. et al. (Eds.). (Lecture Notes in Computer Science)*, p. 476–483. ISBN 978-3-540-32118-7. DOI: 10.1007/11428862_65.

BÆRENHOLDT, Jørgen Ole (Ed.). **Design research: synergies from interdisciplinary perspectives**. New York: Routledge, 2010.

BALL, Geoffrey H.; HALL, David J. **isodata, a novel method of data analysis and pattern classification**. en. [S. l.], Apr. 1965. Section: Technical Reports. Available from: <https://apps.dtic.mil/sti/citations/AD0699616>. Visited on: 21 Feb. 2022.

BEGENAU, Juliane; FARBOODI, Maryam; VELDKAMP, Laura. Big data in finance and the growth of large firms. **Journal of Monetary Economics**, v. 97, p. 71–87, 2018. ISSN 0304-3932. DOI: <https://doi.org/10.1016/j.jmoneco.2018.05.013>.

Available from:

<https://www.sciencedirect.com/science/article/pii/S0304393218302174>.

BELLMAN, Richard. Adaptive control processes: a guided tour princeton university press. **Princeton, New Jersey, USA**, p. 96, 1961.

BENESTY, Jacob *et al.* Pearson correlation coefficient. *In: COHEN, Israel et al. (Eds.). Noise Reduction in Speech Processing*. Berlin, Heidelberg: Springer, 2009. (Springer Topics in Signal Processing). P. 1–4. ISBN 978-3-642-00296-0. DOI: 10.1007/978-3-642-00296-0_5. Available from:

https://doi.org/10.1007/978-3-642-00296-0_5. Visited on: 2 May 2022.

BERNSTEIN, A. J. Analysis of programs for parallel processing. **IEEE Transactions on Electronic Computers**, EC-15, n. 5, p. 757–763, Oct. 1966. Conference Name: IEEE Transactions on Electronic Computers. ISSN 0367-7508. DOI: 10.1109/PGEC.1966.264565.

BERSCH, Christopher V.; AKKERMAN, Renzo; KOLISCH, Rainer. Strategic planning of new product introductions: integrated planning of products and modules in the automotive industry. en. **Omega**, v. 105, p. 102515, Dec. 2021. ISSN 03050483. DOI: 10.1016/j.omega.2021.102515. Available from:

<https://linkinghub.elsevier.com/retrieve/pii/S0305048321001249>. Visited on: 28 Mar. 2022.

BEYER, Mark; LANEY, Douglas. **The importance of 'big data': a definition**. en. [S. l.: s. n.], June 2012. Available from: <https://www.gartner.com/en/documents/2057415/the-importance-of-big-data-a-definition>. Visited on: 25 Nov. 2021.

BEZDEK, James. **Pattern recognition with fuzzy objective function algorithms**. [S. l.: s. n.], Jan. 1981. Journal Abbreviation: Pattern Recognition with Fuzzy Objective Function Algorithms Publication Title: Pattern Recognition with Fuzzy Objective Function Algorithms. ISBN 978-1-4757-0452-5. DOI: 10.1007/978-1-4757-0450-1.

BEZDEK, James C. **What is computational intelligence?** [S. l.], 1994.

BEZDEK, James C.; EHRLICH, Robert; FULL, William. fcm: the fuzzy c-means clustering algorithm. en. **Computers & Geosciences**, v. 10, n. 2-3, p. 191–203, Jan. 1984. ISSN 00983004. DOI: 10.1016/0098-3004(84)90020-7. Available from: <https://linkinghub.elsevier.com/retrieve/pii/0098300484900207>. Visited on: 16 Feb. 2022.

BILLINGSLEY, Patrick. **Probability and measure**. [S. l.]: Wiley, May 1995. Google-Books-ID: z39jQgAACAAJ. ISBN 978-0-471-00710-4.

BONACCORSO, G. **Machine learning algorithms**. [S. l.]: Packt Publishing, 2017. ISBN 9781785884511. Available from: https://books.google.com.br/books?id=%5C_-ZDDwAAQBAJ.

BRADLEY, Paul S; BENNETT, Kristin P; DEMIRIZ, Ayhan. Constrained k-means clustering. **Microsoft Research, Redmond**, v. 20, n. 0, 2000.

BRASIL, Oracle. **O que é big data? | oracle brasil**. [S. l.: s. n.], 2021a. Available from: <https://www.oracle.com/br/big-data/what-is-big-data/>. Visited on: 5 Nov. 2021.

BRASIL, Oracle. **O que é ciência de dados? pt**. [S. l.: s. n.], 2021b. Available from: <https://www.oracle.com/br/data-science/what-is-data-science/>. Visited on: 23 Nov. 2021.

CAMPELLO, R.J.G.B.; HRUSCHKA, E.R. A fuzzy extension of the silhouette width criterion for cluster analysis. en. **Fuzzy Sets and Systems**, v. 157, n. 21, p. 2858–2875, Nov. 2006. ISSN 01650114. DOI: 10.1016/j.fss.2006.07.006. Available from: <https://linkinghub.elsevier.com/retrieve/pii/S0165011406002892>. Visited on: 22 Feb. 2022.

CELEBI, M. Emre; KINGRAVI, Hassan A. Linear, deterministic, and order-invariant initialization methods for the k-means clustering algorithm. In: **Partitional Clustering Algorithms**. Ed. by M. Emre Celebi. Cham: Springer International Publishing, 2015. P. 79–98. ISBN 978-3-319-09259-1. DOI: 10.1007/978-3-319-09259-1_3. Available from: https://doi.org/10.1007/978-3-319-09259-1_3.

CHANIAS, Simon; HESS, Thomas. Understanding digital transformation strategy formation: insights from europe’s automotive industry. **PACIS**, v. 296, 2016.

