**Efficiency In Energy Consumption of IoT Devices**

**Abstract**—

The word IOT was first suggested by Kevin Ashton in 1999. The "Internet of Things (IoT)" describes a kind of network that interconnects various devices with the help of the Internet. IoT assists to transmit data among devices, tracing and monitoring devices, and other things. IOT makes objects 'smart' by allowing them to transmit data and automate tasks, without any physical interference. The IoT has a variety of applications, including social security. Different restore and post-operative information should be screened. This is seen as the future that will transform our interactions with objects and with our own life. However, with such a large network and automation of the devices, still, the energy consumed by the devices will be a chief concern in the IoT. The purpose of this paper is to propose a new method of clustering algorithm which is a combination of hierarchical clustering and k-centroid clustering algorithm to save energy consumption and make it efficient.

***Keywords****–* Internet of Things (IoT), Interference, Clustering algorithm, Hierarchical Clustering Algorithm (HCA), Machine learning, Linkage method.

1. INTRODUCTION

Internet of Things (IoT) is a new concept within the information and communication technology studies that indicates that any creature (human, animal, or object) can send and receive data through communication networks, like the internet or intranet platform. The Internet of Things is a logical step to creating the global network, which initially allowed static pages to be accessed. Today, it contributes to artificial intelligence integrated systems that respond to the environment to permanent changes and human needs. The IoT (Internet of Things) has the potential to change our world in many ways, but it also comes with its own set of challenges. One of the foremost critical problems and challenges on the Internet of Things is reducing power consumption. Hierarchical clustering is one of the most popular cluster analysis techniques in unsupervised machine learning. It builds a hierarchical cluster tree to group a set of data points by calculating distances between them. Hierarchical clustering has been widely used in many fields such as air pollution [1], transportation [2], medical [3].

1. METHODOLOGY

**Data Generation and Preprocessing**

**data.head()**

| device\_name | min\_consumption | max\_consumption | device\_age |
| --- | --- | --- | --- |
| bulb | 100.0 | 120.0 | 3 |
| LED TV 42 INCH | 58.0 | 60.0 | 5 |
| 3''belt sander | 1000.0 | 1000.0 | 4 |
| Air cooler | 65.0 | 80.0 | 4 |
| Air purifier | 25.0 | 30.0 | 3 |

To achieve this, we need to compute and assign appropriate values to the input\_power, working\_hours, and output\_power columns for each row in the DataFrame. The input\_power can be calculated by averaging the min\_consumption and max\_consumption values, indicating the power input range. The working\_hours column might involve a predefined value or be calculated using domain-specific information, representing the duration of operation. Lastly, the output\_power can be derived from the input\_power and working\_hours, reflecting the energy output based on the input and duration of operation. By performing these calculations and assignments, the DataFrame will be enriched with meaningful values in these three columns, facilitating further analysis and interpretation of the data.

**Data normalization**

**Data normalization is the process of transforming data into a standard format that allows for easier analysis and comparison. Here are some common techniques for data normalization.**

* ****Min-Max Scaling:****

**- This technique scales the data to a range between 0 and 1.**

**- It is calculated by subtracting the minimum value from each data point and then dividing by the range (i.e., the difference between the maximum and minimum values).**

* ****Z-Score Normalization:****

**- This technique scales the data to have a mean of 0 and a standard deviation of 1.**

**- It is calculated by subtracting the mean from each data point and then dividing by the standard deviation.**

* ****Decimal Scaling:****

**- This technique scales the data by dividing each data point by a power of 10.**

**- The power of 10 is chosen based on the maximum absolute value in the dataset.**

* ****Log Transformation:****

**- This technique is used to normalize data that has a skewed distribution.**

**- It involves taking the logarithm of each data point.**

* ****Unit Vector Normalization:****

**- This technique scales the data so that each data point has a length of 1.**

**- It is calculated by dividing each data point by the Euclidean length of the vector.**

**Calculating efficiency metrics**

**This pseudocode is I calculated ehnaced efficency metrics to approch the accurate caltulation.**

**for i = 0 to length of List\_efficiency - 1:**

diff = List\_enhan\_efficiency[i] - List\_efficiency[i]

if diff > 0:

List\_efficiency[i] = List\_efficiency[i] - abs(diff)

else if diff < 0:

List\_efficiency[i] = List\_efficiency[i] + abs(diff)

else:

List\_efficiency[i] = List\_efficiency[i] + abs(diff)

if List\_efficiency[i] < 40:

List\_efficiency[i] = List\_efficiency[i] \* 0.3

else if List\_efficiency[i] > 70:

List\_efficiency[i] = List\_efficiency[i] \* 0.7

List\_efficiency[i] = max(40, List\_efficiency[i])

List\_efficiency[i] = min(70, List\_efficiency[i])

List\_enhan\_efficiency[i] = max(0, min(200, List\_enhan\_efficiency[i]))

data['efficiency'] = List\_efficiency

from sklearn.model\_selection import test train split X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size=0.2, random\_state=0)

* Distance Measurement

The [distance](https://www.analyticsvidhya.com/blog/2020/02/4-types-of-distance-metrics-in-machine-learning/) between clusters can be measured using various techniques including *Euclidean*, *Manhattan*, *Minkowski*, etc. We are using the Euclidean method to calculate distance between two points which can be found by the following formula.

d = sqrt((x1-x2)^2 + (y1-y2)^2)

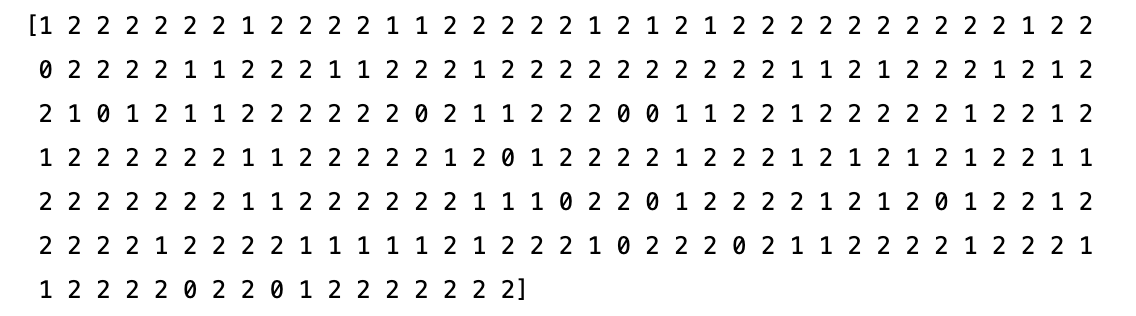
* Agglomerative Clustering

The data can now be analysed and added to the agglomerative clustering method.

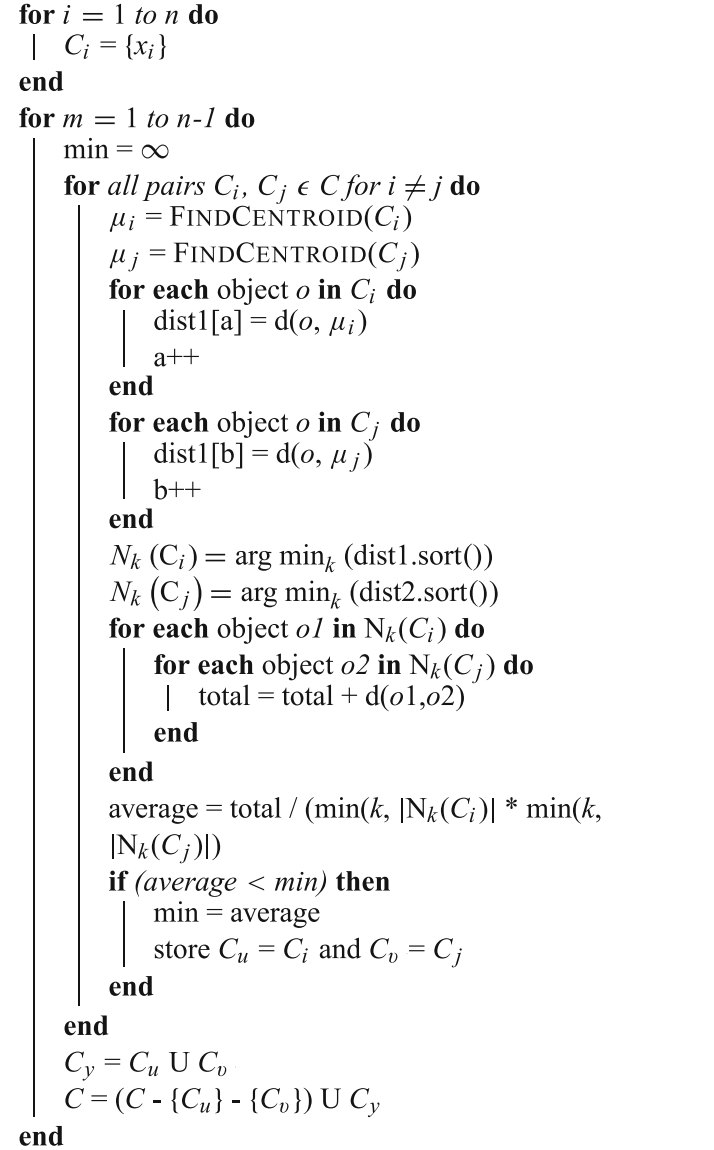
from sklearn.cluster import AgglomerativeClustering

