# **1 Introduction (Business Understanding)**

Steel has always been one of the biggest industries in the world. A lot of companies prefer to go this path of business because of the amount of money that it generates. One of the negative parts that implies doing this business is the amount of energy that it consumes and that is going to be the focus of this project. We will use a dataset of the energy consumption of an industry that is stored on the website of the Korea Electric Power Corporation. The information gathered is from the DAEWOO Steel Co. Ltd in Gwangyang, South Korea. With this dataset, we expect to apply cluster algorithms for the usage of kw/h and find patterns in the consumption.

# **2 Motivation**

The main motivation for this assignment is to acquire skills and deep knowledge in the use of different clustering algorithms because it is a versatile tool that we can apply in various fields in the future. It helps us understand and organize data, making it easier to gain insights, find patterns, and improve decisions. We also seek to understand how these algorithms perform and learn how changing the different hyperparameters can improve the results. This will be carried out with experimentation and finally, we will compare the results obtained.

We chose to use a clustering technique for this project in order to group the power consumption. We decided to focus on the amount of lagging power in order to cluster the groups and decide how much power it needed to be considered as low, medium, and high load power types. Data scientists and others use clustering to gain important insights from data by observing what groups (or clusters) the data points fall into when they apply a clustering algorithm to the data. By definition, unsupervised learning is machine learning that searches for patterns in a data set with no pre-existing labels and a minimum of human intervention. Clustering can also be used for anomaly detection to find data points that are not part of any cluster or outliers.

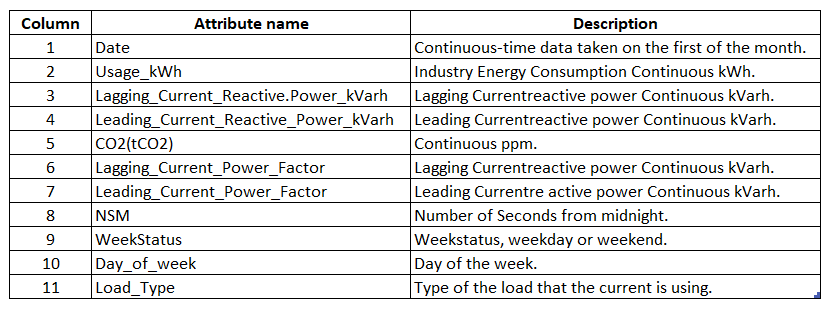
Furthermore, the project aims to find the algorithm that best groups the data in this dataset. Through this project, our group hopes to gain a deep understanding of how to apply machine learning techniques in practice and how to use them to solve real-world problems

# **3. Data**

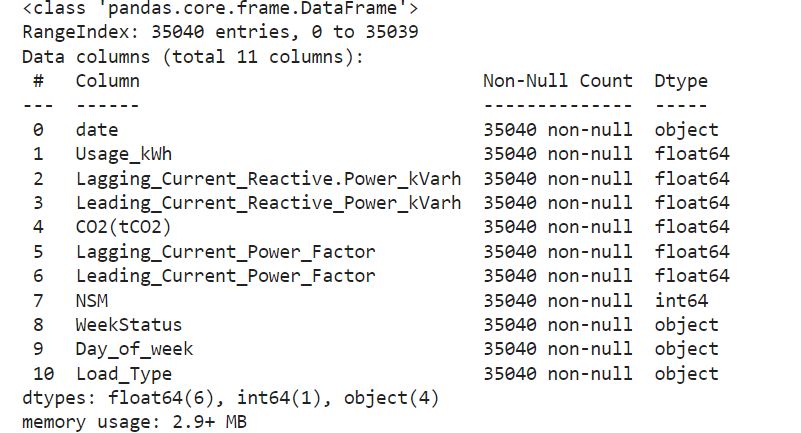
# **3.1. Overview**

For this dataset, we have an amount of 11 features.

The information gathered is from the DAEWOO Steel Co. Ltd in Gwangyang, South Korea.

Table 1 - Dataset Dictionary

This dataset has 35040 entries and 11 columns. There are no missing or duplicate values. Exploratory data analysis was performed with the help of libraries such as Pandas, Seaborn and Matplotlib.



As we can see, most of our data have numerical features, but for the columns Week Status, Day of the Week and Load Type, we are going to categorize the data using One Hot Encoder for Week Status and Ordinal Encoding for Day of the week and Load Type, this because the two variables have an inherent order or hierarchy.

“Ordinal Encoding is best suited for ordinal categorical variables, where there’s a logical order or ranking to the categories. “ Wohlwend, B. (2023).

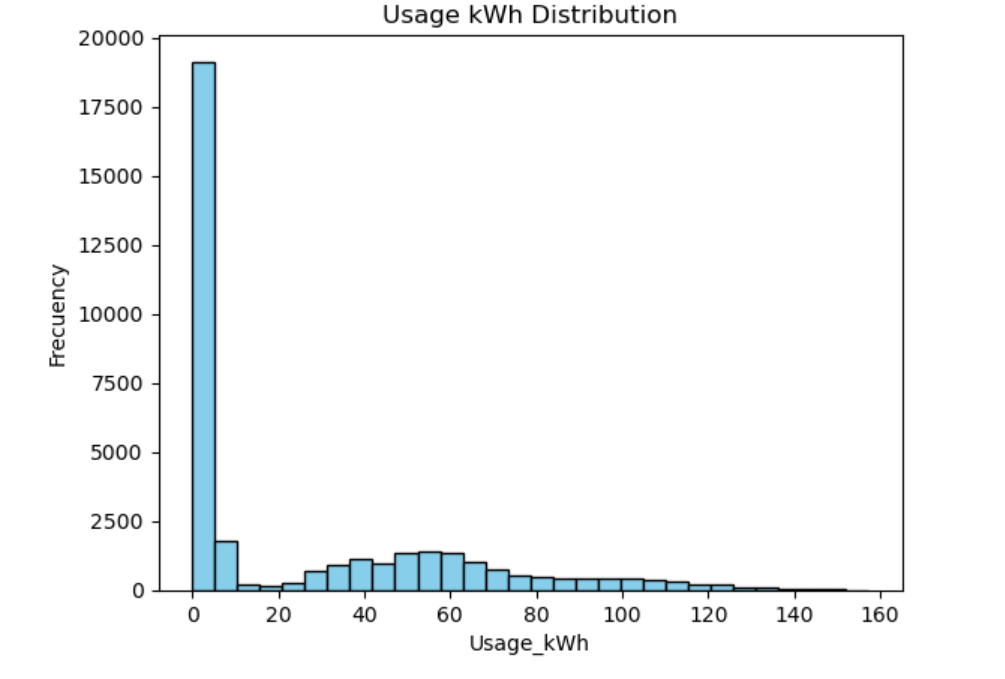
# **3.1 Data Preparation (Characterization and normalization of data)**

Normalization is a technique often applied as part of data preparation for machine learning. The goal of normalization is to change the values of numeric columns in the dataset to a common scale, without distorting differences in the ranges of values. For machine learning, every dataset does not require normalization. It is required only when features have different ranges.

In this case, we have our dataset based on kilowatts, some of them are kwH and others are in KVarH. KVarH is the wasted energy in starting an inductive load, while kWh is the actual energy used. Since it’s a different measure scale, we will use the StandardScalar normalization. Variables that are measured at different scales do not contribute equally to the model fitting and model learned to function and might end up creating a bias. StandardScaler removes the mean and scales each feature/variable to unit variance. This operation is performed feature-wise in an independent way. StandardScaler can be influenced by outliers (if they exist in the dataset) since it involves the estimation of the empirical mean and standard deviation of each feature.

# **3.2 EDA**

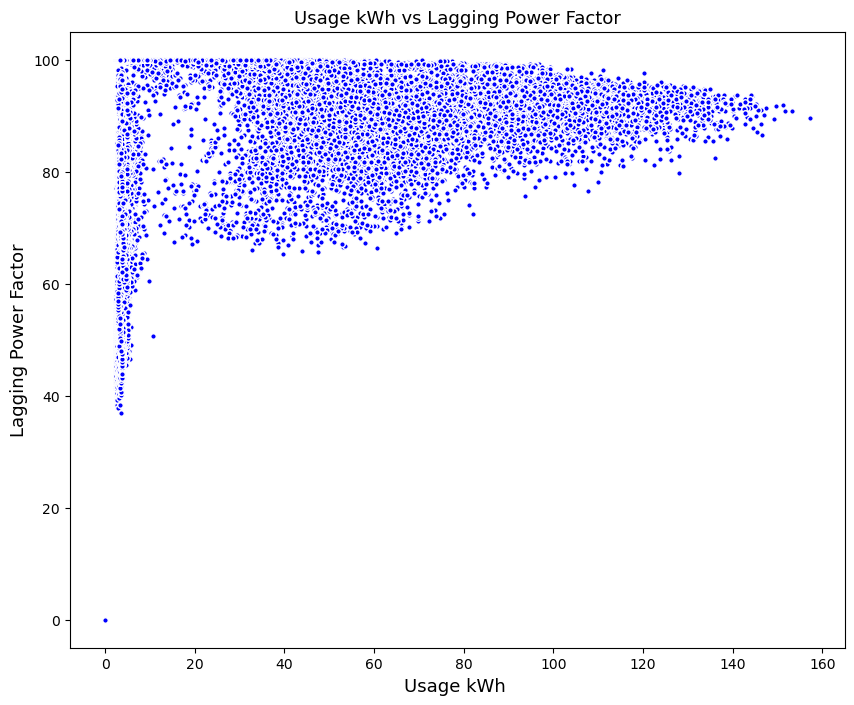
In order to look deeper in our data, we will make some x-y relations to see how the data is behaving towards our feature ‘Usage\_kwh’.



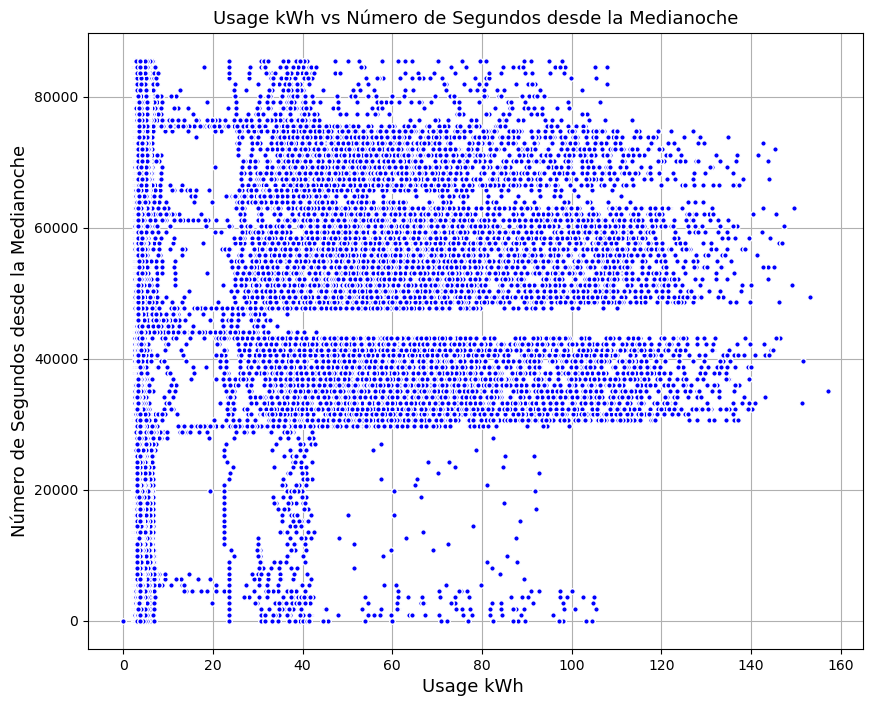
Analyzing the distribution of ‘Usage\_kWh’, we can see that our data is skewed to the left side, most of the values are between 0-15. We can see that there is a pattern where some values are between 25-75, then from 80-120 the pattern becomes lower. We also have some outliers that are from 140-160. Analyzing the distribution of your data is important because it can have a significant impact on the results of your analysis. Understanding the distribution of your data can help you choose the right statistical test, identify outliers, check for normality, and visualize the data. By understanding the distribution of your data, you can ensure that your results are accurate, reliable, and valid.



In this ‘Usage\_kW’- ‘Laggin\_Power\_kVah’ we can see that there is a pattern. This means that the correlation between these variables is positive. The greater amount of ‘Usage\_kW’ means that ‘Lagging\_Pwer\_kVah’ will increase. This is why the pattern is created. A positive correlation is a relationship between two variables that tend to move in the same direction. A positive correlation exists when one variable tends to decrease as the other variable decreases, or one variable tends to increase when the other increases.



In this ‘Usage\_kW’ - ‘Lagging\_Power\_Factor’ relation we can see that there is a little correlation between them. ‘Lagging\_Power\_Factor’ does not depend on the value of ‘Usage\_kW’, but it makes a little impact on it. This means that even though they move in the same direction, the values of one variable will have a minimum impact on the direction of the other. In other words, as one variable moves one way, the other moves around having a little impact on its direction.



In this ‘Usage\_kWH’ - ‘NSM’ we can see that there is no relation between these variables so that means they do not have any correlation at all. This means the two variables moved in opposite directions. Zero or no correlation. A correlation of zero means there is no relationship between the two variables. In other words, as one variable moves one way, the other moves in another unrelated direction.

We choose to make scatterplots for the EDA because These graphs display symbols at the X, Y coordinates of the data points for the paired variables. Scatterplots are also known as scattergrams and scatter charts. The pattern of dots on a scatterplot allows you to determine whether a relationship or [correlation](https://statisticsbyjim.com/glossary/correlation/) exists between two continuous variables. If a relationship exists, the scatterplot indicates its direction and whether it is a linear or curved relationship.

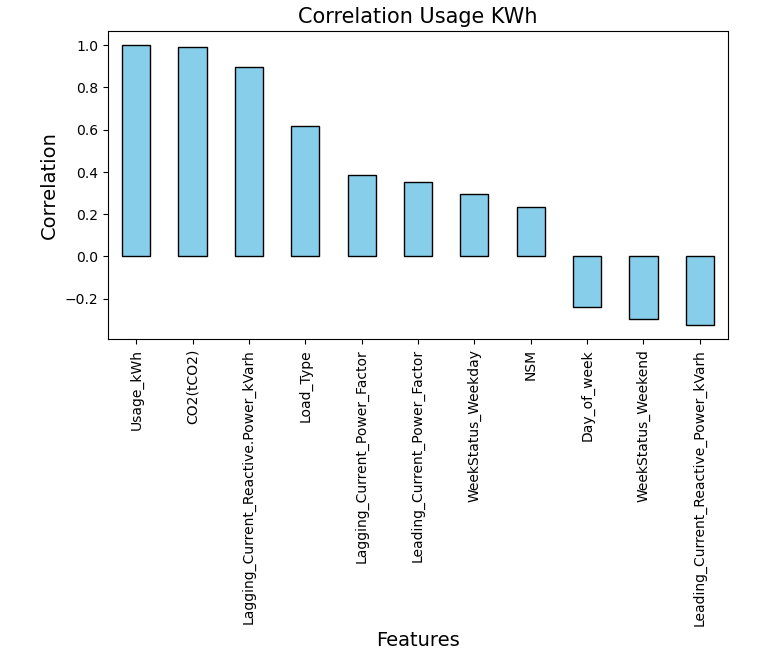
# **3.3 Correlation**

It can be useful in data analysis and modeling to better understand the relationships between variables. The statistical relationship between two variables is referred to as their correlation.

A correlation could be positive, meaning both variables move in the same direction, or negative, meaning that when one variable’s value increases, the other variables’ values decrease. Correlation can also be neutral or zero, meaning that the variables are unrelated.

* Positive Correlation: both variables change in the same direction.
* Neutral Correlation: No relationship in the change of the variables.
* Negative Correlation: variables change in opposite directions.

Having a look into our variables, we will see the correlation they have with ‘Usage kWh’. This will help us know which variables have a major impact on it. With this information we will be able to select features and list them by a big to a lesser importance in our dataset. This also works in being able to decide if a variable is not necessarily for the rest of the analysis. It’s logical that when a variable does not add or subtract information into our work, to delete it and move on without it.



In this correlation analysis, we can see the top 3 with more correlation towards ‘Usage\_kWh’. C02, Lagging\_Current\_Power and Load\_Type.These variables are the ones who have the most impact on Usage\_kWh, which means that they are the ones making its pattern.

On the other hand, we have other variables with less impact. Lagging\_Current, Leading\_Current, WeekStatus\_ and NSM have some correlation but not the same as the top 3. This does not mean that this variable doesn’t matter. It does matter when it comes to a deeper analysis.

And last but not least, we have the variables which have a negative correlation. These variables have a negative impact in Usage\_kWh which means they do not form the positive pattern. If this variable increases, it means our variable will go in a different direction.

**4 Machine Learning Models - Clustering**

It is basically a collection of objects on the basis of similarity and dissimilarity between them. For example The data points in the graph below clustered together can be classified into one single group. The method of identifying similar groups of data in a large dataset is called clustering or cluster analysis. It is one of the most popular clustering techniques in data science used by data scientists. Entities in each group are comparatively more similar to entities of that group than those of the other groups.

It reveals subgroups in the available heterogeneous datasets such that every individual cluster has greater homogeneity than the whole. In simpler words, these clusters are groups of like objects that differ from the objects in other clusters. In clustering, the machine learns the attributes and trends by itself without any provided input-output mapping. The clustering algorithms extract patterns and inferences from the type of data objects and then make discrete classes of clustering them suitably. Selecting appropriate analysis methods ensures that the conclusions drawn from the data are reliable and free from biases or errors. For instance, using inferential statistical tests to assess the significance of relationships or differences in data can provide a level of confidence in the findings.

**4.1. Feature selection**

Feature Selection is the method of reducing the input variable to your model by using only relevant data and getting rid of noise in data.

In the machine learning process, feature selection is used to make the process more accurate. It also increases the prediction power of the algorithms by selecting the most critical variables and eliminating the redundant and irrelevant ones. This is why feature selection is important.

It is the process of automatically choosing relevant features for your machine learning model based on the type of problem you are trying to solve. We do this by including or excluding important features without changing them. It helps in cutting down the noise in our data and reducing the size of our input data.

In this scenario, since we are applying clusters we decided to check between the variables in our top 3 in correlation.

CO2: Even though this feature is the one with more positive correlation, after doing some experiments with tables and some plots, we saw that the values do not help us showing the clusters of the usahe\_kWh.

Lagging\_Current\_Reactive\_Power\_kVaH: This is the top 2 features with most correlation. We did x-y relations to see how the data reacts according to the usage\_kWh level and even though this feature is escalated in a different scale, it was no problem for us since Standard Scale helped us solve this issue. In the end we decide to give it a go to this feature and choose it to continue with the analysis of the project.

Load\_Type: We notice that this variable is categorical and is going to make it hard for us to make a deeper analysis due to the lack of values. The clusters were too exact and didn't leave any chance to notice the density or any other behavior.

# **4.2. Model selection**

Clustering techniques consider data tuples as objects. They partition the objects into groups, or clusters, so that objects within a cluster are “similar” to one another and “dissimilar” to objects in other clusters. Similarity is commonly defined in terms of how “close” the objects are in space, based on a distance function. The “quality” of a cluster may be represented by its diameter, the maximum distance between any two objects in the cluster.

We decide to use a cluster model to see some patterns according to the level of energy consumption. With the cluster we can get an idea of how the values group towards finding different sections of the clusters. The good thing about clusters is that they can draw a line to classify the data according to the numbers of clusters that we can form.

# **4.2.1. K-Means**

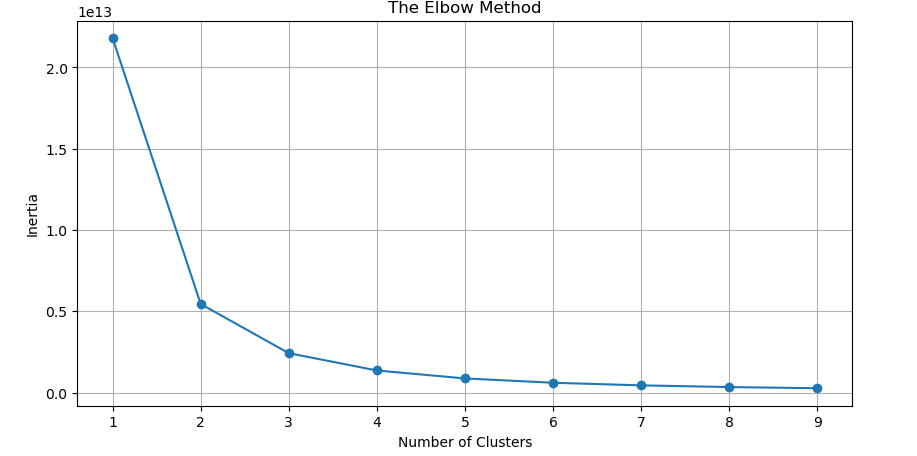
K-means clustering, a part of the unsupervised learning family in AI, is used to group similar data points together in a process known as clustering. Clustering helps us understand our data in a unique way by grouping things together into clusters.

The main element of the algorithm works by a two-step process called expectation maximization. The expectation step assigns each data point to its nearest centroid. Then, the maximization step computes the mean of all the points for each cluster and sets the new centroid.

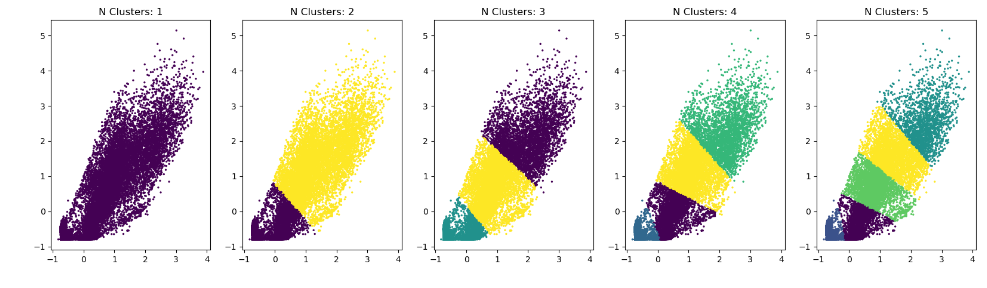
First, we will use the elbow method. The elbow method is a heuristic used in determining the number of clusters in a data set. The method consists of plotting the explained variation as a function of the number of clusters and picking the elbow of the curve as the number of clusters to use.



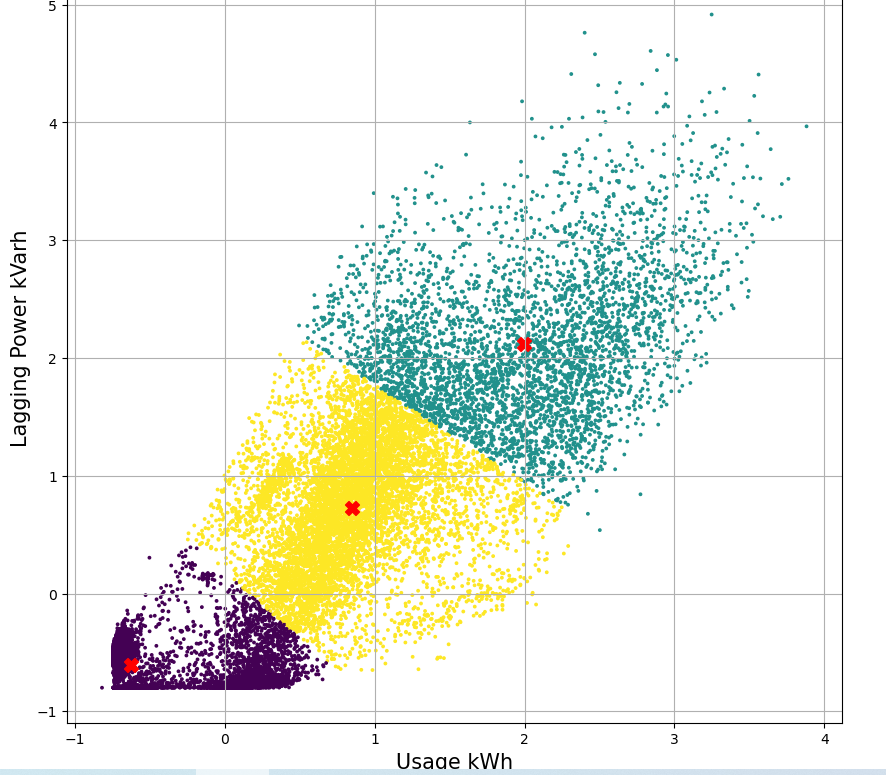
The result:



With this result, we agreed to use only 3 clusters. We can see that the difference between 3 and 4 clusters starts to decrease. This is the reason why we decided to stop at clusters = 3.



In this result, we can compare the differences according to the numbers of clusters. With 3 clusters the centroids of the data are well divided between the plots.

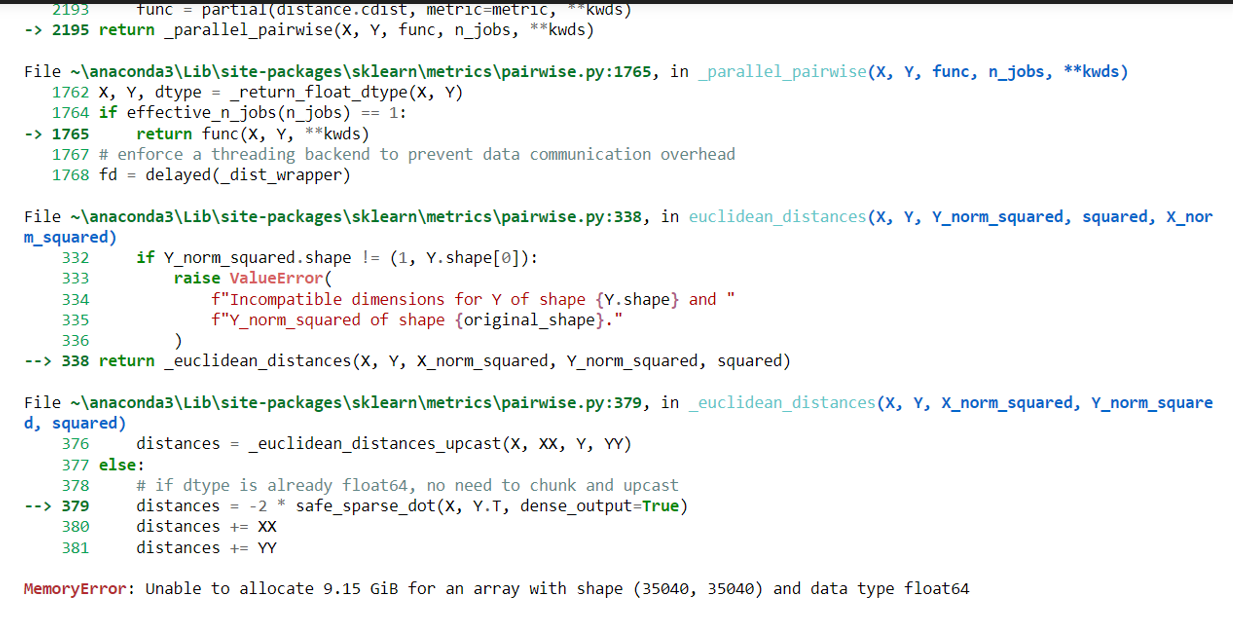


# **4.2.2. K-Medoids- Day Analysis**

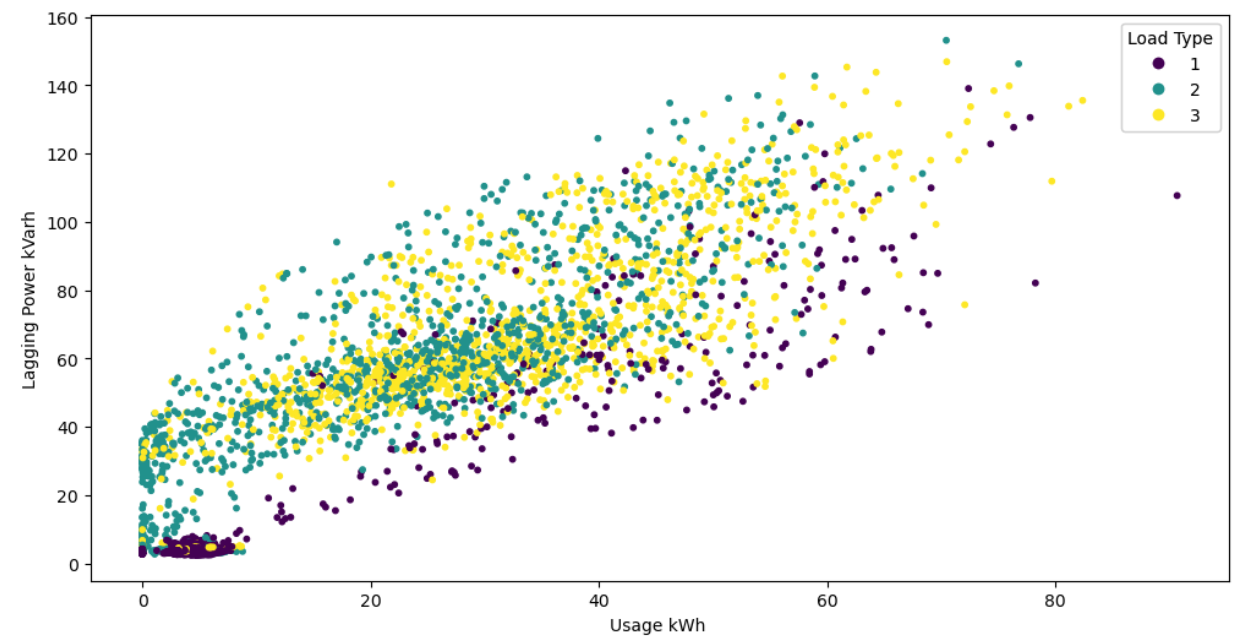
K-medoids is a clustering algorithm very related to the k-means algorithm, both work with partitions and both try to minimize the distance between points that would be added to a group. The difference is that k-means is sensitive to outliers but k-medoids is not.

“K-medoids clustering is a variant of K-means that is more robust to noises and outliers. Instead of using the mean point as the center of a cluster, K-medoids uses an actual point in the cluster to represent it.” Mannor, S., Jin, X., Han, J. and Zhang, X. (2011).

# In this case we tried to use this algorithm for all the data set but the computer was not able to process all the information, we tried several times but we got the same error all the time:

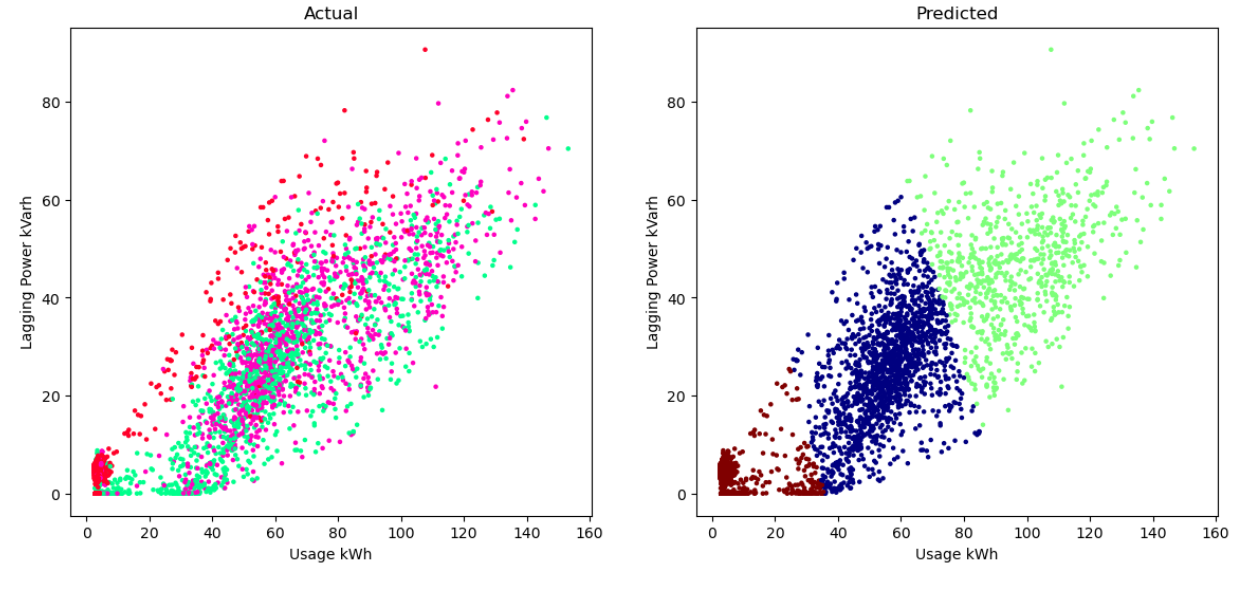


That is why we decided to split the data for the analysis just for this particular algorithm. We based the split on the days of the week.

Plotting the data for Monday we have the following graphic:

We can see some differentiation of the Light Load Type(purple). The Medium and Maximum Load are very mixed and it is difficult to find a pattern.

Now we apply the K-medoid algorithm, we initialize the number of clusters in 3 because we found that using the Elbow method in previous steps:

We can see how the 3 clusters were formed, we have the Light Load in brown, the middle Load in blue, and the High Load in green. At first sight, the clusters look good but we are going to evaluate them in the next steps.

# **4.2.3. DBSCAN**

“DBSCAN is a widely used density-based clustering algorithm. It groups together points that are closely packed together (points with many adjacent neighbors), while filtering out noise points that lie in low-density regions” ‌Yehoshua, D.R. (2023).

This algorithm depends on the density in the data set, in this case, our data does not exhibit clear density-based structures, therefore it is possible that the results are not good. However, we tried it as this is an experimentation process.

DBSCAN determines the density of an area to create clusters based on two hyperparameters:

* Epsilon: The radius of the neighborhood around a data point that specifies how close the points should be to each other to be considered a part of a cluster.
* Minimum Samples: the minimum number of data points required to form a dense region.

For this case, we start with the default hyperparameters of:

eps=0.5, min\_samples=5

# 

# 

# 

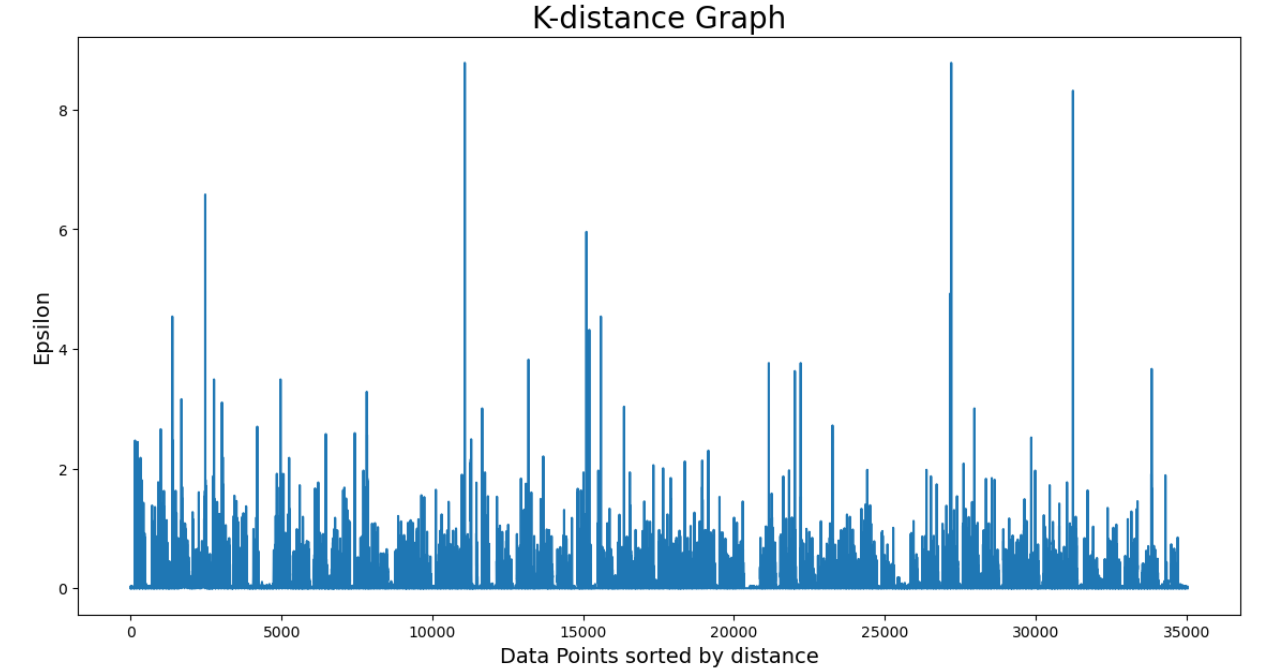
# 

# 

# 

# 

# As we can see, we just got very small clusters that are not very useful, so we proceed to use a Parameter Estimation - K distance graph in order to find the best epsilon for our algorithm. Also we can observe



We can observe that the ideal epsilon is between 1 and 2, we tried both options but using

eps=1 gave us the best results:

# 

We use min\_samples=30 in this case, because as Prado said, “Larger values are usually better for data sets with noise and will form more significant clusters. The minimum value for the minPoints must be 3, but the larger the data set, the larger the minPoints value that should be chosen.” Prado, K.S. (2019). We will evaluate this model in further steps.

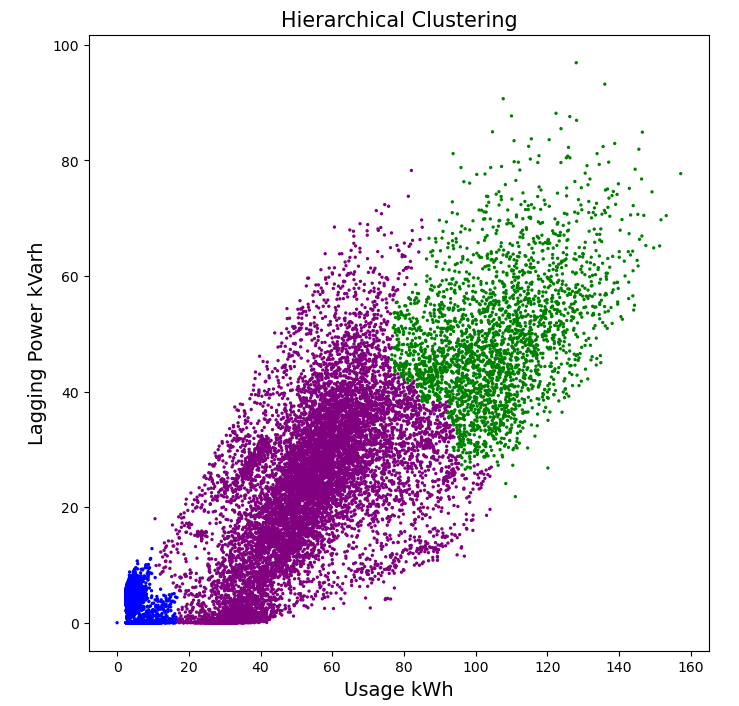
We can see that DBSCAN creates clusters based on varying densities, in this case it is difficult for the algorithm to deal with clusters of similar densities. “Also, as the dimension of data increases, it becomes difficult for DBSCAN to create clusters and it falls prey to the Curse of Dimensionality.” Sharma, A. (2020)

As DBSCAN may not be the appropriate algorithm in this case of consumption of energy, other clustering algorithms like hierarchical clustering might be more suitable and we are going to explain that one in the following section.

# **4.2.4.Hierarchical Clustering - Agglomerative Clustering**

Hierarchical clustering, also known as hierarchical cluster analysis, is an algorithm that groups similar objects into groups called clusters. The endpoint is a set of clusters, where each cluster is distinct from each other cluster, and the objects within each cluster are broadly similar to each other. Hierarchical clustering involves creating clusters that have a predetermined ordering from top to bottom. For example, all files and folders on the hard disk are organized in a hierarchy

In agglomerative or bottom-up clustering methods we assign each observation to its own cluster. Then, compute the similarity (e.g., distance) between each of the clusters and join the two most similar clusters. Finally, repeat steps 2 and 3 until there is only a single cluster left.



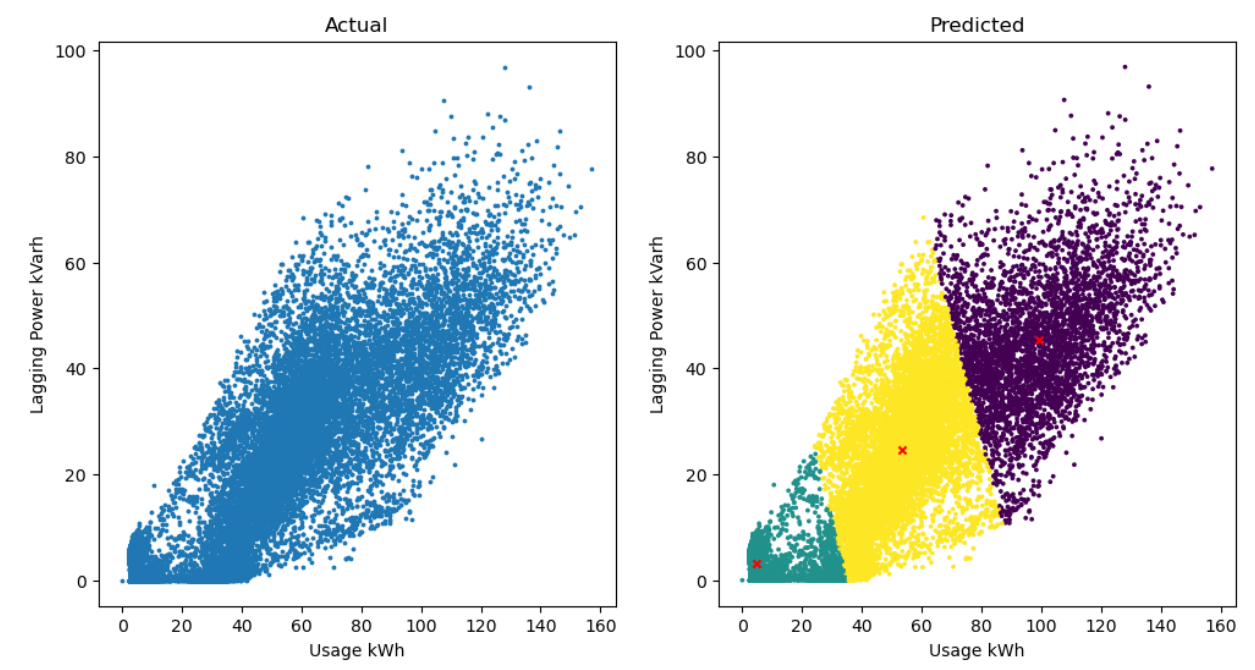
# **4.2.5. Fuzzy C-means clustering**

Fuzzy C-means works by grouping similar data points into clusters based on the similarities between them. Unlike other clustering algorithms, such as k-means, that assign each data point to a single cluster, Fuzzy C-Means assigns each data point a partial membership value that indicates its membership in each cluster.

It is a more flexible algorithm.

“As fuzzy methods determine which data belong to clusters, they often provide better results than definite methods. The Fuzzy C-Mean Cluster (FCM) is the most popular fuzzy method. Its simplicity is one of the positive features of the FCM method.” Seyed Emadedin Hashemi, Fatemeh Gholian-Jouybari, and Mostafa Hajiaghaei–Keshteli (2023).

In this case, we initialized the algorithm for 3 clusters, also we found the centers, and the results were plotted as follows:



We can see that 3 clear clusters are formed, also we see the centers of each cluster and we can notice that the three clusters show how the consumption of energy is lower when the lagging power is lower, and when the lagging power increases, the energy usage also increases.

At first sight, the clusters look much better than others but we are going to evaluate them in the next steps.

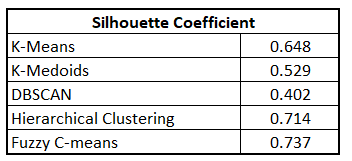
**4.3. Silhouette Coefficient - Validating Clustering Techniques**

As the next step, we applied a technique to find how well the clustering algorithm is. We use the silhouette score, or silhouette coefficient.

“Silhouette Coefficient or silhouette score is a metric used to calculate the goodness of a clustering technique. Its value ranges from -1 to 1.” Bhardwaj, A. (2020).

* 1: Means clusters are well apart from each other and clearly distinguished.
* 0: Means clusters are indifferent, or we can say that the distance between clusters is not significant.
* -1: Means clusters are assigned in the wrong way.

In the next table we can find the results for each alghoritm.



**4.4. Comparison and interpretation of the results.**

After doing our clustering methods, we can see the silhouette coefficient of each one. Typically, a silhouette score of 0.5 or higher is considered to indicate a reasonably good clustering. However, it is important to note that an ideal score is 1, and a score less than 0 indicates that the clustering is poor and the sample is probably assigned to the wrong cluster.

With our result, we can imply that the DBSCAN method is the worst. DBSCAN cannot cluster data sets well with large differences in densities, so we discard this method. The following are K-Medoid and K-means. The main disadvantage of K-Medoid algorithms is that it is not suitable for clustering non-spherical groups of objects. We did see in our plots that our clusters were not even close to a spherical group, so again we discard this one. On the other hand, for the K-means we can see that the coefficient result is 0.64. It’s not bad but we have better ones.

The results of Hierarchical and Fuzzy C-means are superior. They are closer to 1 so this means that clusters are well apart from each other and clearly distinguished. The key to interpreting a hierarchical cluster analysis is to look at the point at which any given pair of cards “join together” in the tree diagram. Cards that join together sooner are more similar to each other than those that join together later. The main purpose of fuzzy c-means clustering is the partitioning of data into collection clusters, where each data point is assigned a membership value for each cluster. The performance of the fuzzy c-means algorithm gives better performance than k-mean, both when using thresholding with mean and median methods. Better performance of fuzzy c-means requires additional time when compared to k-means.

We decided to stay with the Fuzzy C-means due to the silhouette coefficient value.

**5. Conclusion**

There are many different methods in order to analyze data and give a conclusion whether the situation demands it. In this case, having a discussion about the methods we decided to make a clusterization analysis. Our main goal was to cluster our data and have a look at our features. In this way we would better appreciate the relationships between variables and see in our visualizations how every different feature differentiates with each other according to our variable ‘Usage\_kWH’. We discovered rational behaviors, some of them were positive and other negatives. Something surprisingly that we noticed was that we didn’t have a neutral feature correlation in our data.

After deciding to use cluster methods, we had the task to look for the best one among this selection. We knew that our dataset was dense so we had an idea that the results of DBSCAN would not be that great. What surprised me was the results of the Fuzzy C-means. We thought that with the K-means method would be enough to prove and show a clean visualization between clusters, it didn’t have a bad silhouette coefficient result and the plots and centroids were good. After doing the Fuzzy C-means method we realized that this was the right path in order to continue with our analysis.

The main question for this project was which patterns can we find in the energy consumption? In the end we could see a high correlation in the lagging energy. We could cluster the lagging energy of this company in order to calculate and see how many kvarH of lagging energy will be low, medium or high energy is consumed.

**6. References.**

* www.kaggle.com. (n.d.). Steel Industry Energy Consumption. [online] Available at: https://www.kaggle.com/datasets/csafrit2/steel-industry-energy-consumption.
* ‌Wohlwend, B. (2023). Converting Categorical Data into Numerical Form: A Practical Guide for Data Science. [online] Medium. Available at: https://medium.com/@brandon93.w/converting-categorical-data-into-numerical-form-a-practical-guide-for-data-science-99fdf42d0e10#:~:text=Ordinal%20Encoding [Accessed 2 Nov. 2023].
* Mannor, S., Jin, X., Han, J., Jin, X., Han, J., Jin, X., Han, J. and Zhang, X. (2011). K-Medoids Clustering. Encyclopedia of Machine Learning, [online] pp.564–565. doi:https://doi.org/10.1007/978-0-387-30164-8\_426.
* ‌Yehoshua, D.R. (2023). DBSCAN: Density-Based Clustering. [online] Medium. Available at: https://medium.com/ai-in-plain-english/dbscan-density-based-clustering-aaebd76e2c8c [Accessed 3 Nov. 2023].
* ‌Prado, K.S. do (2019). How DBSCAN works and why should we use it? [online] Medium. Available at: https://medium.com/towards-data-science/how-dbscan-works-and-why-should-i-use-it-443b4a191c80 [Accessed 3 Nov. 2023].
* Sharma, A. (2020). How Does DBSCAN Clustering Work? | DBSCAN Clustering for ML. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2020/09/how-dbscan-clustering-works/.
* ‌Seyed Emadedin Hashemi, Fatemeh Gholian-Jouybari and Mostafa Hajiaghaei–Keshteli (2023). A fuzzy C-means algorithm for optimizing data clustering. Expert Systems with Applications, 227, pp.120377–120377. doi:https://doi.org/10.1016/j.eswa.2023.120377.
* ‌Bhardwaj, A. (2020). Silhouette Coefficient : Validating clustering techniques. [online] Medium. Available at: https://towardsdatascience.com/silhouette-coefficient-validating-clustering-techniques-e976bb81d10c.

‌

**7. Individual contribution**

**7.1. Ayllin Bukovic**

Initially my contribution to this project was to look for the data set in different platforms and to perform exploratory data analysis on them in order to test which one was more appropriate to fulfill the requirements of this CA. Once I found the right one I discussed with my teammate on which should be our approach and main goal. At this point, we decided to focus on finding patterns for the energy consumption of this industry. I performed the EDA, where my goal was to understand the structure of the database better and extract relevant information that could help us make informed decisions. I also carried out a descriptive analysis of the different variables of the database, which allowed me to obtain information about the relationship between the different variables. Subsequently, I went to the data preparation stage, where I transformed the categorical variables into numerical ones to be able to include them in the algorithm, for this case I used One-Hot-Encoder as well as Ordinal Codification. On the other hand, the standardization of the data was applied, and we saw that the results were not significantly affected using standardized data.

At this stage, I developed a correlation matrix to review what variables were related to each other and in which grade.

After this, Already having a clean base, with the relevant data, I proceeded to apply different cluster algorithms in order to find patterns related to the consumption of energy and reach our main goal. taking into account that K-means is the most commonly used clustering algorithm, I decided to start by applying it. At the same time, my teammate also applied the same algorithm so we could compare our results and verify that our results were correct.

Thereafter, I used the elbow method in order to find the optimal number of clusters. Moreover, I also used different models such as K-Medoids, DBSCAN, and Fuzzy C-means, in order to experiment with and understand the behavior of the algorithms with different hyperparameters. In the specific case of the DBSCAN, I also created a K-distance graph to find the optimal epsilon value.

Once I applied all these models, I evaluated their performance by applying the Silhouette Coefficient method for each algorithm. The goal of this was to understand and numerically determine which of our algorithms was a better fit for the clusters. I developed a table with the results to help visualize the differences in a clearer way. Therefore I was able to state at this point that Fuzzy C-Means was the best option for this case and the Hierarchical algorithm was the second.

Finally, to visualize each result, I also created graphs showing the clusters and comparing them to the original data. In these graphs, we can see the clear difference between the actual and predicted clusters. Once the entirety of the project was finished I reflected my Python code one last time in a GitHub Repository as well as my general results in this report.

(495 words)

**7.2. Eduardo Saldivar**

In this project I was in charge of focusing on the main question, collecting data and making some clusters methods like K-Means and Hierarchical methods. After revising the data with my teammate and checking the data cleaning I was ready to make some assumptions about our project and get ready to make the cluster methods.

For the K-means, We decided to make the elbow method to check the clusters and their numbers. After revising the plot, it was really sure that the number of clusters to put in the hyperparameter was 3, but anyways we decided to make the plots of all of the clusters. After the analysis I made sure to look for more information about K-mean and its cons and pros. I considered that it is important to understand and research more information about the subjects we are practicing in class, this will help to gather more knowledge in the way of doing the projects and being able to understand and explain step by step our logic towards the goal of the analysis. For hierarchical clusters I did the same, and the result was similar.

After finishing our analysis, we did the silhouette coefficient for each of our clusters methods to be able to differentiate the quality and visibility of our clusters. I had to research quite a bit to make sure to understand what the silhouette coefficient tells us in every method, but at the end we made it simple. 1 is the best value, 0.5 is medium and 0 is the worst possible value.

Having already the result of the silhouette coefficient, I did an analysis of the result. We found out that even when the visualization is good, sometimes the clusters separation and visibility is not that good, this happened with the K-Means method. In other cases, we saw that the data is important in order to know which method to use directly, like it happened with the DBSCAN, DBSCAN doesn’t work very nice with data have density, and in this case, our dataset had quite a bit of it so that means that the DBSCAN was the worst of all. By our surprise, the Fuzzy C-means was the best for us and we found out that it does have a similarity with the K-Means, but it’s important to know when and where to use each method. I decided to do a bit of research to understand the importance of knowing the details and the pros and cons of each method.

This project was a challenge to us because we didn’t know a lot of things about clustering methods, but after realizing this project, we feel more comfortable with the subject and we know the most important details in order to achieve more projects related to this topic in the future.

**‌**