CÔRTE-REAL, Nadine; RUIVO, Pedro; OLIVEIRA, Tiago. Leveraging internet of things and big data analytics initiatives in european and american firms: is data quality a way to extract business value? **Information & Management**, v. 57, n. 1, p. 1–16, Jan. 2020.

ISSN 0378-7206. DOI: 10.1016/j.im.2019.01.003. Available from: <http://www.scopus.com/inward/record.url?scp=85059737130partnerID=8YFLogxK>. Visited on: 5 Nov. 2021.

CURRIE, Janet; KLEVEN, Henrik; ZWIERS, Esmée. Technology and big data are changing economics: mining text to track methods. en. **AEA Papers and Proceedings**, v. 110, p. 42–48, May 2020. ISSN 2574-0768, 2574-0776. DOI: 10.1257/pandp.20201058. Available from: <https://pubs.aeaweb.org/doi/10.1257/pandp.20201058>. Visited on: 28 Mar. 2022.

DAHL, Oskar *et al.* Understanding association between logged vehicle data and vehicle marketing parameters: using clustering and rule-based machine learning. *In: (IMMS 2020)*, p. 13–22. ISBN 978-1-4503-7546-7. DOI: 10.1145/3416028.3417215. Available from: <https://doi.org/10.1145/3416028.3417215>. Visited on: 3 May 2022.

DE MAURO, Andrea; GRECO, Marco; GRIMALDI, Michele. A formal definition of big data based on its essential features. en. **Library Review**, v. 65, n. 3, p. 122–135, Apr. 2016. ISSN 0024-2535. DOI: 10.1108/LR-06-2015-0061. Available from: <https://www.emerald.com/insight/content/doi/10.1108/LR-06-2015-0061/full/html>. Visited on: 26 Nov. 2021.

DENGER, Andrea; ZAMAZAL, Klaus. Product lifecycle challenges for powertrain systems in the automotive industry. *In: Systems Engineering for Automotive Powertrain Development*. Ed. by Hannes Hick, Klaus Küpper and Helfried Sorger. Cham: Springer International Publishing, 2020. P. 1–18. ISBN 978-3-319-68847-3. DOI: 10.1007/978-3-319-68847-3_4-1. Available from: https://doi.org/10.1007/978-3-319-68847-3_4-1.

DEVELOPERS, Google. **Machine learning crash course**. en. [S. l.: s. n.], 2021. Available from: <https://developers.google.com/machine-learning/data-prep/transform/normalization>. Visited on: 10 Feb. 2022.

DIAS, Madson Luiz Dantas. **fuzzy-c-means: An implementation of Fuzzy C-means clustering algorithm**. [S. l.]: Zenodo, May 2019. DOI: 10.5281/zenodo.3066222. Available from: <https://git.io/fuzzy-c-means>.

DINLER, Derya; TURAL, Mustafa Kemal. A survey of constrained clustering. *In: Unsupervised Learning Algorithms*. Ed. by M. Emre Celebi and Kemal Aydin. Cham: Springer International Publishing, 2016. P. 207–235. ISBN 978-3-319-24211-8. DOI: 10.1007/978-3-319-24211-8_9. Available from: https://doi.org/10.1007/978-3-319-24211-8_9.

DOWNTON, P. **Design research**. [S. l.]: RMIT Pub., 2003. ISBN 9780864592675. Available from: <https://books.google.com.br/books?id=QeTQlylyJTYC>.

DUNN, J. C. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. **Journal of Cybernetics**, v. 3, n. 3, p. 32–57, Jan. 1973. Publisher: Taylor & Francis _eprint: <https://doi.org/10.1080/01969727308546046>. ISSN

0022-0280. DOI: [10.1080/01969727308546046](https://doi.org/10.1080/01969727308546046). Available from:
<https://doi.org/10.1080/01969727308546046>. Visited on: 14 Feb. 2022.

EINSTEIN, Albert. **Geometry and experience**. Berlin, Heidelberg, Germany, Jan. 1921. Lecture at the Prussian Academy of Science.

EL NAQA, Issam; MURPHY, Martin J. What is machine learning? *In: MACHINE learning in radiation oncology*. [S. l.]: Springer, 2015. P. 3–11.

FALCINI, Fabio; LAMI, Giuseppe; COSTANZA, Alessandra Mitidieri. Deep learning in automotive software. **IEEE Software**, IEEE, v. 34, n. 3, p. 56–63, 2017.

FAYYAD, Usama M; PIATETSKY-SHAPIRO, Gregory; SMYTH, Padhraic, *et al.* Knowledge discovery and data mining: towards a unifying framework. *In: v. 96*, p. 82–88.

FORTMANN-ROE, Scott. Understanding the bias-variance tradeoff. 2012. **URL: <http://scott.fortmann-roe.com/docs/BiasVariance.html> (visited on 12/12/2017)**, 2015.

FRIENDLY, Michael. A brief history of data visualization. *In: CHEN, Chun-houh; HÄRDLE, Wolfgang; UNWIN, Antony (Eds.). Handbook of Data Visualization*. Berlin, Heidelberg: Springer, 2008. (Springer Handbooks Comp.Statistics). P. 15–56. ISBN 978-3-540-33037-0. DOI: [10.1007/978-3-540-33037-0_2](https://doi.org/10.1007/978-3-540-33037-0_2). Available from:
https://doi.org/10.1007/978-3-540-33037-0_2. Visited on: 10 Nov. 2021.

FUSHIKI, Tadayoshi. Estimation of prediction error by using k-fold cross-validation. *en. Statistics and Computing*, v. 21, n. 2, p. 137–146, 2009. ISSN 0960-3174, 1573-1375. DOI: [10.1007/s11222-009-9153-8](https://doi.org/10.1007/s11222-009-9153-8). Available from:
<http://link.springer.com/10.1007/s11222-009-9153-8>. Visited on: 24 Feb. 2022.

GANDOMI, Amir; HAIDER, Murtaza. Beyond the hype: big data concepts, methods, and analytics. *en. International Journal of Information Management*, v. 35, n. 2, p. 137–144, Apr. 2015. ISSN 02684012. DOI: [10.1016/j.ijinfomgt.2014.10.007](https://doi.org/10.1016/j.ijinfomgt.2014.10.007). Available from:
<https://linkinghub.elsevier.com/retrieve/pii/S0268401214001066>. Visited on: 26 Nov. 2021.

GATH, I.; GEVA, A.B. Unsupervised optimal fuzzy clustering. *en. IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 11, n. 7, p. 773–780, July 1989. ISSN 01628828. DOI: [10.1109/34.192473](https://doi.org/10.1109/34.192473). Available from:
<http://ieeexplore.ieee.org/document/192473/>. Visited on: 25 Jan. 2022.

GAVALI, Pralhad; BANU, J. Saira. Chapter 6 - deep convolutional neural network for image classification on cuda platform. *In: SANGAIAH, Arun Kumar (Ed.). Deep Learning and Parallel Computing Environment for Bioengineering Systems*. [S. l.]: Academic Press, Jan. 2019. P. 99–122. ISBN 978-0-12-816718-2. DOI: [10.1016/B978-0-12-816718-2.00013-0](https://doi.org/10.1016/B978-0-12-816718-2.00013-0). Available from:

<https://www.sciencedirect.com/science/article/pii/B9780128167182000130>.
Visited on: 11 Feb. 2022.

GEISSER, Seymour. The predictive sample reuse method with applications. **Journal of the American Statistical Association**, v. 70, n. 350, p. 320–328, June 1975.
Publisher: Taylor & Francis _eprint:
<https://www.tandfonline.com/doi/pdf/10.1080/01621459.1975.10479865>. ISSN 0162-1459. DOI: 10.1080/01621459.1975.10479865. Available from:
<https://www.tandfonline.com/doi/abs/10.1080/01621459.1975.10479865>.
Visited on: 24 Feb. 2022.

GHASEMINEZHAD, M. H.; KARAMI, A. A novel self-organizing map (som) neural network for discrete groups of data clustering. en. **Applied Soft Computing**, v. 11, n. 4, p. 3771–3778, June 2011. ISSN 1568-4946. DOI: 10.1016/j.asoc.2011.02.009.
Available from:
<https://www.sciencedirect.com/science/article/pii/S156849461100069X>.
Visited on: 1 June 2022.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. Machine learning basics. **Deep learning**, MIT press Cambridge, MA, USA, v. 1, n. 7, p. 98–164, 2016.

GOWER, J. C. A comparison of some methods of cluster analysis. **Biometrics**, v. 23, n. 4, p. 623, Dec. 1967. ISSN 0006341X. DOI: 10.2307/2528417. Available from:
<https://www.jstor.org/stable/2528417?origin=crossref>. Visited on: 21 Feb. 2022.

GÜNTER, Simon; BUNKE, Horst. Self-organizing map for clustering in the graph domain. en. **Pattern Recognition Letters**, v. 23, n. 4, p. 405–417, Feb. 2002. ISSN 0167-8655. DOI: 10.1016/S0167-8655(01)00173-8. Available from:
<https://www.sciencedirect.com/science/article/pii/S0167865501001738>.
Visited on: 9 June 2022.

GUPTA, Manjul; GEORGE, Joey F. Toward the development of a big data analytics capability. en. **Information & Management**, v. 53, n. 8, p. 1049–1064, Dec. 2016. ISSN 03787206. DOI: 10.1016/j.im.2016.07.004. Available from:
<https://linkinghub.elsevier.com/retrieve/pii/S0378720616300787>. Visited on: 5 Nov. 2021.

GUPTA, Manoj; CHANDRA, Pravin. hybcim: hypercube based cluster initialization method for k-means. v. 8, p. 3584–3587, Aug. 2019. DOI: 10.35940/ijitee.J9774.0881019.

HADOOP, Apache. **Apache hadoop**. [S. l.: s. n.], 2021. Available from:
<https://hadoop.apache.org/>. Visited on: 9 Nov. 2021.

HAN, Jiawei; KAMBER, Micheline; PEI, Jian. 2 - getting to know your data. *In*: HAN, Jiawei; KAMBER, Micheline; PEI, Jian (Eds.). **Data Mining (Third Edition)**. Third Edition. Boston: Morgan Kaufmann, 2012a. (The Morgan Kaufmann Series in Data Management Systems). P. 39–82. ISBN 978-0-12-381479-1. DOI:

<https://doi.org/10.1016/B978-0-12-381479-1.00002-2>. Available from:
<https://www.sciencedirect.com/science/article/pii/B9780123814791000022>.

HAN, Jiawei; KAMBER, Micheline; PEI, Jian. 3 - data preprocessing. *In*: HAN, Jiawei; KAMBER, Micheline; PEI, Jian (Eds.). **Data Mining (Third Edition)**. Third Edition. Boston: Morgan Kaufmann, 2012b. (The Morgan Kaufmann Series in Data Management Systems). P. 83–124. ISBN 978-0-12-381479-1. DOI:
<https://doi.org/10.1016/B978-0-12-381479-1.00003-4>. Available from:
<https://www.sciencedirect.com/science/article/pii/B9780123814791000034>.

HAO, Ruru; YANG, Hangzheng; ZHOU, Zhou. Driving behavior evaluation model base on big data from internet of vehicles. *en*. **International Journal of Ambient Computing and Intelligence (IJACI)**, v. 10, n. 4, p. 78–95, Oct. 2019. Publisher: IGI Global. ISSN 1941-6237. DOI: 10.4018/IJACI.2019100105. Available from:
<https://www.igi-global.com/article/driving-behavior-evaluation-model-base-on-big-data-from-internet-of-vehicles/www.igi-global.com/article/driving-behavior-evaluation-model-base-on-big-data-from-internet-of-vehicles/238055>. Visited on: 4 May 2022.

HEY, Toy; TANSLEY, Stewart; TOLLE, Kistin. The fourth paradigm: data-intensive scientific discovery. *en*, p. 287, 2009.

HILALI, Nabil el. contemporary models of design research design research as an interplay with world successful economic models thejobsism. *en*, p. 11, 2015.

HOFFMANN, Marcus; ZAYER, Eric; STREMPPEL, Karl. **A survival guide for europe's car dealers**. [S. l.: s. n.], 2019.
<https://www.bain.com/insights/a-survival-guide-for-europes-car-dealers>. Accessed: 2022-02-01.

HONG, Yoon-Seok; ROSEN, Michael R. Intelligent characterisation and diagnosis of the groundwater quality in an urban fractured-rock aquifer using an artificial neural network. *en*. **Urban Water**, v. 3, n. 3, p. 193–204, Sept. 2001. ISSN 1462-0758. DOI: 10.1016/S1462-0758(01)00045-0. Available from:
<https://www.sciencedirect.com/science/article/pii/S1462075801000450>. Visited on: 1 June 2022.

INDUSTRIA, CCOO. Situación y perspectivas en el sector del automóvil, medidas ambientales, digitalización y automatización de la industria. **Madrid: Area de Estrategicas Sectoriales. Obtenido de <http://industria.ccoo.es/9ddeee3ef0745110d18ae92f9a4bc706000060.pdf>**, 2018.

JAIN, Anil K; MURTY, M Narasimha; FLYNN, Patrick J. Data clustering: a review. **ACM computing surveys (CSUR)**, Acm New York, NY, USA, v. 31, n. 3, p. 264–323, 1999.

JANOSKI, T.; LEPADATU, D. **Dominant divisions of labor: models of production that have transformed the world of work**. [S. l.]: Palgrave Macmillan US, 2013. (Palgrave Connect). ISBN 9781137370235. Available from:
<https://books.google.com.br/books?id=9w4tAgAAQBAJ>.

JATANA, Nishtha *et al.* **A survey and comparison of relational and non-relational database.** [S. l.: s. n.], 2012.

JAVAHERI, Sadaf Hossein; SEPEHRI, Mohammad Mehdi; TEIMOURPOUR, Babak. Chapter 6 - response modeling in direct marketing: a data mining-based approach for target selection. *In*: ZHAO, Yanchang; CEN, Yonghua (Eds.). **Data Mining Applications with R.** Boston: Academic Press, Jan. 2014. P. 153–180. ISBN 978-0-12-411511-8. DOI: 10.1016/B978-0-12-411511-8.00006-2. Available from: <https://www.sciencedirect.com/science/article/pii/B9780124115118000062>. Visited on: 11 Feb. 2022.

JOHNSON, Jeff S.; FRIEND, Scott B.; LEE, Hannah S. Big data facilitation, utilization, and monetization: exploring the 3vs in a new product development process. *en. Journal of Product Innovation Management*, v. 34, n. 5, p. 640–658, 2017. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/jpim.12397>. ISSN 1540-5885. DOI: 10.1111/jpim.12397. Available from: <https://onlinelibrary.wiley.com/doi/abs/10.1111/jpim.12397>. Visited on: 5 Nov. 2021.

JOSEPH, Seena; OLUGBARA, Oludayo O. Preprocessing effects on performance of skin lesion saliency segmentation. *en. Diagnostics*, v. 12, n. 2, p. 344, Feb. 2022. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute. ISSN 2075-4418. DOI: 10.3390/diagnostics12020344. Available from: <https://www.mdpi.com/2075-4418/12/2/344>. Visited on: 3 Feb. 2022.

JUNTUNEN, Petri *et al.* Cluster analysis by self-organizing maps: an application to the modelling of water quality in a treatment process. **Applied Soft Computing**, v. 13, p. 3191–3196, July 2013. DOI: 10.1016/j.asoc.2013.01.027.

KAMBATLA, Karthik *et al.* Trends in big data analytics. *en. Journal of Parallel and Distributed Computing*, v. 74, n. 7, p. 2561–2573, July 2014. ISSN 07437315. DOI: 10.1016/j.jpdc.2014.01.003. Available from: <https://linkinghub.elsevier.com/retrieve/pii/S0743731514000057>. Visited on: 28 Mar. 2022.

KARGARI, Mehrdad; SEPEHRI, Mohammad Mehdi. Stores clustering using a data mining approach for distributing automotive spare-parts to reduce transportation costs. *en. Expert Systems with Applications*, v. 39, n. 5, p. 4740–4748, Apr. 2012. ISSN 0957-4174. DOI: 10.1016/j.eswa.2011.09.121. Available from: <https://www.sciencedirect.com/science/article/pii/S0957417411014448>. Visited on: 3 May 2022.

KATZ, Yuri A.; BIEM, Alain. Time-resolved topological data analysis of market instabilities. *en. Physica A: Statistical Mechanics and its Applications*, v. 571, p. 125816, June 2021. ISSN 0378-4371. DOI: 10.1016/j.physa.2021.125816. Available from: <https://www.sciencedirect.com/science/article/pii/S0378437121000881>. Visited on: 10 Nov. 2021.

KAUFMAN, Leonard; ROUSSEEUW, Peter J. **Finding groups in data: an introduction to cluster analysis**. [S. l.]: John Wiley & Sons, Sept. 2009. Google-Books-ID: YeFQHiikNo0C. ISBN 978-0-470-31748-8.

KAY, Alan. **Alan kay | speaker | ted**. en. [S. l.: s. n.], 2008. Available from: https://www.ted.com/speakers/alan_kay. Visited on: 17 Dec. 2021.

KELLEHER, John D.; TIERNEY, Brendan. **Data science**. [S. l.]: MIT Press, Apr. 2018. Google-Books-ID: UlpVDwAAQBAJ. ISBN 978-0-262-34703-7.

KOHONEN, Teuvo. **Self-organizing maps**. 3. ed. [S. l.: s. n.], 2001. Available from: <https://link.springer.com/book/10.1007/978-3-642-56927-2>. Visited on: 31 May 2022.

KOLEN, J.F.; HUTCHESON, T. Reducing the time complexity of the fuzzy c-means algorithm. **IEEE Transactions on Fuzzy Systems**, v. 10, n. 2, p. 263–267, Apr. 2002. ISSN 10636706. DOI: 10.1109/91.995126. Available from: <http://ieeexplore.ieee.org/document/995126/>. Visited on: 16 Feb. 2022.

KOTARBA, Marcin. Digital transformation of business models. **Foundations of Management**, v. 10, n. 1, p. 123–142, 2018. DOI: doi:10.2478/fman-2018-0011. Available from: <https://doi.org/10.2478/fman-2018-0011>.

KRZANOWSKI, W. J.; LAI, Y. T. A criterion for determining the number of groups in a data set using sum-of-squares clustering. **Biometrics**, v. 44, n. 1, p. 23, Mar. 1988. ISSN 0006341X. DOI: 10.2307/2531893. Available from: <https://www.jstor.org/stable/2531893?origin=crossref>. Visited on: 21 Feb. 2022.

KUMAR, Pawan; SIROHI, Deepika. **Comparative analysis of FCM and HCM algorithm on iris data set**. [S. l.: s. n.], 2010.

LANEY, Douglas. **3d data management: controlling data volume, velocity, and variety**. [S. l.], 2001. Available from: <http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf>.

LARSON, S. C. The shrinkage of the coefficient of multiple correlation. **Journal of Educational Psychology**, v. 22, n. 1, p. 45–55, 1931. Place: US Publisher: Warwick & York. ISSN 1939-2176. DOI: 10.1037/h0072400.

LASI, Heiner *et al.* Industry 4.0. en. **Business & Information Systems Engineering**, v. 6, n. 4, p. 239–242, Aug. 2014. ISSN 1867-0202. DOI: 10.1007/s12599-014-0334-4. Available from: <http://link.springer.com/10.1007/s12599-014-0334-4>. Visited on: 23 Nov. 2021.

LEE, Jooyoung; JANG, Kitae. A framework for evaluating aggressive driving behaviors based on in-vehicle driving records. **Transportation Research Part F: Traffic Psychology and Behaviour**, v. 65, p. 610–619, 2019. ISSN 1369-8478. DOI:

<https://doi.org/10.1016/j.trf.2017.11.021>. Available from:
<https://www.sciencedirect.com/science/article/pii/S1369847816306684>.

LEE, Kyung-Jin; YUN, Seong-Taek, *et al.* The combined use of self-organizing map technique and fuzzy c-means clustering to evaluate urban groundwater quality in seoul metropolitan city, south korea. **Journal of Hydrology**, v. 569, p. 685–697, 2019. ISSN 0022-1694. DOI: <https://doi.org/10.1016/j.jhydrol.2018.12.031>. Available from: <https://www.sciencedirect.com/science/article/pii/S0022169418309806>.

LEHRER, Christiane *et al.* How big data analytics enables service innovation: materiality, affordance, and the individualization of service. **Journal of Management Information Systems**, v. 35, p. 424–460, May 2018. DOI: [10.1080/07421222.2018.1451953](https://doi.org/10.1080/07421222.2018.1451953).

LI, Jun; CHENG, Hong; GUO, Hongliang, *et al.* Survey on artificial intelligence for vehicles. en. **Automotive Innovation**, v. 1, n. 1, p. 2–14, Jan. 2018. ISSN 2096-4250, 2522-8765. DOI: [10.1007/s42154-018-0009-9](https://doi.org/10.1007/s42154-018-0009-9). Available from: <http://link.springer.com/10.1007/s42154-018-0009-9>. Visited on: 25 Jan. 2022.

LI, Sheng-Tun; CHENG, Yi-Chung; LIN, Su-Yu. A fcm-based deterministic forecasting model for fuzzy time series. en. **Computers & Mathematics with Applications**, v. 56, n. 12, p. 3052–3063, Dec. 2008. ISSN 0898-1221. DOI: [10.1016/j.camwa.2008.07.033](https://doi.org/10.1016/j.camwa.2008.07.033). Available from: <https://www.sciencedirect.com/science/article/pii/S0898122108004409>. Visited on: 16 Feb. 2022.

LIN, Na *et al.* An overview on study of identification of driver behavior characteristics for automotive control. en. **Mathematical Problems in Engineering**, v. 2014, e569109, Mar. 2014. Publisher: Hindawi. ISSN 1024-123X. DOI: [10.1155/2014/569109](https://doi.org/10.1155/2014/569109). Available from: <https://www.hindawi.com/journals/mpe/2014/569109/>. Visited on: 3 May 2022.

LLOPIS-ALBERT, Carlos; RUBIO, Francisco; VALERO, Francisco. Impact of digital transformation on the automotive industry. en. **Technological Forecasting and Social Change**, v. 162, p. 120343, Jan. 2021. ISSN 00401625. DOI: [10.1016/j.techfore.2020.120343](https://doi.org/10.1016/j.techfore.2020.120343). Available from: <https://linkinghub.elsevier.com/retrieve/pii/S0040162520311690>. Visited on: 25 Jan. 2022.

LONES, Michael A. How to avoid machine learning pitfalls: a guide for academic researchers. en. **arXiv:2108.02497 [cs]**, Aug. 2021. arXiv: 2108.02497. Available from: <http://arxiv.org/abs/2108.02497>. Visited on: 6 Apr. 2022.

MA, Shuaiyin *et al.* Big data driven predictive production planning for energy-intensive manufacturing industries. en. **Energy**, v. 211, p. 118320, Nov. 2020. ISSN 03605442. DOI: [10.1016/j.energy.2020.118320](https://doi.org/10.1016/j.energy.2020.118320). Available from: <https://linkinghub.elsevier.com/retrieve/pii/S0360544220314274>. Visited on: 10 Nov. 2021.

MACQUEEN, J. some methods for classification and analysis of multivariate observations. en. **MULTIVARIATE OBSERVATIONS**, p. 17, 1967.

MALACA, Pedro *et al.* Online inspection system based on machine learning techniques: real case study of fabric textures classification for the automotive industry. **Journal of Intelligent Manufacturing**, v. 30, n. 1, p. 351–361, Jan. 2019. ISSN 1572-8145. DOI: 10.1007/s10845-016-1254-6. Available from: <https://doi.org/10.1007/s10845-016-1254-6>.

MALLE, Julien. **Fuzzy clustering: an application to distributional reinforcement learning**. 2021. PhD thesis.

MELIN, Patricia *et al.* Analysis of spatial spread relationships of coronavirus (covid-19) pandemic in the world using self organizing maps. **Chaos, Solitons & Fractals**, v. 138, p. 109917, 2020. ISSN 0960-0779. DOI: <https://doi.org/10.1016/j.chaos.2020.109917>. Available from: <https://www.sciencedirect.com/science/article/pii/S0960077920303179>.

MIKALEF, Patrick *et al.* Big data and business analytics: a research agenda for realizing business value. en. **Information & Management**, v. 57, n. 1, p. 103237, Jan. 2020. ISSN 03787206. DOI: 10.1016/j.im.2019.103237. Available from: <https://linkinghub.elsevier.com/retrieve/pii/S0378720619310687>. Visited on: 5 Nov. 2021.

MOHD, Wan *et al.* maxd k-means: a clustering algorithm for auto-generation of centroids and distance of data points in clusters. *In*: COMMUNICATIONS in Computer and Information Science. [S. l.: s. n.], Jan. 2012. v. 316. Journal Abbreviation: Communications in Computer and Information Science. P. 192–199. ISBN 978-3-642-34288-2. DOI: 10.1007/978-3-642-34289-9_22.

MOORE, Gordon E. Cramming more components onto integrated circuits. en. v. 38, n. 8, p. 4, 1965.

MORAES, Felipe C. *et al.* Parallel high dimensional self organizing maps using cuda. *In*: p. 302–306. DOI: 10.1109/SBR-LARS.2012.56.

MOSTELLER, Frederick; WALLACE, David L. Inference in an authorship problem. **Journal of the American Statistical Association**, v. 58, n. 302, p. 275–309, 1963. Publisher: [American Statistical Association, Taylor & Francis, Ltd.] ISSN 0162-1459. DOI: 10.2307/2283270. Available from: <https://www.jstor.org/stable/2283270>. Visited on: 24 Feb. 2022.

MURTAGH, Fionn; CONTRERAS, Pedro. Algorithms for hierarchical clustering: an overview. en. **WIREs Data Mining and Knowledge Discovery**, v. 2, n. 1, p. 86–97, Jan. 2012. ISSN 1942-4787, 1942-4795. DOI: 10.1002/widm.53. Available from: <https://onlinelibrary.wiley.com/doi/10.1002/widm.53>. Visited on: 6 Apr. 2022.

NIEUWENHUIS, P.; WELLS, P. **The automotive industry and the environment**. [S. l.]: Elsevier Science, 2003. (Woodhead Publishing in environmental management). ISBN

9781855737136. Available from:
<https://books.google.com.br/books?id=ZyrTzv0MSE4C>.

NIEUWENHUIS, P.; WELLS, P. **The global automotive industry**. [S. l.]: Wiley, 2015. (Automotive Series). ISBN 9781118802359. Available from:
<https://books.google.com.br/books?id=I9VZCgAAQBAJ>.

NIST, Big Data Public Working Group Definitions and Taxonomies Subgroup. **nist big data interoperability framework: volume 1, definitions**. en. [S. l.], Oct. 2015. nist sp 1500-1. DOI: 10.6028/NIST.SP.1500-1. Available from: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1500-1.pdf>. Visited on: 26 Nov. 2021.

OHKUMA, Toshiaki *et al.* Ankle-brachial index measured by oscillometry is predictive for cardiovascular disease and premature death in the japanese population: an individual participant data meta-analysis. en. **Atherosclerosis**, v. 275, p. 141–148, Aug. 2018. ISSN 0021-9150. DOI: 10.1016/j.atherosclerosis.2018.05.048. Available from: <https://www.sciencedirect.com/science/article/pii/S0021915018302910>. Visited on: 10 Nov. 2021.

ÖZEMRE, Murat; KABADURMUS, Ozgur. A big data analytics based methodology for strategic decision making. en. **Journal of Enterprise Information Management**, v. 33, n. 6, p. 1467–1490, May 2020. ISSN 1741-0398. DOI: 10.1108/JEIM-08-2019-0222. Available from: <https://www.emerald.com/insight/content/doi/10.1108/JEIM-08-2019-0222/full/html>. Visited on: 28 Mar. 2022.

PATEL, K M Archana; THAKRAL, Prateek. The best clustering algorithms in data mining. *In*: p. 2042–2046. DOI: 10.1109/ICCSP.2016.7754534.

PEARSON, Karl. Note on regression and inheritance in the case of two parents. **Proceedings of the Royal Society of London Series I**, v. 58, p. 240–242, Jan. 1895. ADS Bibcode: 1895RSPS...58..240P. Available from:
<https://ui.adsabs.harvard.edu/abs/1895RSPS...58..240P>. Visited on: 2 May 2022.

PEDREGOSA, F. *et al.* Scikit-learn: machine learning in python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

PFEIFFER, Sabine. The vision of “industrie 4.0” in the making—a case of future told, tamed, and traded. en. **NanoEthics**, v. 11, n. 1, p. 107–121, Apr. 2017. ISSN 1871-4757, 1871-4765. DOI: 10.1007/s11569-016-0280-3. Available from:
<http://link.springer.com/10.1007/s11569-016-0280-3>. Visited on: 9 Nov. 2021.

PILLONI, Virginia. How data will transform industrial processes: crowdsensing, crowdsourcing and big data as pillars of industry 4.0. en. **Future Internet**, v. 10, n. 3, p. 24, Mar. 2018. Number: 3 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/fi10030024. Available from: <https://www.mdpi.com/1999-5903/10/3/24>. Visited on: 2 Dec. 2021.

PINTO, Rafael Coimbra; ENGEL, Paulo Martins. A fast incremental gaussian mixture model. **PloS one**, Public Library of Science San Francisco, CA USA, v. 10, n. 10, e0139931, 2015.

QI, Geqi *et al.* Leveraging longitudinal driving behaviour data with data mining techniques for driving style analysis. **IET intelligent transport systems**, Wiley Online Library, v. 9, n. 8, p. 792–801, 2015.

RAWASHDEH, Mohammad; RALESCU, Anca. Fuzzy cluster validity with generalized silhouettes. *en*, p. 8, 2012.

REDDY, Chandan K.; VINZAMURI, Bhanukiran. A survey of partitional and hierarchical clustering algorithms. *In*: AGGARWAL, Charu C.; REDDY, Chandan K. (Eds.). **Data Clustering**. 1. ed. [S. l.]: Chapman and Hall/CRC, Sept. 2018. P. 87–110. ISBN 978-1-315-37351-5. DOI: 10.1201/9781315373515-4. Available from: <https://www.taylorfrancis.com/books/9781315362786/chapters/10.1201/9781315373515-4>. Visited on: 6 Apr. 2022.

REFAEILZADEH, Payam; TANG, Lei; LIU, Huan. Cross-validation. *In*: LIU, Ling; ÖZSU, M. Tamer (Eds.). **Encyclopedia of Database Systems**. New York, NY: Springer New York, 2016. P. 1–7. ISBN 978-1-4899-7993-3. DOI: 10.1007/978-1-4899-7993-3_565-2. Available from: http://link.springer.com/10.1007/978-1-4899-7993-3_565-2. Visited on: 24 Feb. 2022.

RIASANOW, Tobias; GALIC, Gabriela; BÖHM, Markus. Digital transformation in the automotive industry: towards a generic value network, 2017.

RIBEIRO, Rhubens Ewald Moura; ABREU, Cecília Rochele Silva de. inovação em sistemas de produção na era da indústria 4.0. *pt*, p. 164, 2020.

ROOSTA, Seyed H. **Parallel processing and parallel algorithms: theory and computation**. [S. l.]: Springer Science & Business Media, Dec. 2012. Google-Books-ID: oyPUBwAAQBAJ. ISBN 978-1-4612-1220-1.

ROSENBERG, Andrew; HIRSCHBERG, Julia. V-measure: a conditional entropy-based external cluster evaluation measure. *In*: p. 410–420. Available from: <https://aclanthology.org/D07-1043>. Visited on: 24 Feb. 2022.

SCALA JUNIOR, Newton La. Emissão de dióxido de carbono em solos de áreas de cana-de-açúcar sob diferentes estratégias de manejo. *In*. Available from: https://fapesp.br/eventos/2013/09/conclima/12/Newton_La_Scala.pdf.

SCHEINER, Nicolas *et al.* A multi-stage clustering framework for automotive radar data. *In*: p. 2060–2067. DOI: 10.1109/ITSC.2019.8916873.

SCHUBERT, Eugen *et al.* Clustering of high resolution automotive radar detections and subsequent feature extraction for classification of road users. *In*: p. 174–179. ISSN: 2155-5753. DOI: 10.1109/IRS.2015.7226315.

AL-SHBOUL, Bashar; MYAENG, Sung-Hyon. Initializing k-means using genetic algorithms. en, p. 6, 2009.

SINAGA, Kristina P.; YANG, Miin-Shen. Unsupervised k-means clustering algorithm. en. **IEEE Access**, v. 8, p. 80716–80727, 2020. ISSN 2169-3536. DOI: 10.1109/ACCESS.2020.2988796. Available from: <https://ieeexplore.ieee.org/document/9072123/>. Visited on: 25 Jan. 2022.

SINGH, Kanwar Bharat; ARAT, Mustafa Ali. Deep learning in the automotive industry: recent advances and application examples. **arXiv preprint arXiv:1906.08834**, 2019.

SINGHAL, Swasti; JENA, Monika. A study on weka tool for data preprocessing, classification and clustering. en. **Classification and Clustering**, v. 2, n. 6, p. 4, 2013.

STOLZ, Martin *et al.* High resolution automotive radar data clustering with novel cluster method. *In*: p. 0164–0168. ISSN: 2375-5318. DOI: 10.1109/RADAR.2018.8378550.

STONE, M. Cross-validators choice and assessment of statistical predictions. **Journal of the Royal Statistical Society. Series B (Methodological)**, v. 36, n. 2, p. 111–147, 1974. Publisher: [Royal Statistical Society, Wiley]. ISSN 0035-9246. Available from: <https://www.jstor.org/stable/2984809>. Visited on: 24 Feb. 2022.

THEISLER, Andreas *et al.* Predictive maintenance enabled by machine learning: use cases and challenges in the automotive industry. en. **Reliability Engineering & System Safety**, v. 215, p. 107864, Nov. 2021. ISSN 09518320. DOI: 10.1016/j.ress.2021.107864. Available from: <https://linkinghub.elsevier.com/retrieve/pii/S0951832021003835>. Visited on: 25 Jan. 2022.

TOMAŠEV, Nenad; RADOVANOVIĆ, Miloš. Clustering evaluation in high-dimensional data. *In*: **Unsupervised Learning Algorithms**. Ed. by M. Emre Celebi and Kemal Aydin. Cham: Springer International Publishing, 2016. P. 71–107. ISBN 978-3-319-24211-8. DOI: 10.1007/978-3-319-24211-8_4. Available from: https://doi.org/10.1007/978-3-319-24211-8_4.

VIDGEN, Richard; SHAW, Sarah; GRANT, David B. Management challenges in creating value from business analytics. en. **European Journal of Operational Research**, v. 261, n. 2, p. 626–639, Sept. 2017. ISSN 03772217. DOI: 10.1016/j.ejor.2017.02.023. Available from: <https://linkinghub.elsevier.com/retrieve/pii/S0377221717301455>. Visited on: 5 Nov. 2021.

WANG, Lu *et al.* Compositional data analysis of regional geochemical data in the lhasa area of tibet, china. en. **Applied Geochemistry**, v. 135, p. 105108, Dec. 2021. ISSN 0883-2927. DOI: 10.1016/j.apgeochem.2021.105108. Available from: <https://www.sciencedirect.com/science/article/pii/S0883292721002390>. Visited on: 10 Nov. 2021.

WANG, Wei *et al.* Vehicle trajectory clustering based on dynamic representation learning of internet of vehicles. **IEEE Transactions on Intelligent Transportation Systems**, v. 22, n. 6, p. 3567–3576, June 2021. Conference Name: IEEE Transactions on Intelligent Transportation Systems. ISSN 1558-0016. DOI: 10.1109/TITS.2020.2995856.

WANG, Xiangyu; WANG, Haixing. Driving Behavior Clustering for Hazardous Material Transportation Based on Genetic Fuzzy C-Means Algorithm. **IEEE Access**, v. 8, p. 11289–11296, 2020. DOI: 10.1109/ACCESS.2020.2964648.

WANG, Xinyi *et al.* Making the right business decision: forecasting the binary npd strategy in chinese automotive industry with machine learning methods. en. **Technological Forecasting and Social Change**, v. 155, p. 120032, June 2020. ISSN 00401625. DOI: 10.1016/j.techfore.2020.120032. Available from: <https://linkinghub.elsevier.com/retrieve/pii/S0040162519317949>. Visited on: 25 Jan. 2022.

WARD, Jonathan Stuart; BARKER, Adam. Undefined by data: a survey of big data definitions. en. **arXiv:1309.5821 [cs]**, Sept. 2013. arXiv: 1309.5821. Available from: <http://arxiv.org/abs/1309.5821>. Visited on: 30 Mar. 2022.

WARING, S.P. **Taylorism transformed: scientific management theory since 1945**. [S. l.]: University of North Carolina Press, 2016. ISBN 9781469619644. Available from: <https://books.google.com.br/books?id=vwo1DgAAQBAJ>.

WESSEL, Maxwell. How big data is changing disruptive innovation. **Harvard Business Review**, Jan. 2016. Section: Disruptive innovation. ISSN 0017-8012. Available from: <https://hbr.org/2016/01/how-big-data-is-changing-disruptive-innovation>. Visited on: 5 Nov. 2021.

YANG, Lingzhi *et al.* A new data preprocessing technique based on feature extraction and clustering for complex discrete temperature data. en. **Procedia Computer Science**, v. 129, p. 78–80, Jan. 2018. ISSN 1877-0509. DOI: 10.1016/j.procs.2018.03.050. Available from: <https://www.sciencedirect.com/science/article/pii/S1877050918302734>. Visited on: 3 Feb. 2022.

YI, Dewei *et al.* Trajectory clustering aided personalized driver intention prediction for intelligent vehicles. **IEEE Transactions on Industrial Informatics**, v. 15, n. 6, p. 3693–3702, June 2019. Conference Name: IEEE Transactions on Industrial Informatics. ISSN 1941-0050. DOI: 10.1109/TII.2018.2890141.

ZHANG, Xuegong; LI, Yanda. Self-organizing map as a new method for clustering and data analysis. *In*: v. 3, 2448–2451 vol.3. DOI: 10.1109/IJCNN.1993.714219.

ZHAO, Qinpei; FRÄNTI, Pasi. wb-index: a sum-of-squares based index for cluster validity. en. **Data & Knowledge Engineering**, v. 92, p. 77–89, July 2014. ISSN 0169023X. DOI: 10.1016/j.datak.2014.07.008. Available from:

<https://linkinghub.elsevier.com/retrieve/pii/S0169023X14000676>. Visited on: 20 Feb. 2022.

ZHOU, Tuqiang; ZHANG, Junyi. Analysis of commercial truck drivers' potentially dangerous driving behaviors based on 11-month digital tachograph data and multilevel modeling approach. en. **Accident Analysis & Prevention**, v. 132, p. 105256, Nov. 2019. ISSN 0001-4575. DOI: 10.1016/j.aap.2019.105256. Available from: <https://www.sciencedirect.com/science/article/pii/S0001457519304737>. Visited on: 4 May 2022.