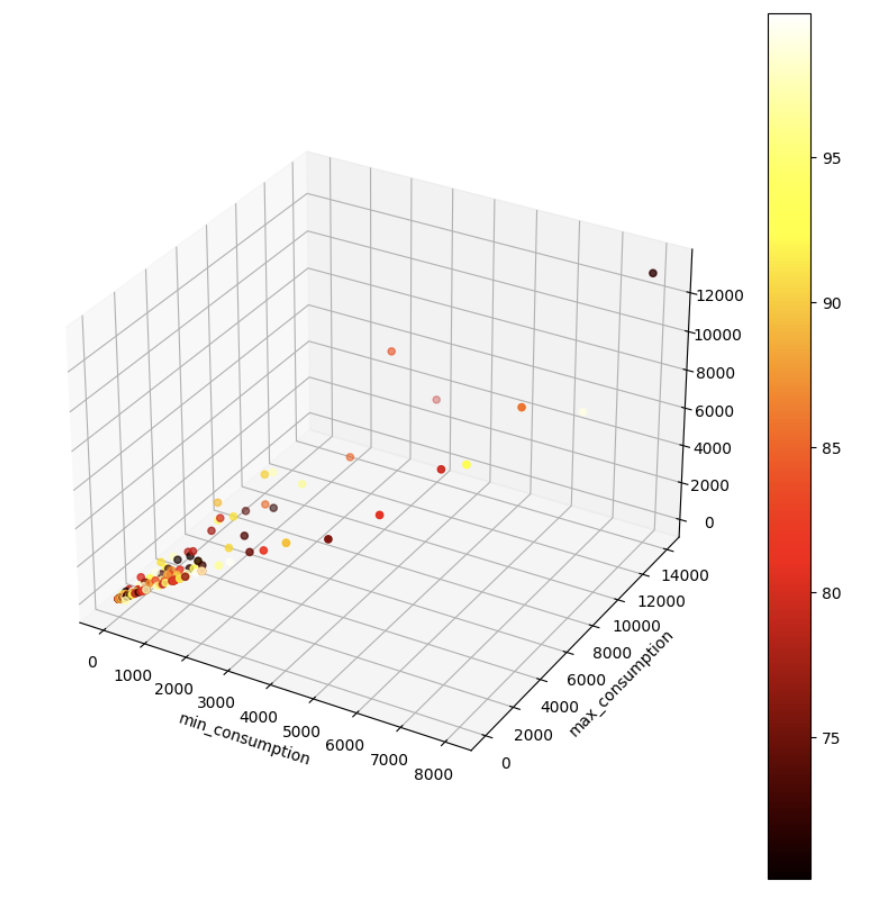
Hc=AgglomerativeClustering(n\_clusters=3, affinity='Euclidean', linkage='ward')

y\_hc= hc.fit\_predict(X\_train)



Now inducing hierarchical clustering for which we will be using the algorithm pseudo code suggested by 'Alican Dogan' and 'Derya Birant'





The k-centroid link algorithm offers a novel perspective on hierarchical clustering by incorporating the average of pairwise distances between k closest objects to the center points of clusters. This innovative approach provides the flexibility to interpolate between conventional centroid link and average link scenarios, enhancing the overall clustering process.

1. REFERENCES
2. A-pplication of k-means and hierarchical clustering techniques for analysis of air pollution: A review (1980-2019).
3. He L, Agard B, Trepanier M (2020) A classification of public transit users with smart card data based on time series distance metrics and a hierarchical clustering method. Transportmetrica A: Transport Science 16(1):56–75
4. Xu, R., & Wunsch, D. (2010). Clustering algorithms in biomedical research: A review. 961 IEEE Reviews in Biomedical Engineering, 3, 120–154.
5. Alican Dogan, Derya Birant (2021) 'k-centroid link: a novel hierarchical clustering linkage method'.

# What is Normalization of Data?

# Everything you need to know about Min-Max normalization: