Turkish Online Journal of Qualitative Inquiry (TOJQI)

Volume 12, Issue 3, July 2021:2122- 2134

Research Article

Clustering and Exploratory Data Analysis for IoT based Heart Rate Monitoring System

M.A. Ajay Mari Kumar[1], M. Rohit[2], S. Nagadevi[3]

Abstract

This paper is aimed at developing an IoT based heart rate monitoring system using a heart rate sensor, NodeMCU which is a free open source platform for IoT and ThingSpeak Cloud. After this the acquired sensor data is been processed with cluster and exploratory data analysis using R Studio. The cluster analysis of heart rate monitoring can also be used for prediction solution of heart rate for a particular patient. The historical heart rate data are collected in Cloud for further analysis and also can be used for prediction. Since the data collected from sensor is high in volume, the data can be categorized as Big Data. Hence the R script is used for processing the data.

Keywords: IoT, *heart rate sensor, thingspeak cloud, cluster, exploratory data analysis, prediction*.

Introduction

The development in health care is enormous that the diagnose and treatment of cardiac disease and malfunctions are easily carried out. The post-surgery or post-operational care is very vital for the effectiveness of the surgery. Continuous monitoring of the patient's heart rate can help to save the patient from any accidental failure. It can also help to make better decisions and conclusions for the medical team as they will be having a larger historical record of the patient.

State of the Art

The present health care service is very widely available to the each and anyone through the extensive[1] remote or IoT based health care systems.The bio sensors are easier and more convenient to use when it wearable[2].

The paper[2] also explains various cloud accessing and execution for monitoring of big data analytics.

[1] UG Scholar, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India, aa1527@srmist.edu.in

[2] UG Scholar, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India, rs5234@srmist.edu.in

[3] Assistant Professor (Sr.G), School of Computing, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India, nagadevs@srmist.edu.in

An Iot cloud site[3] can be used to upload data into it. It provides the developer with various channels from which data can be added. This server also provides the instant visualization of the data which is uploaded from the devices into the cloud server.

The sample data is available from the paper[4]. This data set can be used for data analysis. The data is collected through a FitBit band which is similar to the heart rate sensor which is used implemented the paper.

The cluster and exploratory data analysis of the heart rate data is done in the R Studio which is the novelty from this paper.

Prposed Work

The IoT based heart rate monitoring System architecture is shown in figure 1. ThingSpeak is the IoT agent server used here. It provides a platform to store the IoT based data. It is also used to process the data using various algorithms.

The signal pin of the heart rate monitoring system is connected the A0 pin present in the NodeMCU. Since the analog values are given out from the heart rate sensor it is connected to the analog pin present in the NodeMCU. The other pins present in the heart rate sensor are for power supply to the sensor. The positive terminal of the heart rate monitoring sensor is connected to +3.3v terminal. The neutral or the ground of heart rate sensor is connected to the GND (Ground Terminal) of the NodeMCU.

The NodeMCU acts as a micro controller and also as a wifi modem. The IoT server which is Thinkspeak is connected to the setup through the WiFi module which is present in the NodeMCU. The NodeMCU is powered by the DC supply of 5v. The power source can be a power bank. The power bank is connected to the NodeMCU by means of micro-USB cable. The acquired data is used for the analysis phase. The analysis is done in R studio by various functions and algorithms. After the analysis various insights of the data and also the character of data is derived.

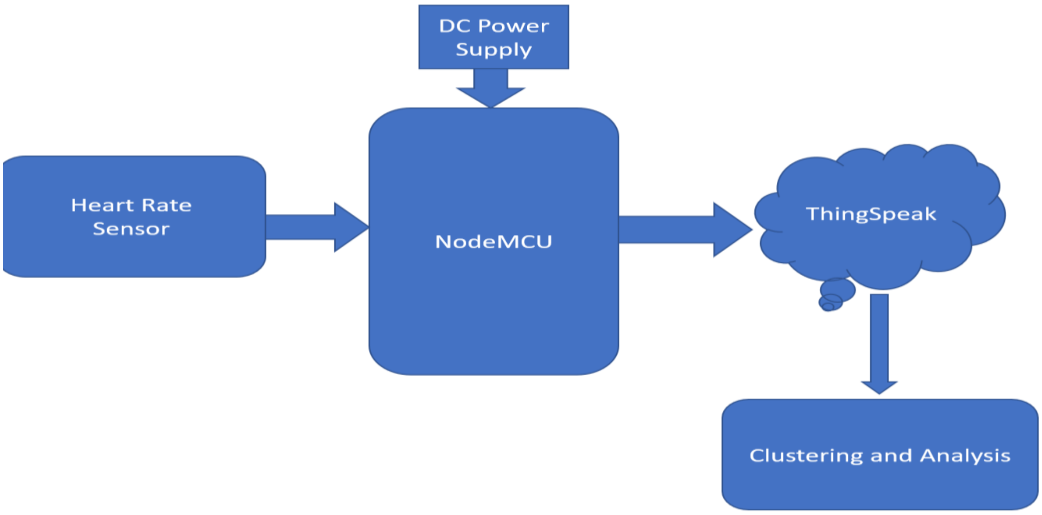


Fig. 1. IoT System Architecture

Implementation

The proposed heart rate monitoring system consists of the NodeMCU and the heart rate sensor as the core hardware parts. Sensors have wide usage in the field of medical Science. Out of which heart rate sensor is used in this paper. It is used to collect data in a continuous phase. The collected data is uploaded in the cloud server through NodeMCU. Further this data is processed in R Script. The entire system is intra-connected by jumper cables. It is powered by a power bank. Figure 2 shows the entire proposed system. The components which are used and their details are given below.

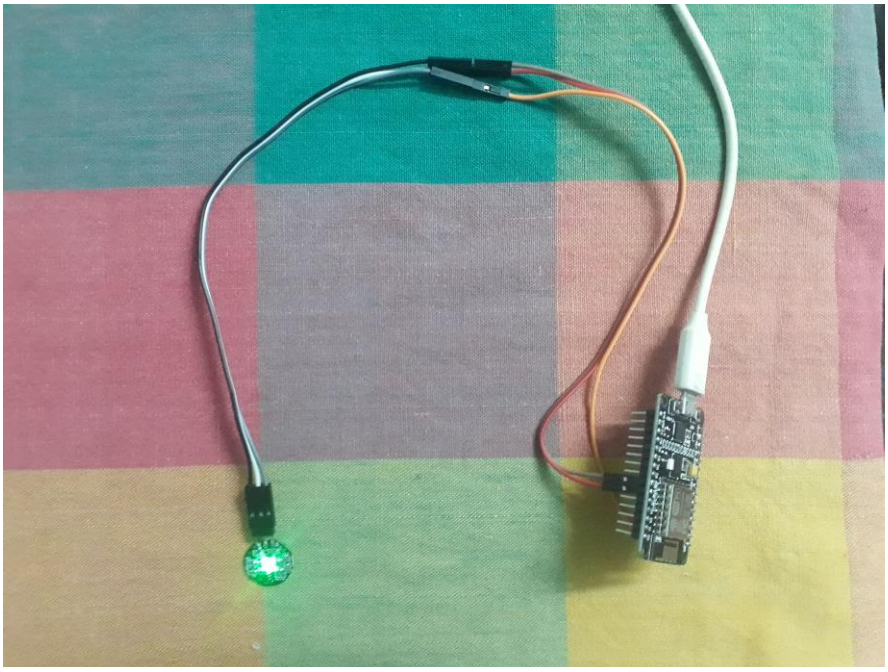


Fig. 2. Heart Rate Monitoring System Setup

A. Node MCU

Fig. 3. Node MCU

NodeMCU is one among the widely used open source IoT platform recently. The NodeMCU consists of an inbuilt firmware which runs on top of the ESP8266 Wi-Fi SoC which is made from Espressif Systems and hardware which is based on the ESP-12 module. This is a microcontroller with some additional features for the common usage. This microcontroller includes a inbuilt support for WiFi connectivity which is used to send data to the ThingSpeak server. Figure 3 Shows the NodeMCU which is used.

B. Heart Rate Sensor

The heart rate sensor[6] used in the setup uses a magnet and a read switch which is involved to detect the liquid presence. As the change of volume of blood is synchronous to the heart rate, this technique or method is used to measure the heart rate. It measures heart rate in bits per minute(Bpm). The figure below 4(a) shows the back side view of heart rate sensor and the figure 4(b) shows the front side view of heart rate sensor.

4(a) 4(b)

Fig. 4(a,b). Heart Rate Sensor

C. Cloud Server Source

ThingSpeak cloud server is a widely used open-source Internet of Things (IoT) application [5] and an API to store and retrieve data from IoT devices using the Hyper Text Transfer Protocol through the Internet. ThingSpeak server is used to create the sensor logging, location tracking applications, and also a social network of IoT devices with status updates. It is widely used by IoT developers. The figure 5 shows the profile which is created in the ThingSpeak server.

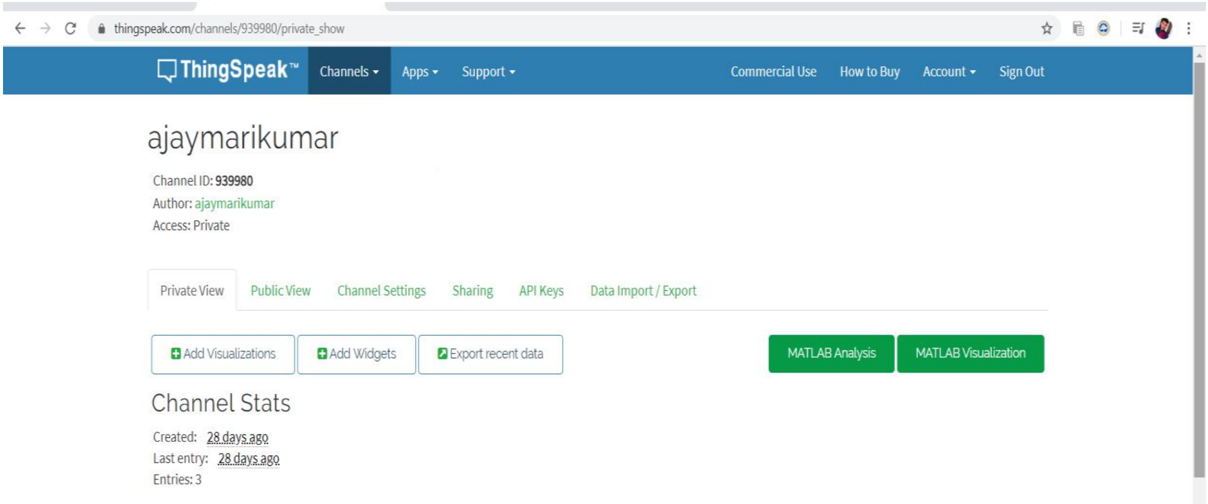


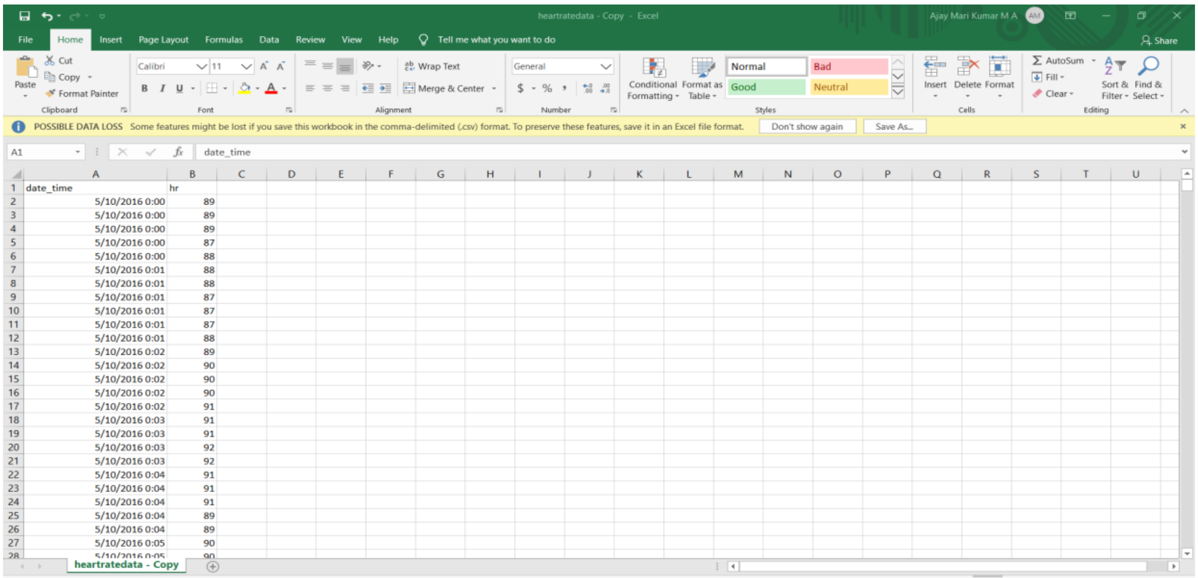
Fig. 5. ThingSpeak Server Profile

D. Setup Process

The pins of the heart rate sensor (s,+,-) are connected to the NodeMCU pins (A0,+3v,GND) by using jumper cables. The heart rate sensor gives out data in the form of analog data thus it is connected to the analog pin in NodeMCU which is A0.The code for the microcontroller is coded in the Arduino software and flashed into the NodeMCU by means of micro-usb cable. The data is displayed in the thinkspeak.com server and can be monitored.

Results Discussion

The patient who is to be monitored is made to press his/her thumb on the heart beat rate sensor so that his/her heart beat rate is recorded in IoT agent and further it is deployed to the ThingSpeak cloud server. This data is further used to process in the RStudio to discover various insights of the data and also the character of data is derived. Figure 6 is the data to be processed.



	A	B
1	date_time	hr
2	5/10/2016 0:00	89
3	5/10/2016 0:00	89
4	5/10/2016 0:00	89
5	5/10/2016 0:00	87
6	5/10/2016 0:00	88
7	5/10/2016 0:01	88
8	5/10/2016 0:01	88
9	5/10/2016 0:01	87
10	5/10/2016 0:01	87
11	5/10/2016 0:01	87
12	5/10/2016 0:01	88
13	5/10/2016 0:02	89
14	5/10/2016 0:02	90
15	5/10/2016 0:02	90
16	5/10/2016 0:02	90
17	5/10/2016 0:02	91
18	5/10/2016 0:03	91
19	5/10/2016 0:03	91
20	5/10/2016 0:03	92
21	5/10/2016 0:03	92
22	5/10/2016 0:04	91
23	5/10/2016 0:04	91
24	5/10/2016 0:04	91
25	5/10/2016 0:04	89
26	5/10/2016 0:04	89
27	5/10/2016 0:05	90
28	5/10/2016 0:05	90

Fig. 6. Raw Data Set

E. Processing of Data

i. Data Preprocessing and Transformation

The data is stored as comma separated value file.(.csv()).This data is at first converted to tab delimited file(.txt) for the operational conveniences. The file had 411,799 entries. The duplicate columns are removed and filtered using the functions[7]: duplicated()- this function determines which elements of a vector contains duplicates of elements which contains smaller subscripts, and returns a logical vector which indicates the elements are duplicates. After removing the duplicates the file had 68,664 entries. The figure 7 shows the data set from which duplicates have been removed.

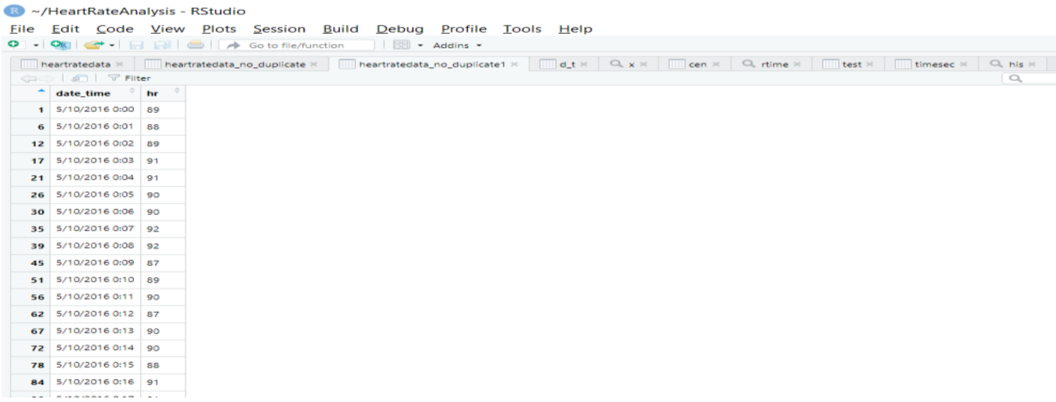


Fig. 7. No Duplicates

The data base had two columns which stored date&time and the heart rate. The date & time column was separated into two columns as date and time. This was done using the function: separate (data, col, into, sep). This function is available in the Library-tidyr. The date in the data base is in the form of character, so it is converted to date format by function: as.Date(x). The figure 8 shows the characters converted to time format.

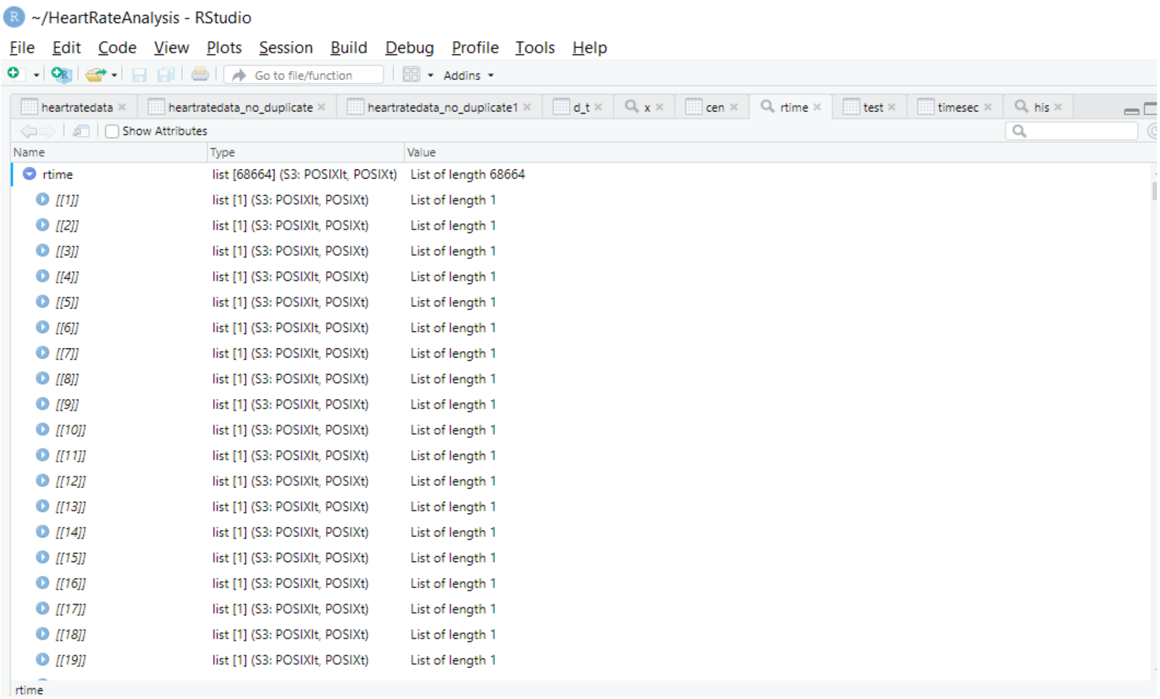


Fig. 8. Rtime

F. Data Mining and Evaluation

i. Cluster and Exploratory Data Analysis

The data set was clustered using the K-Means Algorithm into five clusters. Thus the five most concentrated areas were found out. This is also used to depict the colors of various clusters in the exploratory data analysis. Function used: kmeans (x, centers, iter.max = 10, nstart = 1,algorithm). The figure 9 shows the details of the clustered data.

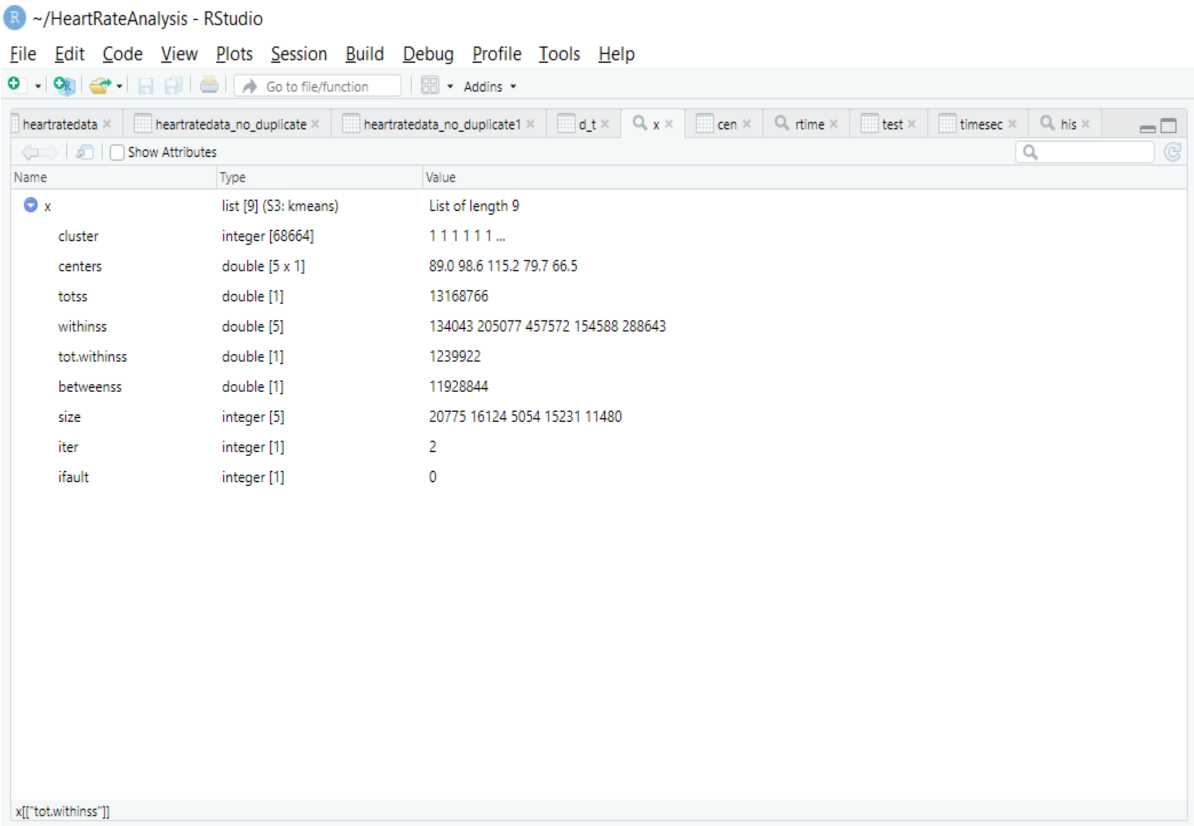
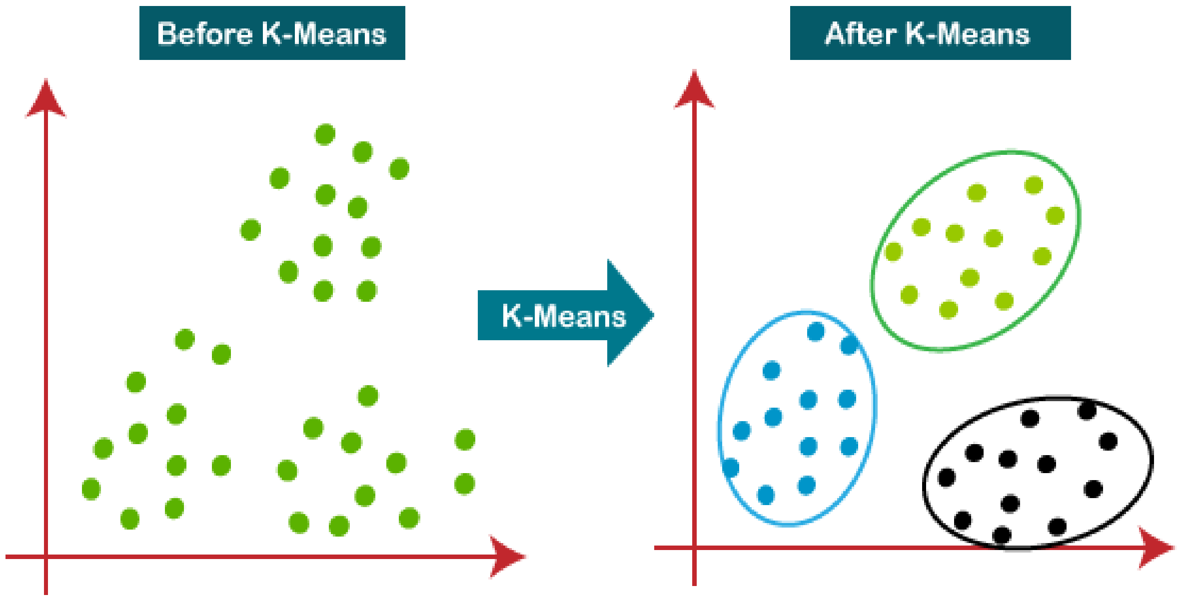


Fig. 9. Cluster Details

K-means Clustering

K-Means Clustering is a clustering methodology which uses and unsupervised learning algorithm. It collects un-labeled data into various collections. The letter K refers to the number of predefined clusters that which should be formed. For example if K is set as 2, then two clusters will be created, and if K is set as 2,then the three clusters will be formed, and so on. It basically processes to collect data on different groups and in an easy way to select groups of categories in a database that is not written alone without the usage of training. It works on a centroid-based algorithm, in which each collection is associated with the selected centroid. The main aim of the algorithm is to find the distance between the data point and every clusters. The algorithm inputs unwritten data, splits the data into k number of clusters. Then the process repeated until the best cluster is found. The numeric value of k should be predefined in this algorithm.



K-means Algorithm

Steps to be followed:

1: Decide and fix the number K as the number of clusters.

2: Select any K point or centroid. (It could be something from the given database).

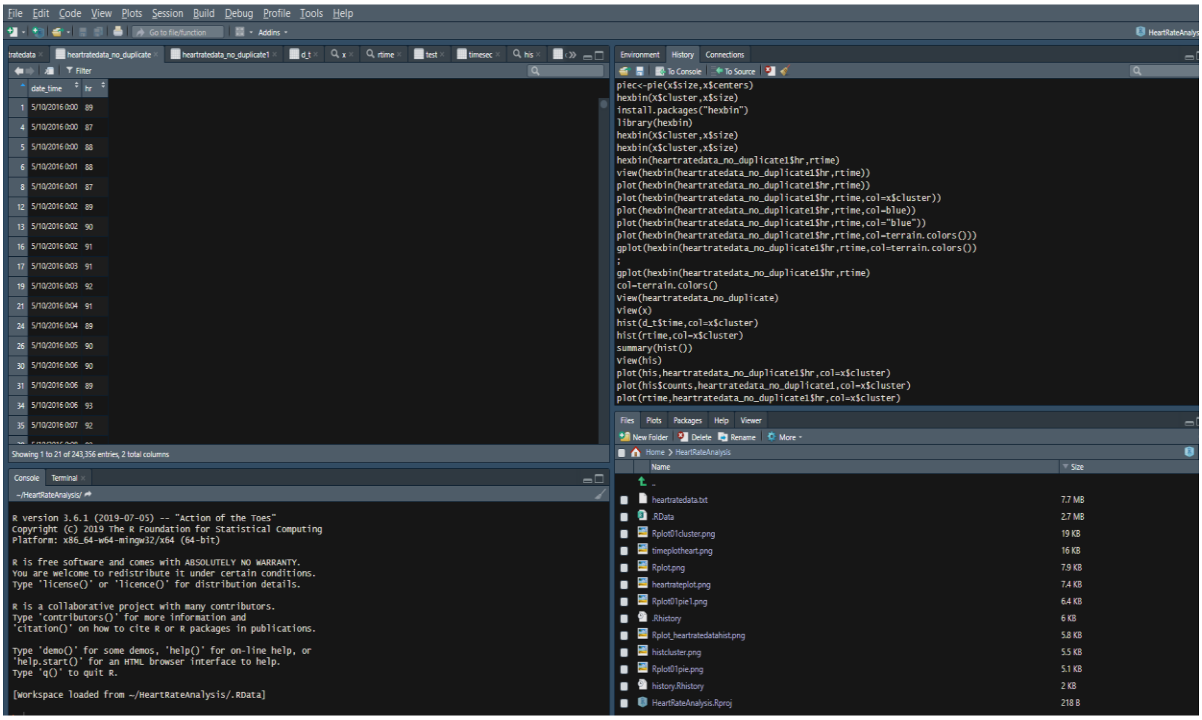
3: Assign one by one point of data to its nearest centroid, which forma the K pre-defined clusters.

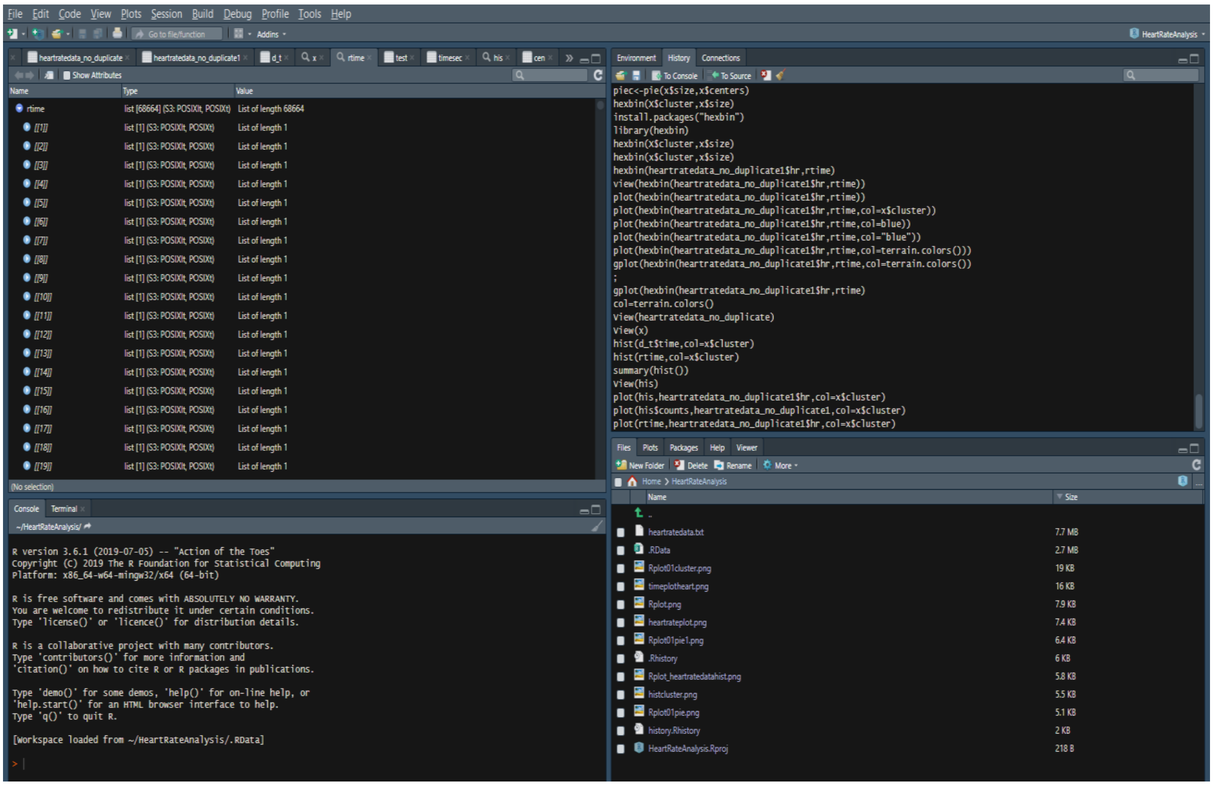
4: Calculate and determine the variation and select a new centroid for each collection.

5: Repeat the step-3, which redistributes each datapoint to its nearest new centroid for each collection.

6: If redistribution has occured, go again to step-4 further to end.

7: The model is clustered.





a) Histogram

Histogram was created to compare and visualize the relations and also the spread of data.

Function used: histogram (x, data, ...)

1-The occurrence of the heart rate was visualized with the frequency. It showed that the heart rate in the range of 80bpm-90bpm was most frequent. The figure 10 depicts the histogram of the heart rate versus frequency.

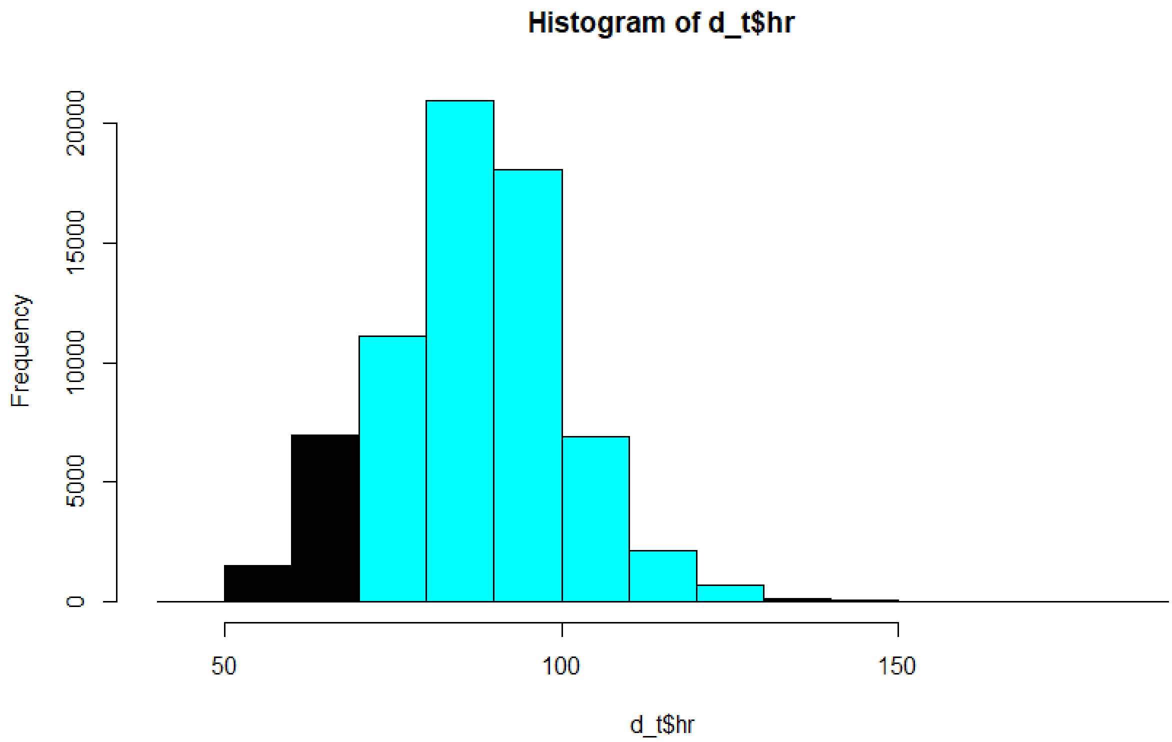


Fig. 10. Histogram

b) Scatter Plot

Scatter Plot was used to visualize the spread of individual points.

Function used:plot(x,…)

By using cluster analysis, the various clusters was expressed in various colors and visualized with the time and heartrate. The colors depicted the three safe zones and two abnormal zones, which was the five clusters. The figure 11 shows the clustered scatterplot. The heart rate was visualized in relation with the date. It showed that it is quite low in some days. The figure 12 shows the date versus heart rate scatterplot.

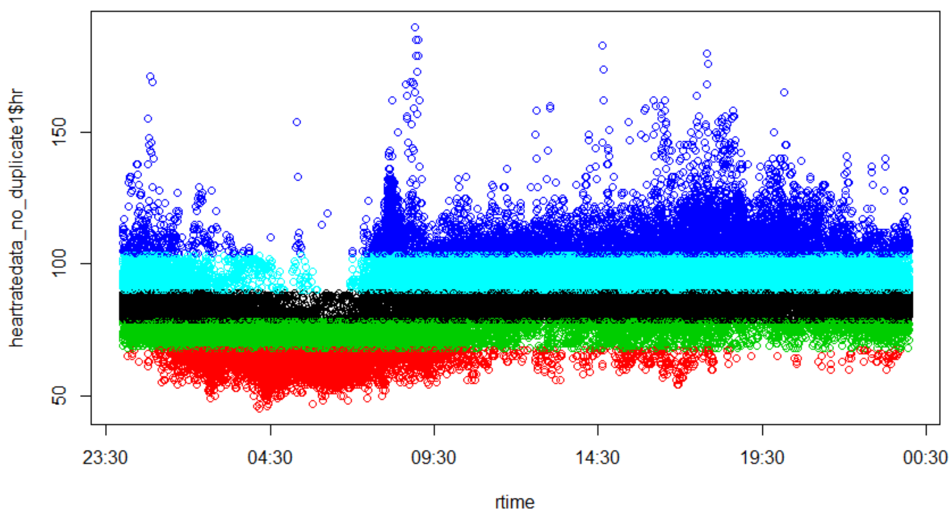


Fig. 11. Clustered Scatterplot

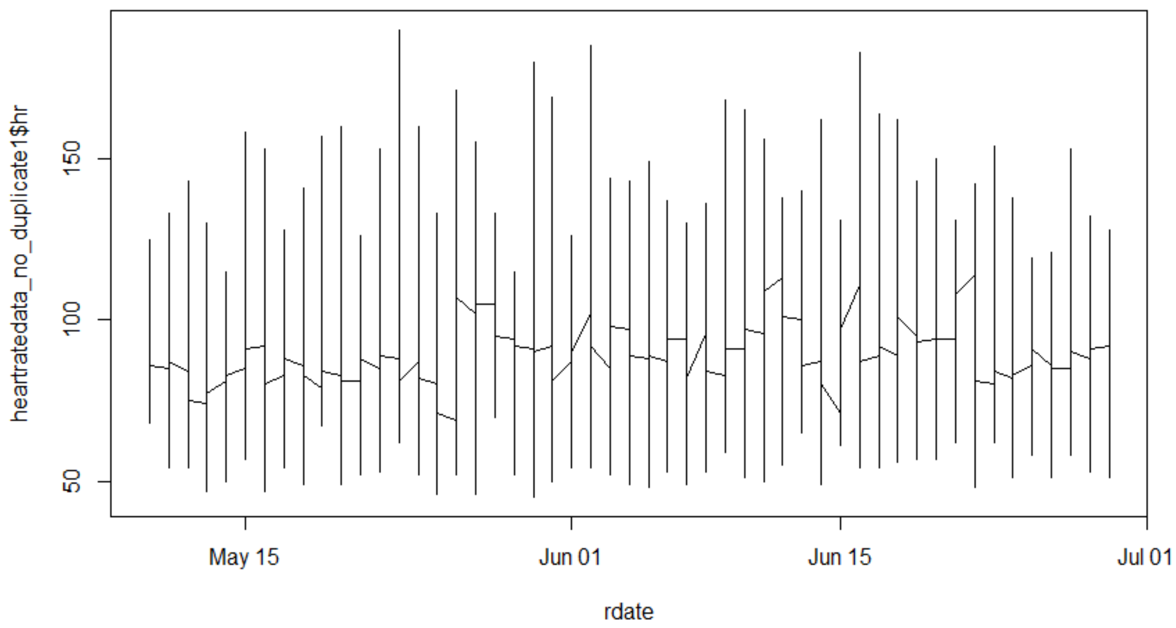


Fig. 12. Date Scatterplot

c) Pie Charts

Function: pie(x, labels = names(x), edges = 200, radius = 0.8, clockwise = FALSE, init.angle = if(clockwise) 90 else 0,density = NULL, angle = 45, col = NULL, border = NULL, lty = NULL, main = NULL) Pie chart was used to visualize the relation between the cluster centers and frequencies. The figure 13 and figure 14 shows relation between the cluster centers and the count.

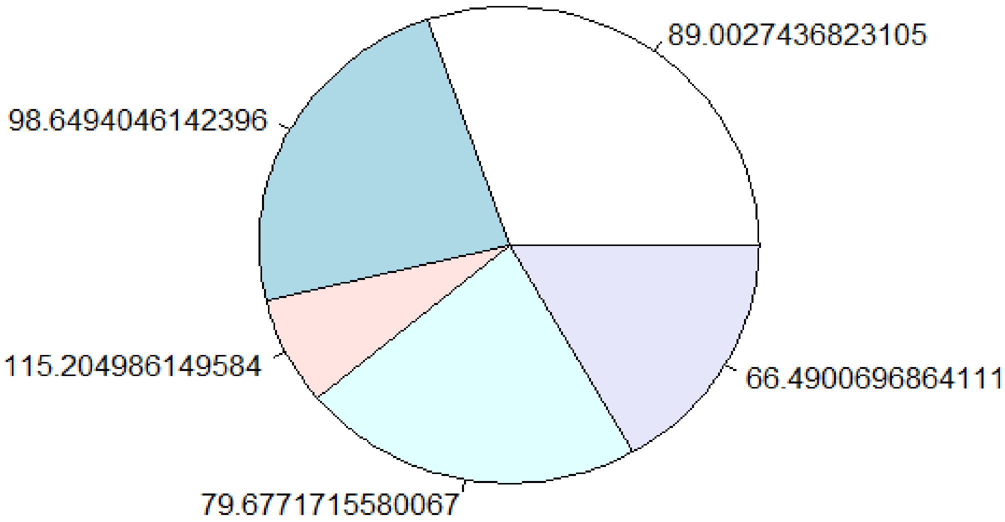


Fig. 13. Cluters Center Pie

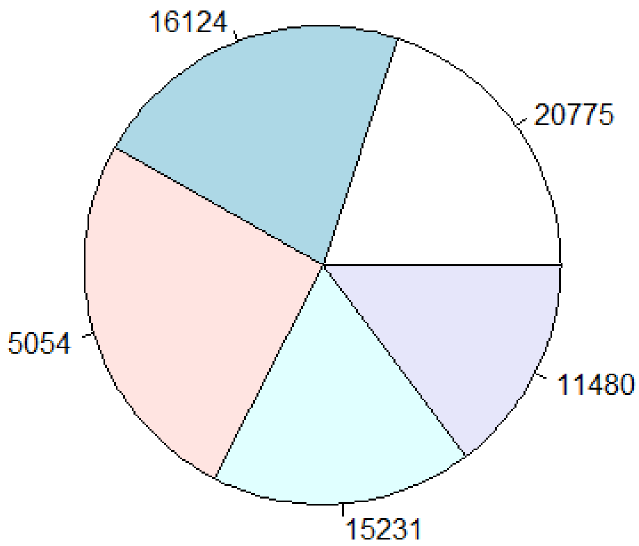


Fig. 14. Cluster Count Pie

Conclusion

An IoT based setup to monitor and analyze the heart rate of a patient is implemented. The ThingSpeak cloud is used as a platform to deploy the data. The deployed data is further processed using R Script. The cluster and exploratory data analysis are being performed in the R studio. The proposed system can highly help the medical team or the physicians to understand the situation or the condition of the patient better. It can also help to make better decisions as they are being provided with a large amount of data. The data can also be used to make assumptions using predictive methods.

References

1. Triantafyllidis, A.K., Velardo, C., Salvi, D., Shah, S.A., Koutkias, V.G., & Tarassenko, L. (2015). A survey of mobile phone sensing, self-reporting, and social sharing for pervasive healthcare. *IEEE journal of biomedical and health informatics, 21*(1), 218-227. https://doi.org/10.1109/JBHI.2015.2483902

2. Korzun, D.G., Nikolaevskiy, I., & Gurtov, A. (2015). Service intelligence support for medical sensor networks in personalized mobile health systems. *In Internet of things, smart spaces, and next generation networks and systems*, 116-127.

3. https://thingspeak.com/pages/learn_more?Education

4. Salamon, J., & Mouček, R. (2017). Heart rate and sentiment experimental data with common timeline. *Data in brief, 15*, 851-861. https://doi.org/10.1016/j.dib.2017.10.037

5. Rajanna, R.R., Natarajan, S., & Vittal, P.R. (2018). An IoT Wi-Fi Connected Sensor For Real Time Heart Rate Variability Monitoring. *In 2018 3rd International Conference on Circuits, Control, Communication and Computing (I4C)*, 1-4. https://doi.org/10.1109/CIMCA.2018.8739323

6. Senthilkumar, R., Ponmagal, R.S., & Sujatha, K. (2016). Efficient health care monitoring and emergency management system using IoT. *International Journal of Control Theory and Applications, 9*(4), 137-145.

7. https://ieeexplore.ieee.org/document/8079620

8. https://youtu.be/E7kbJFxocvU

9. Maheswari, K.U., & Govindarajan, S. (2019). Hybridisation of oppositional centre-based genetic algorithms for resource allocation in cloud. *International Journal of Networking and Virtual Organisations*, *21*(3), 307-325.

10. Umamaheswari, K.M., Chanana, T., & Fernandes, K. (2020). Meme Chat: An Innovative Social Media Platform for Content Monetization & Corporate Outreach Using OCR & NLP. *Journal of Critical Reviews, 7*(4), 206-208.

11. Maheswari, K.U., Roy, S., & Govindarajan, S. (2006). Hybrid Green Scheduling Algorithm Using Genetic Algorithm and Particle Swarm Optimization Algorithm in IAAS Cloud. *ARPN Journal of Engineering and Applied Sciences*, *12*(12), 3762-3766.