Chapter 1

Introduction

1.1 Introduction to Pulse oximeter

Pulse oximeter is the device that work on the principle of Pulse oximetry. Pulse oximetry is the method for monitoring the people's oxygen saturation (SpO2) without harming their body. The Standard SpO2 level is between 95 to 100 percentages. SpO2 level is mainly depends on R (ratio) value. R value is the ratio value of Red light to Infrared light. The wave length of Red is at 660nm and IR (infrared) is at 940nm.

Pulse oximeter checks to see if people's blood oxygen level is within a healthy range or not. When your oxygen level is down from normal range, it is called hypoxemia.

When person's oxygen level falls down from normal range, the person's lungs starting to become narrow, it is res started to slower down the blood flow by the lungs. When this things happen more stress has been placed on the heart. This could be lead to Serious Heart failure problem for people.

Some symptoms of hypoxemia is given below:

- Feeling stress on Heart
- Problem in normal breathing
- Feeling stress
- Feeling restlessness

Pulse oximetry was developed in 1972, by Takuo Aoyagi and Michio Kishi, bioengineers, at Nihon Kohden using the ratio of red to infrared light absorption of pulsating components at the measuring site. [1]

Susumu Nakajima, a surgeon, and his associates first tested the device in patients, reporting it in 1975. It was commercialized by Biox in 1980. [1]

Pulse oximeter mainly uses in Hospital in below conditions:

- chronic obstructive pulmonary disease (COPD),[2]
- asthma[2]
- pneumonia[2]
- lung cancer[2]
- anemia[2]
- heart attack or heart failure[2]
- congenital heart defects[2]

1.2 Introduction to Internet of Things (IoT):

IoT (internet of Things) is a technology that can connect all the devices to the cloud and devices are controlled and monitor through cloud. After IoT is coming in to the Picture the sensor are attached with device that can give the values that can be measured from any were in the world through cloud by accessing remotely. The IoT has been used in many sectors like smart home, smart city, smart hospitals, smart Television and Smart Phones etc. IoT has reduced the cost of monitoring the patients through cheap sensors that can be helpful for doctor as well as patients. The use of IoT devices has been increased rapidly day by day. IoT can change the way of our Living.

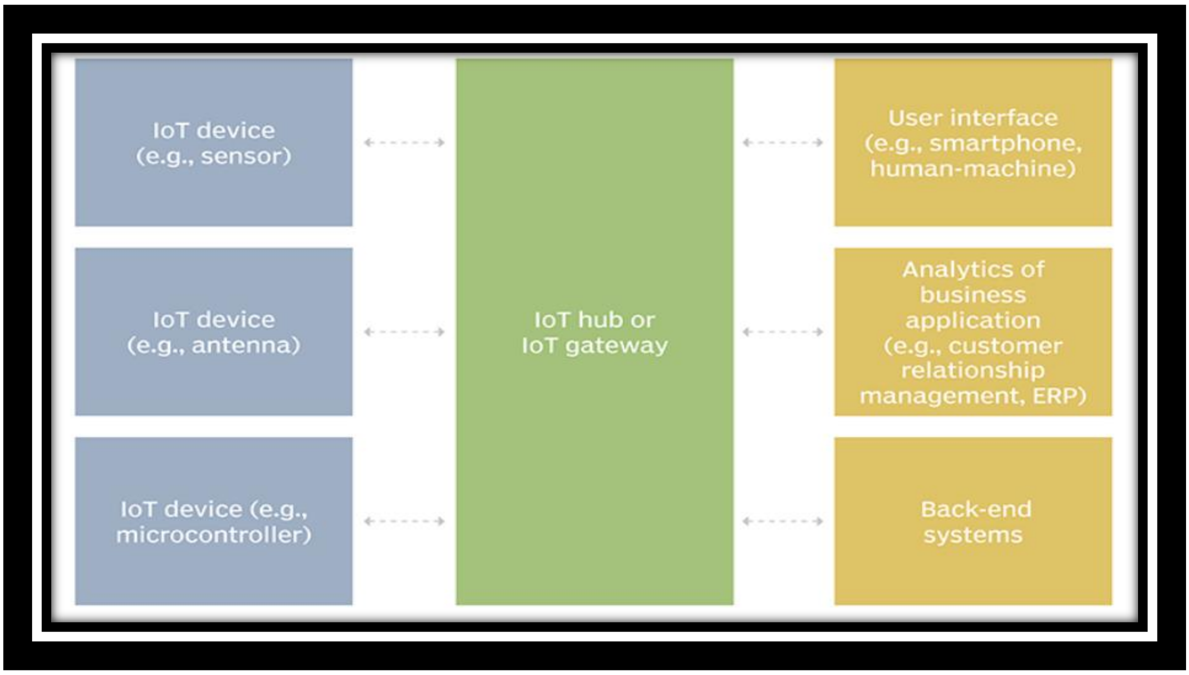


Fig1.1: IoT Architecture

1.3 Introduction to Machine Learning (ML):

Machine leaning is the way to teach the Algorithms to learn from its past experience and updated itself. Machine learning is a part of AI (artificial intelligence).Machine learning algorithms are build model in Joblib file on sample data. On the sample Data the partition should be possible for train and test data.so that it is used later to measure the accuracy if the algorithms. There are many software used for performing machine learning algorithms like spyder, pycharm, jupyter notebook etc.

There are three types of Machine Learning given below:

- Supervised machine learning
- Unsupervised machine learning
- Reinforcement machine learning

In supervised Machine learning Labels are attached with data. In unsupervised Machine learning labels are not attached with data.in unsupervised machine learning Clustering based approach is taken. In Reinforcement Machine learning feedback Based approach is taken.

Based on the Data the machine learning algorithms are two types given below:

- Classification
- Regression

The programmer is decide to base on data they want to choose classification or regression. Normally when we want to classify some values at that time Classification algorithms are used and when we want to predict continues values like temperature at that time regression algorithms are used.

Chapter 2

Literature Review

2.1:

Title: web based patient health monitoring system using raspberry pi [3]
Author: Dhiraj sunehra and Pini Ramakrishna [3]
Publication: 2nd international conference on contemporary computing and Informatics (IEEE) [3]
Year: 2016

Objective:
To monitor patient's condition remotely when patient is not at Hospital and save the patient's life Before going to critical condition.

Problem:
Make the device that monitor the patient from any were anytime and Give the report to doctor as Well as patient.

Methodology:
Various sensors such as Blood pressure, Heart beat, Oxygen Saturation in the blood (Spo2), fall Detection and Body Temperature Are interfaced with Arduino Micro controller board for further Signal processing. All computed Parameters are send to Raspberry pi Web server for display on the web page. ZigBee module is used for Communication between raspberry pi and Arduino Uno.

Implementation:
The flow of implementation at software and Arduino side shows in Fig 2.1 the flow of Implementation at software raspberry pi and Web server side shows in Fig 2.2.

Results: The result of a patient is updated at every 60 seconds. The results shows in Fig 2.3

Conclusion: This System Is Useful for monitoring the health Status of elderly and ill patients. Who Are Not Able To Visit Hospital Daily and Require Assistance in Critical Condition. [3]

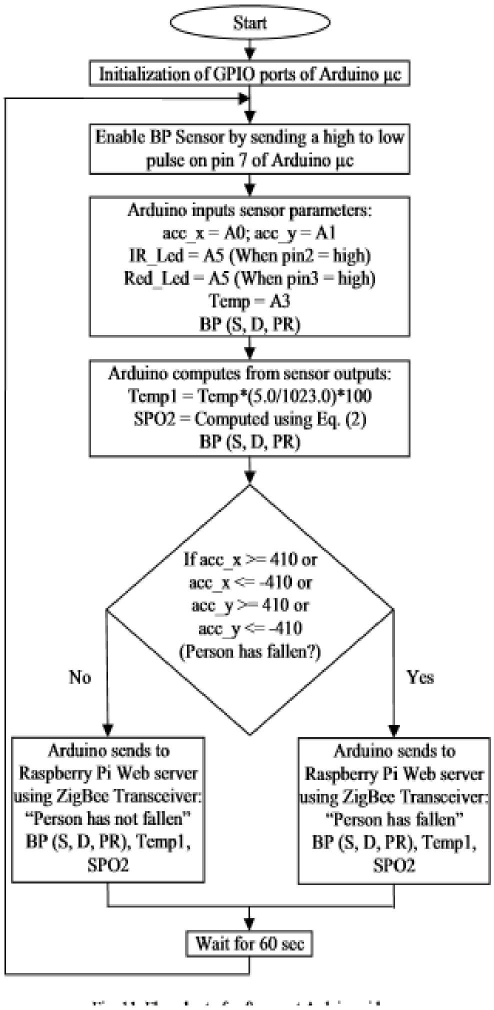


Fig 2.1: Flow chart on software and Arduino side [3]

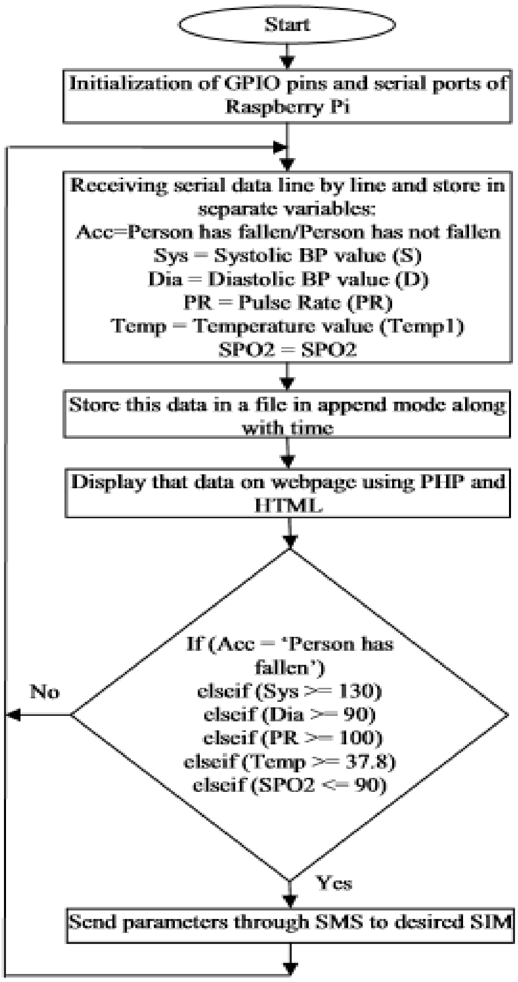


Fig 2.2: Flow chart on Raspberry pi and server side [3]

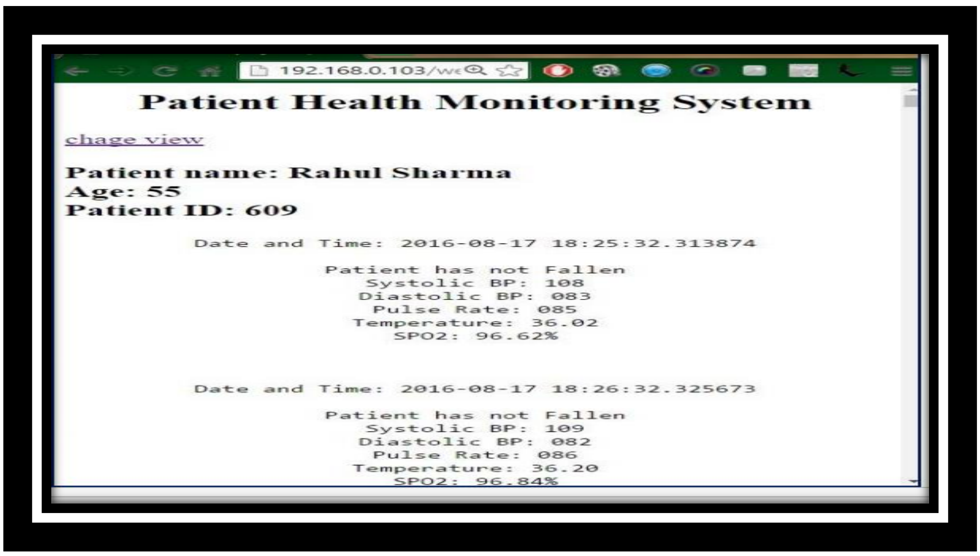


Fig 2.3: Web Result [3]

2.2:

Title: Micro controller Based Pulse Oximeter for Undergraduate Capstone Design [4]
Author: Michael Tamayo, Andrew Westover and Ying Sun [4]
Publication: Proceedings of the 2010 IEEE 36th Annual Northeast Bioengineering Conference [4]
Year: 2010

Objective:
To make the Pulse oximeter design using PIC18f452 micro controller.

Problem:
The purpose of this design is to examine the possibility of using a simple Micro controller to measure pulse and blood-oxygen saturation in real time, As well as the level of difficulty and learning curve associated with assigning Such a task as a senior capstone design project. The circuit consists of a PIC18F452 micro controller, transistor network, photoplethysmogram amplifier (PPG), pulse oximeter probe, digital-to-analog converter (DAC), and an LCD Screen to display results. Software is written in C++ for the microprocessor, which supports a limited C syntax. [4]

Methodology:
Code is written in C++ using the CC8E compiler with MPLA (Math Works, Natick, MA) in order to integrate the PIC processor With IDE. [4]

Results:
The LCD screen updates when four peaks have been detected after switching LEDs, Accurately displaying pulse and oxygen saturation of the blood. Figure 2 depicts a user's arterial pressure waveform with accurate peak-valley detection and LED switching across a 20 second span. These results were obtained using an oscilloscope, with channel 1 connect To the DAC to show the arterial pressure waveform and channel2 to an output port of the PIC set to an alternate between 0 and 5 volts corresponding to peak-valley detection. [4]

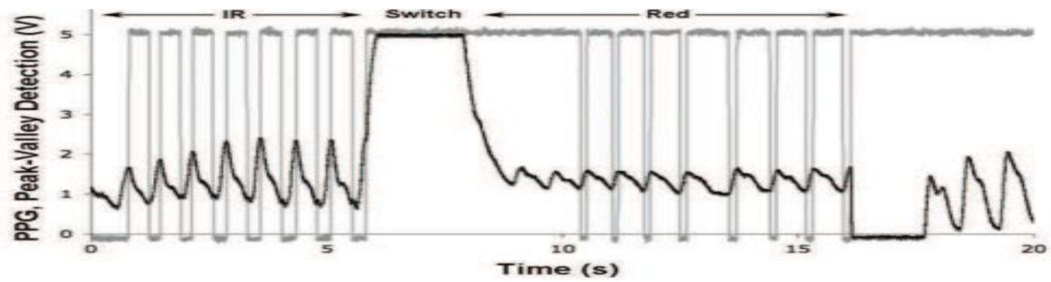


Fig: 2.4 Result of Arterial pressure results from peak and valley [4]

Conclusion:
Accurate heart rate and oxygen saturation results were obtained using a PIC Processor incapable of complex operations.

2.3:

Title: Mini Home-Based Vital Sign Monitor with Android Mobile Application (my Vital Gear) [5]
Author: Yuan wen hau and Mas Azalya Yusof [5]
Publication: IEEE-EMBS Conference on Biomedical Engineering and Sciences [5]
Year: 2018

Objective:
To monitor the health using Arduino microcontroller and Make the Android mobile application.

Problem:
High accurate mini home based vital sign monitor which can measure human vital signs of SpO2, ECG and body temperature based on Arduino Nano technology with Android-based mobile application support and auto alarm system. [5]

Methodology:
The software used to programming is Arduino IDE (integrated development environment) and code is written in C language.

Results:
The result is shows in Fig 2.5 and Fig 2.6

Conclusion:
The Mini Vital Sign monitor is integrated with an Android-based mobile application with simple user interface to display real-time vital sign monitoring result and auto-send notification in case of any abnormality is detected.[5]

Fig 2.5: (a) Home Screen of the mobile application [5]

(b)Second page interface of the mobile application [5]

(c)Temperature measurement is displayed on the screen [5]

Vital Signs	Accuracy (%)
Heart Rate	99.4
Temperature	99.7
Oxygen Saturation(Spo2)	98.1

Table: 2.1 Vital Sign Measurement Accuracy [5]

2.4:

Title: An ANN based SpO2 Measurement for Clinical Management Systems. [6]
Author: Garima P Gupta, Rajashree R Nair, R Jeyanthi. [6]
Publication: Energy Procedia (Elsevier) [6]
Year: 2017

Objective:
To measure the Spo2 using LabVIEW software and compare three methods are fuzzy Logic, Linear Regression and ANN.

Problem:
Pulse oximeters are more accurate method then other methods for measuring the Spo2 Level. The samples of 20 patients are taken for normal to breathing problem patients.

Methodology:

SpO2	Saturation oxygen level in blood in percentage
A	Total amount of blood
B	Amount of oxygenated blood
R	Ratio value

Table: 2.2 Naming [6]

EQUATION: SpO2=B/A*100

Results:

subjects	R value	ANN	Fuzzy logic	Linear regression
1	0.100	99.26	97.10	93.59
2	0.250	96.83	96.72	90.19
3	0.140	98.65	96.81	92.67
4	0.220	97.37	96.69	90.87
5	0.270	96.50	96.22	89.74
6	0.160	98.34	96.74	92.26
7	0.110	99.11	97.02	93.35
8	0.130	97.19	96.71	90.64
9	0.249	96.87	96.75	90.21
10	0.290	96.15	95.74	89.29
11	0.154	98.43	96.75	92.36
12	0.102	99.23	97.08	89.53
13	0.275	96.41	96.10	89.62
14	0.172	98.18	96.71	92.00
15	0.262	96.64	96.43	89.92
16	0.245	96.94	96.75	92.30
17	0.160	98.34	96.74	92.22
18	0.132	98.78	96.86	92.85
19	0.210	97.53	96.66	91.09
20	0.249	96.87	96.75	90.21

Table: 2.3 Comparison Table [6]

Conclusion:
Results of 20 subjects are shown in Table.3. In this, linear regression method shows poor capturing ability. ANN and Fuzzy based methods are show quite consistent values, but ANN captures accurate value than Fuzzy logic based. [6]

2.5:

Title: Machine Learning based SpO2 Computation Using Reflectance Pulse Oximetry [7]
Author: Swaathi Venkat,Mohamed Tanveejul Arsath PS,Preejith SP [7]
And Annamol Alex [7]
Publication: 41st Annual International Conference of the IEEE Engineering in Medicine and Medicine and biological science [7]
Year: 2019

Objective: To use ML algorithm (Linear regression) for different age of persons.

Problem: To make Spo2 measurement training module using LabVIEW software And trying to get accurate result of different age persons using ML Algorithm (Linear regression).

Methodology:

Age group	No. of male subjects	No. of female subjects	Pulmonary Disorders (No.)		
			COPD	Respiratory failure	Others
19-20	0	1	0	0	1
21-30	2	2	0	0	4
31-40	3	3	0	1	5
41-50	8	10	0	2	16
51-60	11	14	4	0	21
61-70	15	16	3	1	27
71-80	8	2	0	0	10

Table: 2.4 subject Meta data [7]

Results:

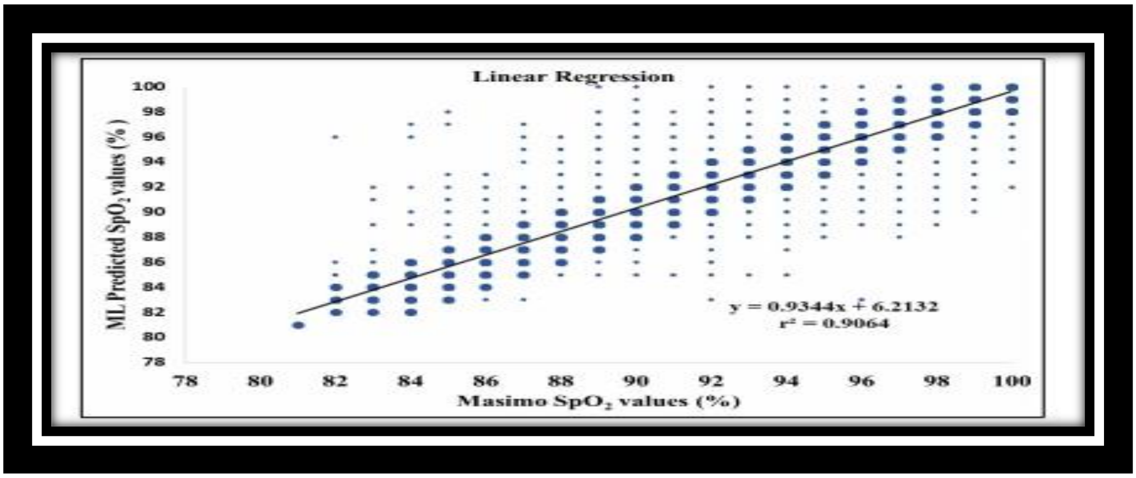


Fig 2.6: Linear regression plot [7]

Class	Total no. of test observations	No. of observations predicted with ±2% error	Accuracy (in %)	Absolute mean error (in %)
81	1	1	100	0
82	15	9	60	2.86
83	52	46	88.46	1.59
84	112	105	93.75	1.05
85	223	205	91.92	0.90
86	551	534	96.91	0.37
87	311	283	90.99	0.78
88	498	446	89.55	0.90
89	490	427	87.14	1.25
90	1095	1042	95.15	0.75
91	1552	1499	96.58	0.61
92	2149	2073	96.46	0.51
93	1897	1862	98.15	0.48
94	2427	2364	97.40	0.35
95	1823	1767	96.92	0.47
96	1900	1848	97.26	0.54
97	3243	3205	98.82	0.27
98	2050	2020	98.53	0.42
99	891	845	94.83	0.62
100	785	747	95.15	0.30

Table 2.5: class wise observations [7]

Conclusion:
The model was less accurate for 81-90 class data set because of given insufficient Data set. Data set for which Spo2 is less than 90% is crucial Have to take timely because of hypoxia. Also the given model can be extended on wrist and chest for Spo2 Reflectance PPG measurements

Chapter 3

Implementation of Pulse oximeter

3.1 Components needed:

- JHD 162A 16*2 LCD display
- ESP8266 (NODEMCU) Wi-Fi module
- Bread board
- Male to Male, Female to Male
- And Female To Female connecter
- MAX 30100 moduleI2C Module

3.1.1: JHD 162A 16*2 LCD Display:

- LCD display module with Yellow Backlight[8]
- SIZE : 20x4 (2 Rows and 16 Characters Per Row)[8]
- Built-in industry standard HD44780 equivalent LCD controller[8]
- Commonly Used in: Student Project, Collage, copiers, fax machines, laser printers, industrial test equipment, networking equipment such as routers and storage devices[8]
- Operate with 5V DC LCD type: Characters[8]

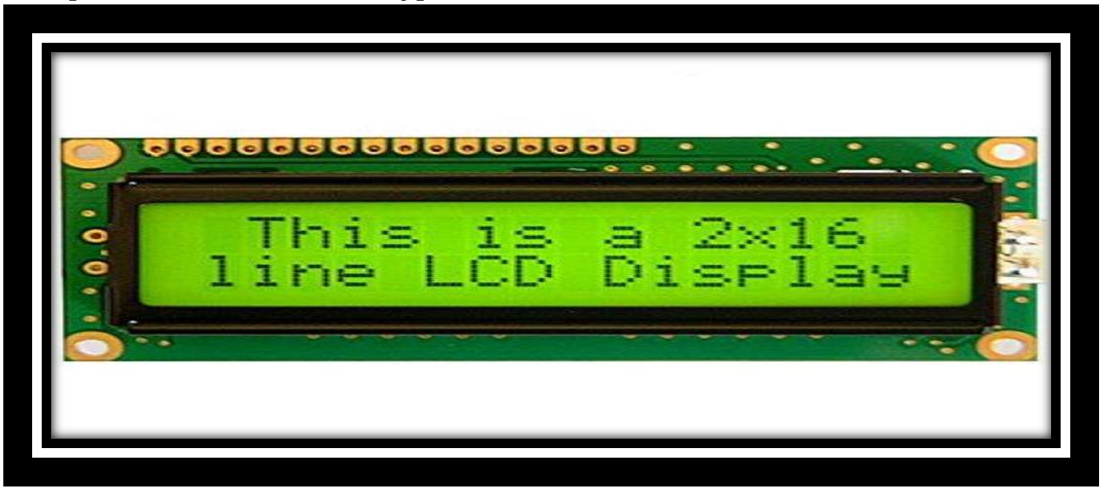


Fig: 3.1 JHD 162A LCD Display [8]

3.1.2:ESP8266 Wi-Fi module:

Node MCU is an IoT Module based on ESP8266 Wi-Fi Module. Node MCU uses Lua Scripting language and is an open source Internet of Things (IoT) platform. This modules has CH340g USB to TTL IC. [9]

Specification of Node-MCU IoT Module:

- It is based on ESP8266, integrates GPIO, PWM, IIC, 1-Wire and ADC all in one board.[9]
- Power your development in the fastest way combining with Node MCU Firmware.[9]
- USB-TTL included, plug&play. [9]
- 10 GPIO, every GPIO can be PWM, I2C, 1-wire[9]

Features of Node-MCU IoT Module:-`

- Open source IoT Platform. [9]
- Easily Programmable. [9]
- Low cost & Simple to Implement.[9]
- WI-FI enabled[9]

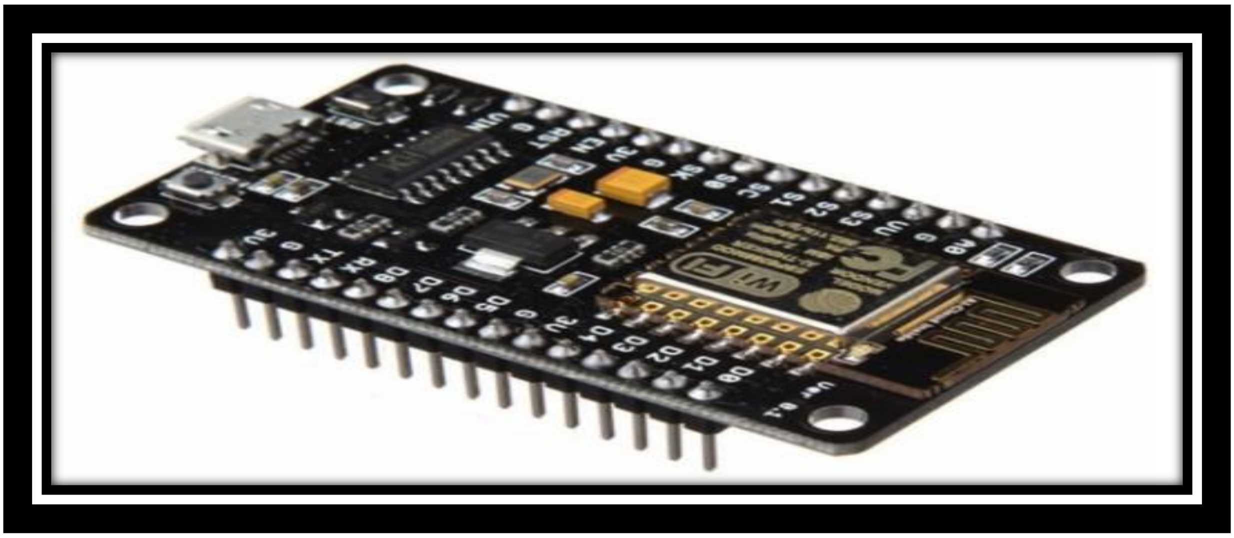


Fig: 3.2 Node MCU Wi-Fi Module [9]

3.1.3:MAX30100 Sensor:

The MAX30100 is an integrated pulse oximetry and heart-rate monitor sensor solution. It combines two LEDs, a photodetector, optimized optics, and low-noise analog signal processing to detect pulse oximetry and heart-rate signals.[10]
The MAX30100 operates from 1.8V and 3.3V power supplies and can be powered down through software with negligible standby current, permitting the power supply to remain connected at all times.

Applications/Uses

- Fitness Assistant Devices[10]
- Medical Monitoring Devices[10]
- Wearable Devices[10]

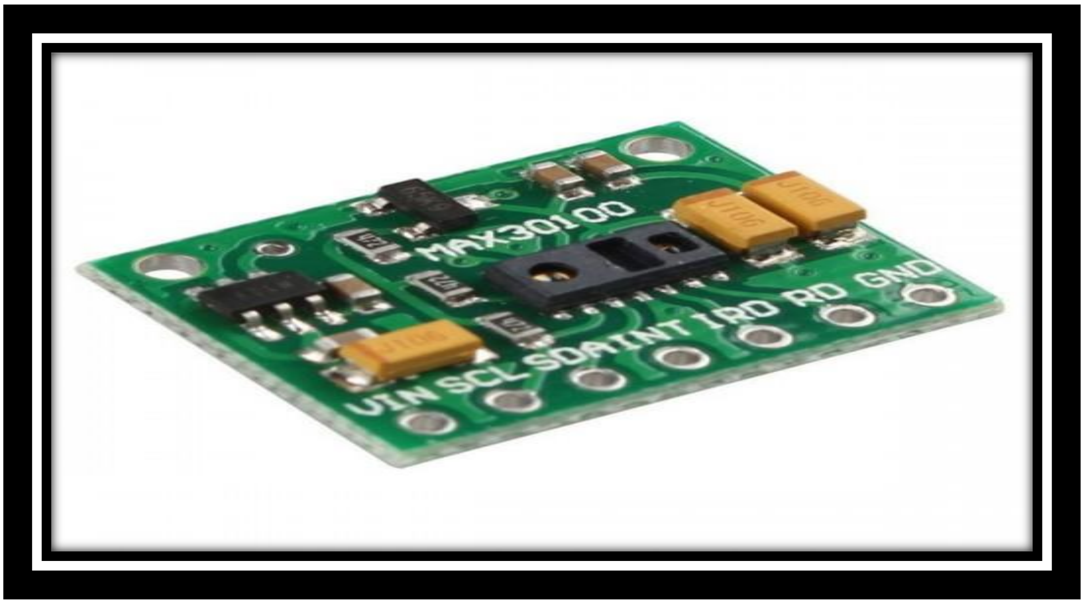


Fig3.3: MAX30100 Module [10]

3.1.4:I2C module:

IIC/I2C Interface Adapter Module is used for 16×2 LCD Display. It uses the PCF8574T IC chip which converts I2C serial data to parallel data for the LCD display. Also this interface module simplifies connecting an Arduino to a 16×2 Liquid Crystal display using only 4 wires. [11]

Features:

- Serial I2C control of LCD display using PCF8574 [11]
- 5V power supply [11]
- Backlight can be enabled or disabled via a jumper on the board [11]
- Contrast control via a potentiometer [11]

Fig3.4:I2C Module [11]

3.2 Interfacing LCD, Node MCU and MAX30100:

The software for writing the code is Arduino IDE (Integrated Development Environment).The code is similar to C language code. The sketch written in Arduino IDE when it complied successfully after that code is uploaded to Node MCU. The compiling code is given in Fig.3.5.The results of Spo2 and Heart rate shown in display on Fig.3.6.

For Compilation of code the following libraries are added compulsory:

- MAX30100_PulseOximeter.
- Wire.h
- ESP8266WiFi.h
- LiquidCrystal_I2C.h

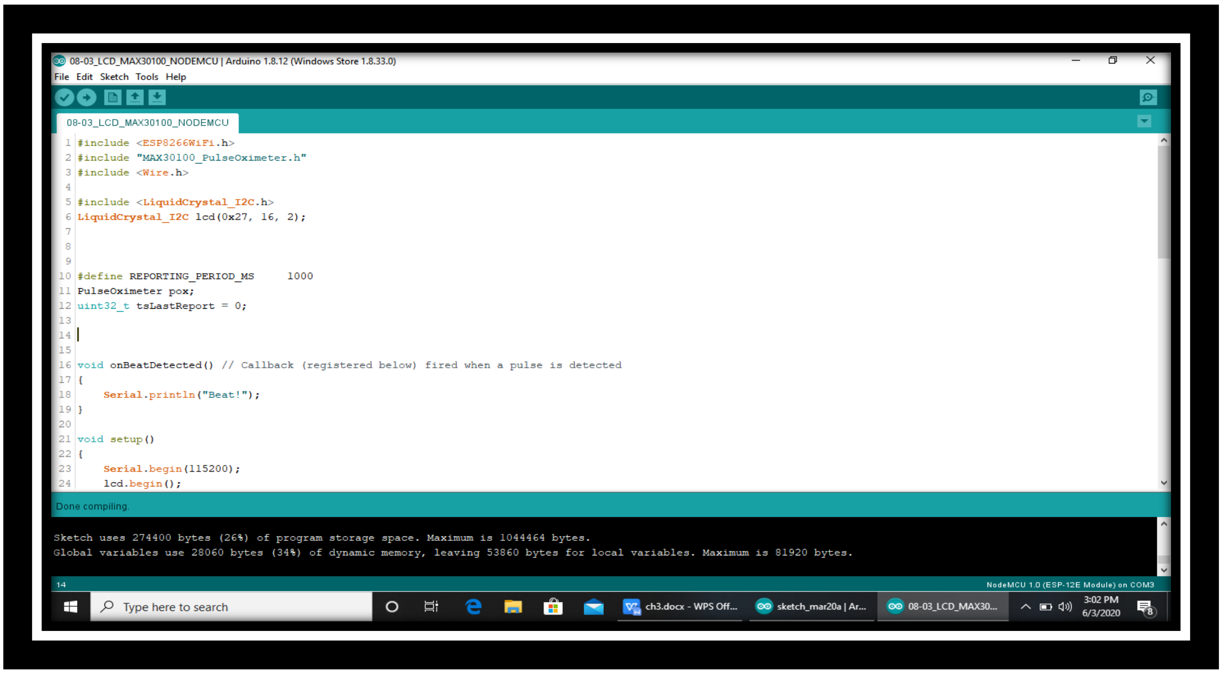


FIg3.5: Code compilation of LCD, Node MCU and MAX30100

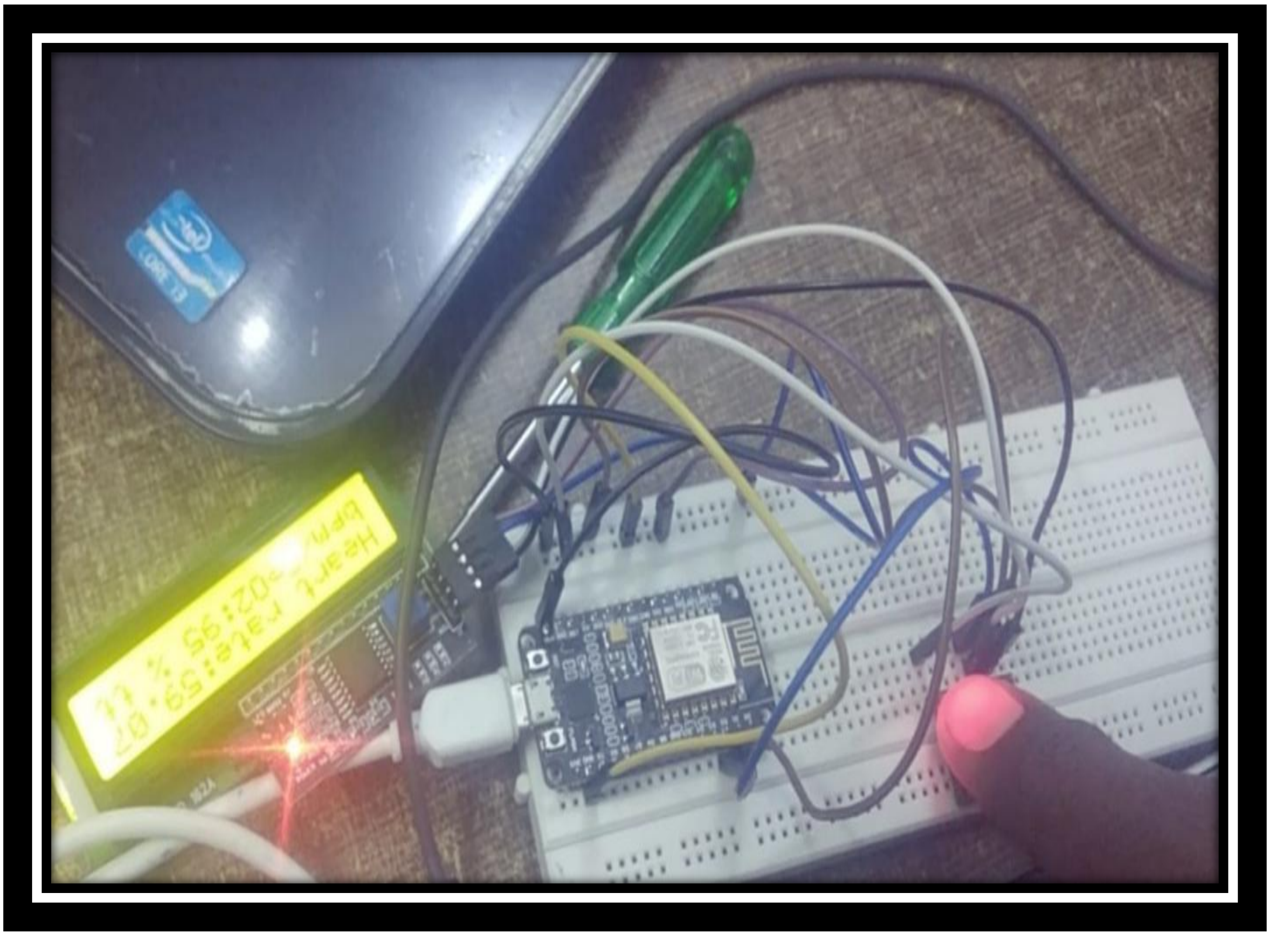


FIg3.6: Result of Spo2 and Heart rate on Display

Chapter 4

Implementation of Mobile app

4.1 Preparation of App on Blynk platform:

Blynk is open source IoT platform that provide connectivity to connect the IoT devices to them. Blynk is also provide the facility to making our own app as per our device configuration. The data in Blynk app is secured and reliable with authentication key. Features are provided by Blynk Platform is very useful and reliable. First we have to new project and then select the devices with we want to connect with Blynk application. After that we need to configure and making app as per our requirement.

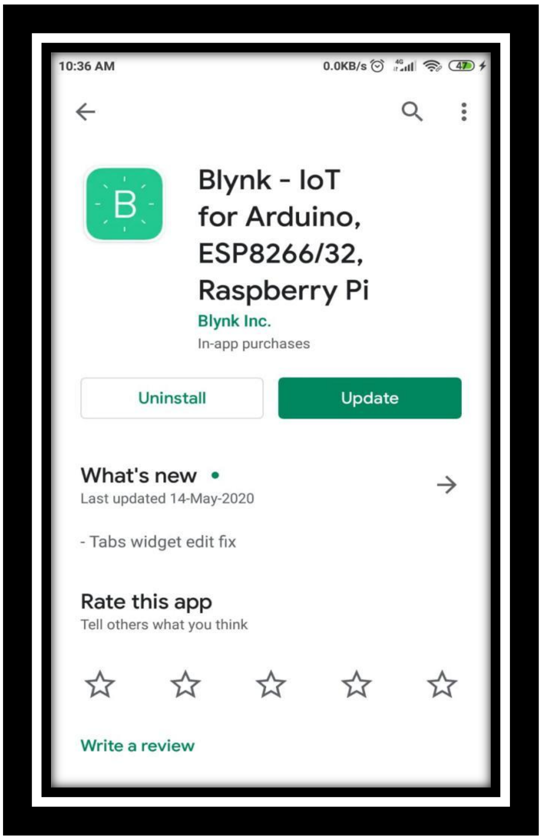


Fig 4.1: Blynk app Platform

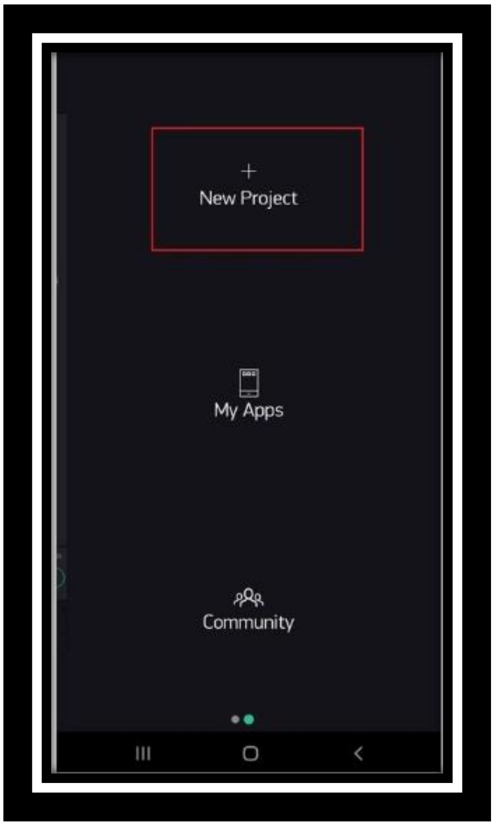


Fig 4.2: Creating a New project

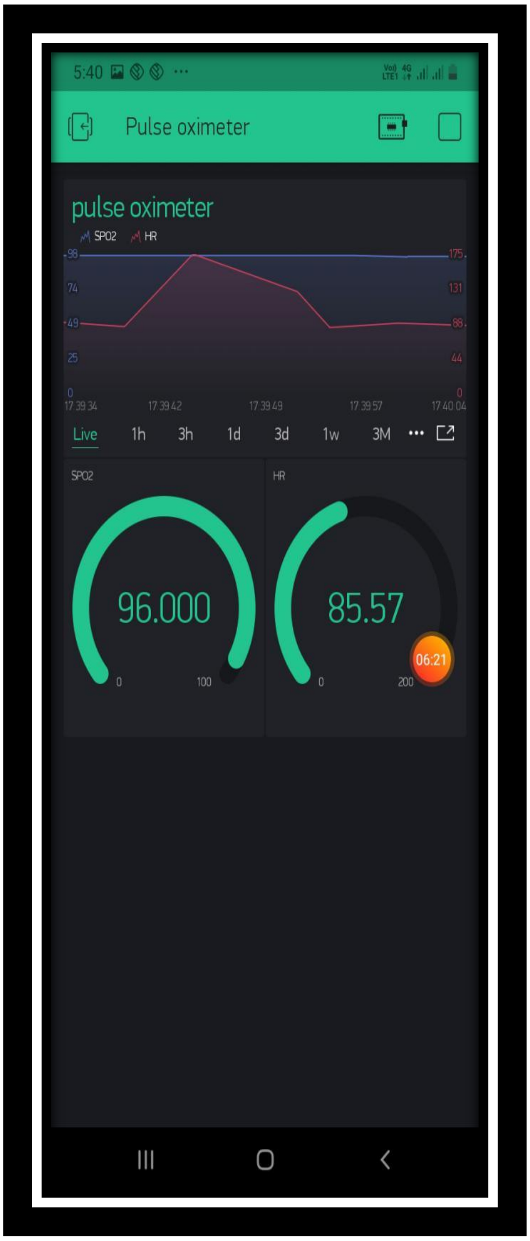


Fig 4.3: Pulse oximeter app on Blynk Platform

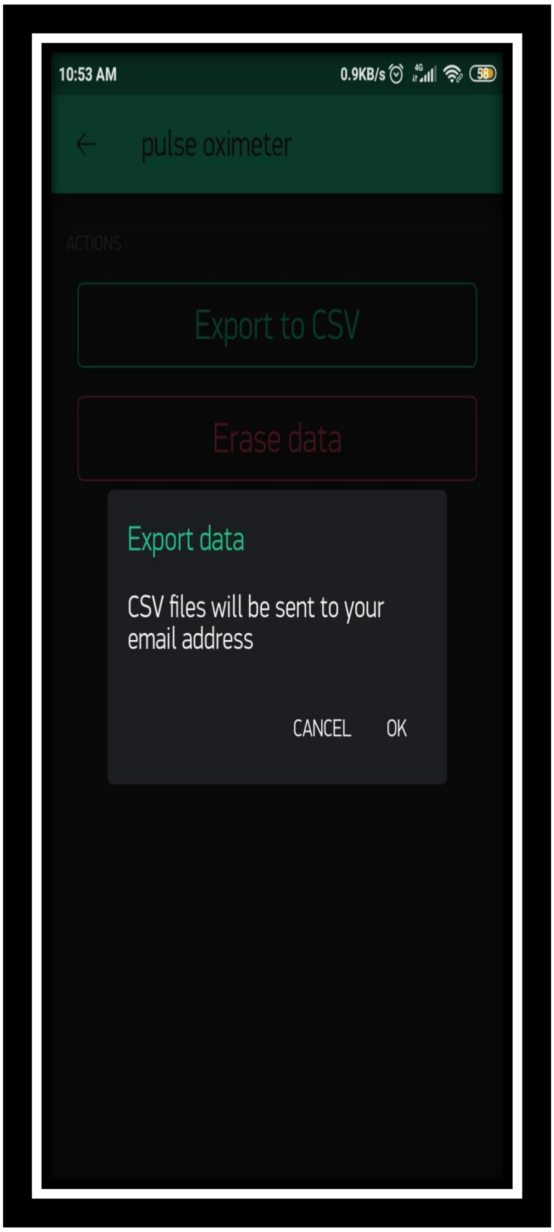


Fig 4.4: Historical data sent to mail

Fig 4.5: Code compilation on Arduino IDE for Blynk Platform

Fig 4.1 shows Blynk Platform on Google Play store. Fig 4.2 shows the way to create new project on Blynk Platform. The data of Spo2 was stored in V5 (virtual Pin) and The Data of Heart rate (HR) was stored in V6 (Virtual pin).Fig 4.3 shows the application of Pulse oximeter on Blynk Platform. The real time data of Spo2 and HR is shows in label and Historical data is shows in graphical form. The historical Data is stored in Blynk cloud. Fig 4.4 shows the historical data is sent in E-Mail for Further analysis.Fig4.5 shows the compilation is done on Arduino IDE software.

Chapter 5

Creating Health conditions on Historical data and data validation by doctor

5.1 health conditions:

The health condition are create on 1304 samples of data shown in Table 5.1.The data set has been taken from Kaggle data science and machine learning data set repository. Out of 1304 samples nine health condition has been created after analyzing the data set. After that all conditions are verified by doctor.

Health conditions	Labels	Samples	conditions
Some how reduced	1	173	SPO_2 : 92 to 95
Bradycardia and middle SPO_2	2	18	HR:<60 SPO_2 : 92 to 95
Tachycardia and middle SPO_2	3	59	HR:>100 SPO_2:92 to 95
COPD[3]	4	32	SPO_2:88 to 91
Critical Condition	5	46	SPO_2< 91 HR:<60 and >100
Bradycardia[1]	6	71	HR:<60
Tachycardia[2]	7	179	HR:>100
SPO_2 critical[5]	8	21	SPO_2< 87
Normal[6]	9	704	SPO_2 : 96 to 100 HR:60 to 100

Table 5.1: Health conditions

Tachycardia:

Tachycardia happens when person's heart beat more Than 100 time per minute. There are three types of Tachycardia.

1. Supraventricular
2. Ventricular
3. Sinus tachycardia

The main Reasons for Tachycardia are strenuous exercise, a fever, fear, stress, anxiety, certain medications, street drugs and smoking.

Bradycardia:

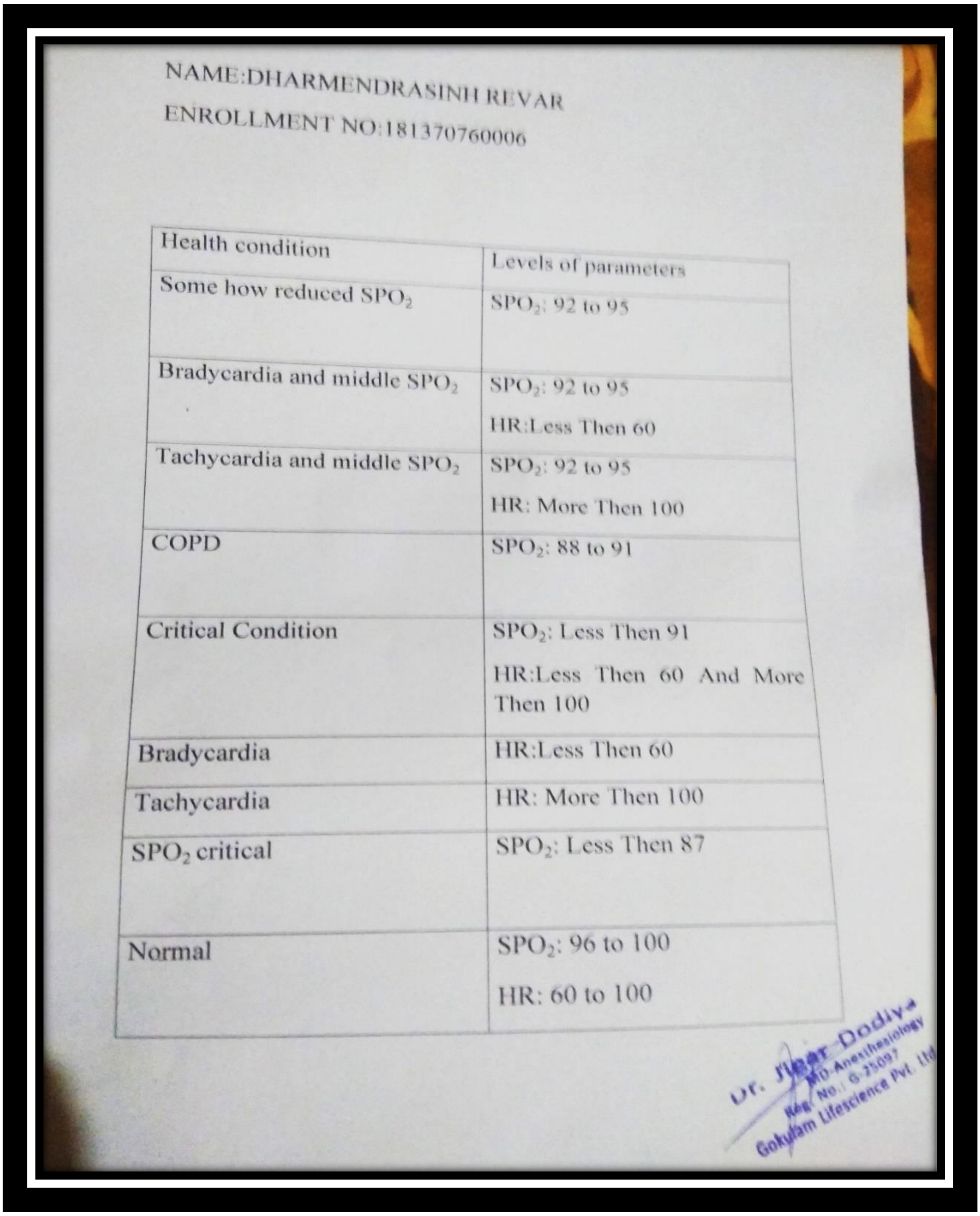
Bradycardia happen when person's heart beat slower than 60 Time per minute. Normal heart beat range is Between 60 to 100. The symptoms of Bradycardia are Fatigue, shortness of breath, Pain in chest and memory problem. The main risk factor for Bradycardia is age. Bradycardia normally happens in older age Persons.

Chronic obstructive pulmonary disease (COPD):

COPD is common lung issue. COPD person's normally have breathing problems. Smoking is the main reason for COPD. But some people do not smoke still COPD Happens. Heavy amount of pollution also cause COPD.

5.2 Data Validation by Doctor:

Data validation is done by Dr.jigar dodiya(Anesthesiology) From Gokul super speciality,Rajkot as shown in Fig 5.1



NAME:DHARMENDRASINH REVAR

ENROLLMENT NO:181370760006

Health condition	Levels of parameters
Some how reduced SPO_2	SPO_2: 92 to 95
Bradycardia and middle SPO_2	SPO_2: 92 to 95 HR:Less Then 60
Tachycardia and middle SPO_2	SPO_2: 92 to 95 HR: More Then 100
COPD	SPO_2: 88 to 91
Critical Condition	SPO_2: Less Then 91 HR:Less Then 60 And More Then 100
Bradycardia	HR:Less Then 60
Tachycardia	HR: More Then 100
SPO_2 critical	SPO_2: Less Then 87
Normal	SPO_2: 96 to 100 HR: 60 to 100

Dr. Jigar Dodiya
MD-Anesthesiology
Reg. No.: G-25097
Gokulam Lifescience Pvt. Ltd

Fig 5.1 Data validation done by doctor

Chapter 6

Machine learning algorithms of validate data

6.1 Data analysis:

Fig 6.1 shows statistics of Data set of 1304 samples like number of variables, number of observations, missing cells, duplicate rows, total size of memory, average size of memory etc.

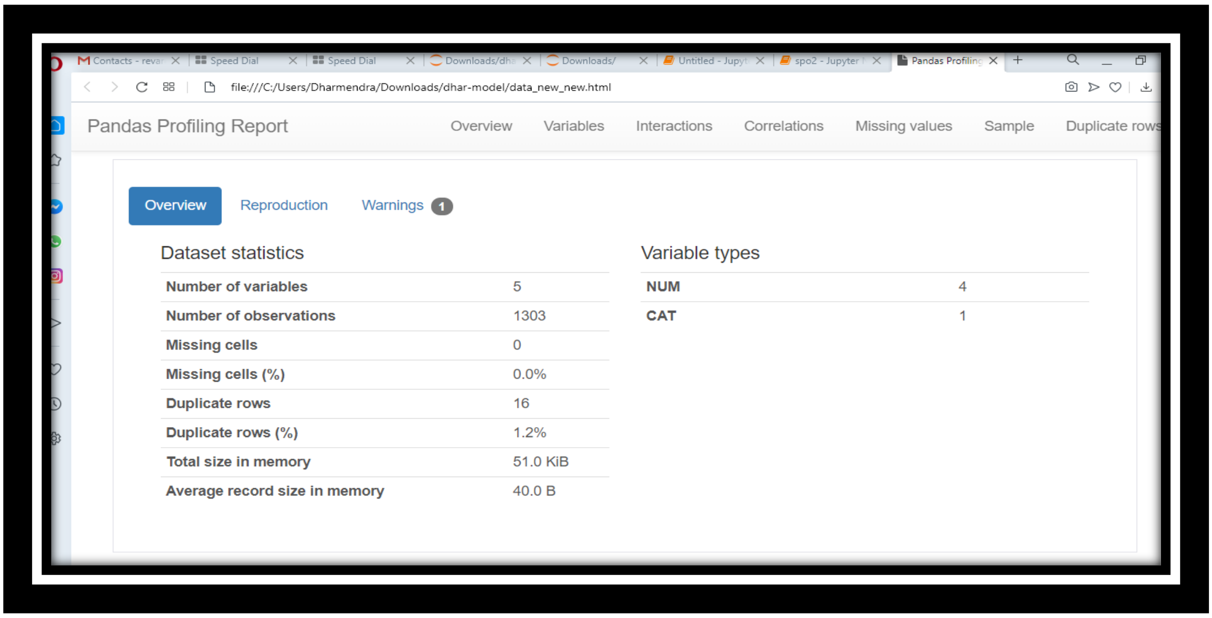


Fig 6.1: Data set statistics

Fig 6.2 Shows Genders values, its percentage and its memory size. Label 1 shows the values of male and label 2 shows the values of female.

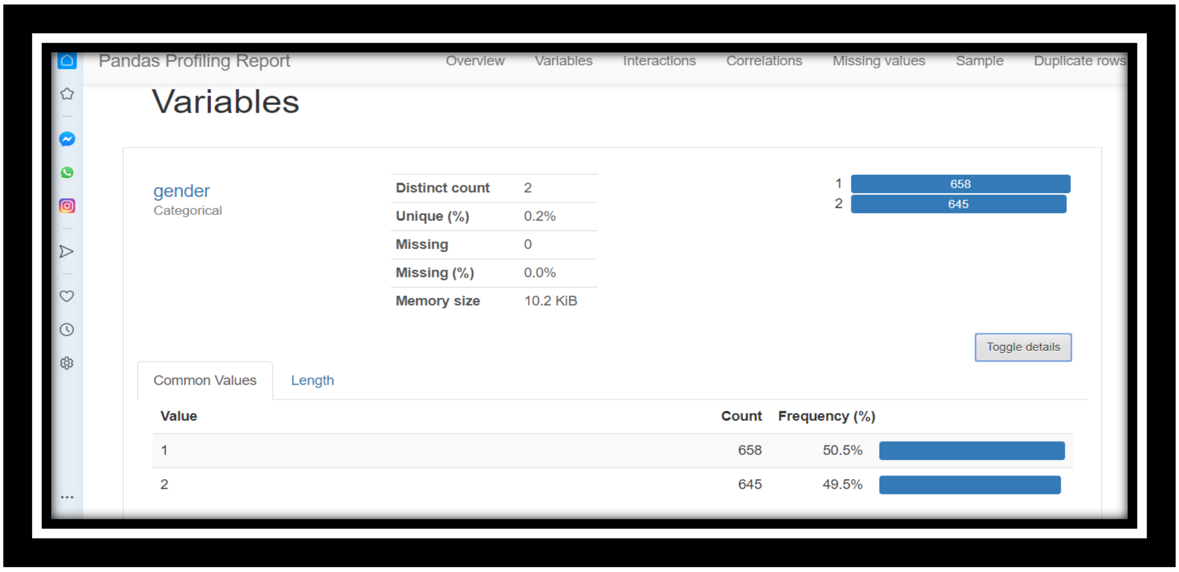
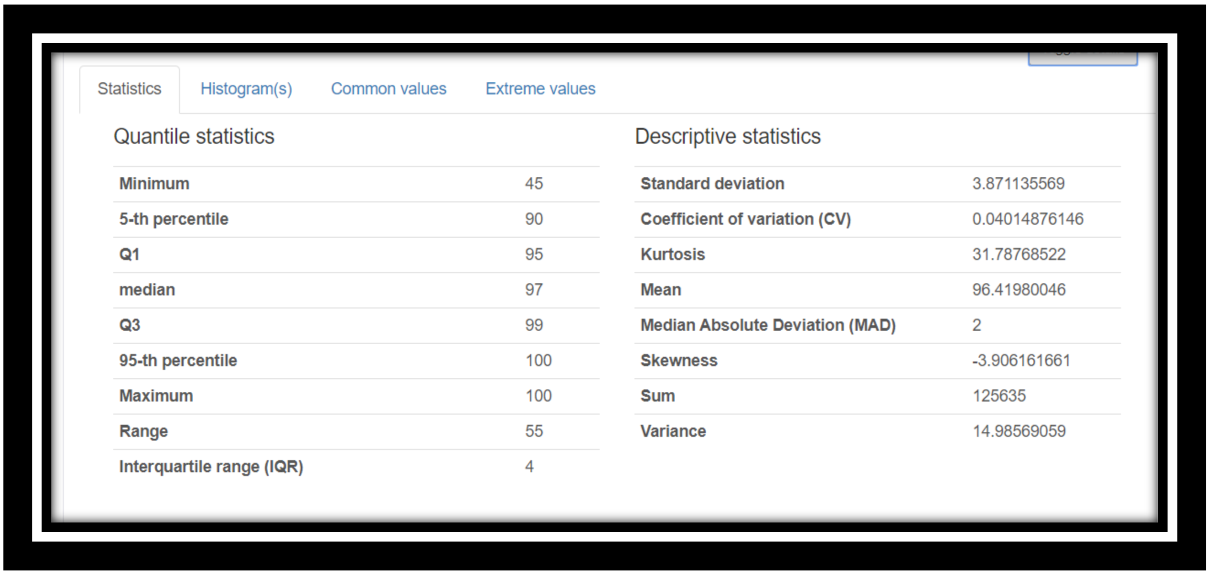


Fig 6.2: Gender count

Fig 6.3 shows details about Spo2 like mean, maximum, zeros, memory Size, minimum, zeros, unique, infinite, missing etc. the graph shows that Spo2 values on X-axis and number of samples on Y-axis.

Fig 6.3:Spo2 details

Fig 6.4 shows statistics of Spo2 like standard deviation, mean, sum, variance, skew ness, Median absolute deviation, coefficient of variation, range, Q1, Q3, kurtosis, Interquartile range etc.


Statistics Histogram(s) Common values Extreme values

Quantile statistics

Minimum	45
5-th percentile	90
Q1	95
median	97
Q3	99
95-th percentile	100
Maximum	100
Range	55
Interquartile range (IQR)	4

Descriptive statistics

Standard deviation	3.871135569
Coefficient of variation (CV)	0.04014876146
Kurtosis	31.78768522
Mean	96.41980046
Median Absolute Deviation (MAD)	2
Skewness	-3.906161661
Sum	125635
Variance	14.98569059

Fig 6.4 Spo2 Statistics

Fig 6.5 shows histogram of Spo2 Values. On the X-axis Shows Spo2 values and on Y-axis shows number of samples.

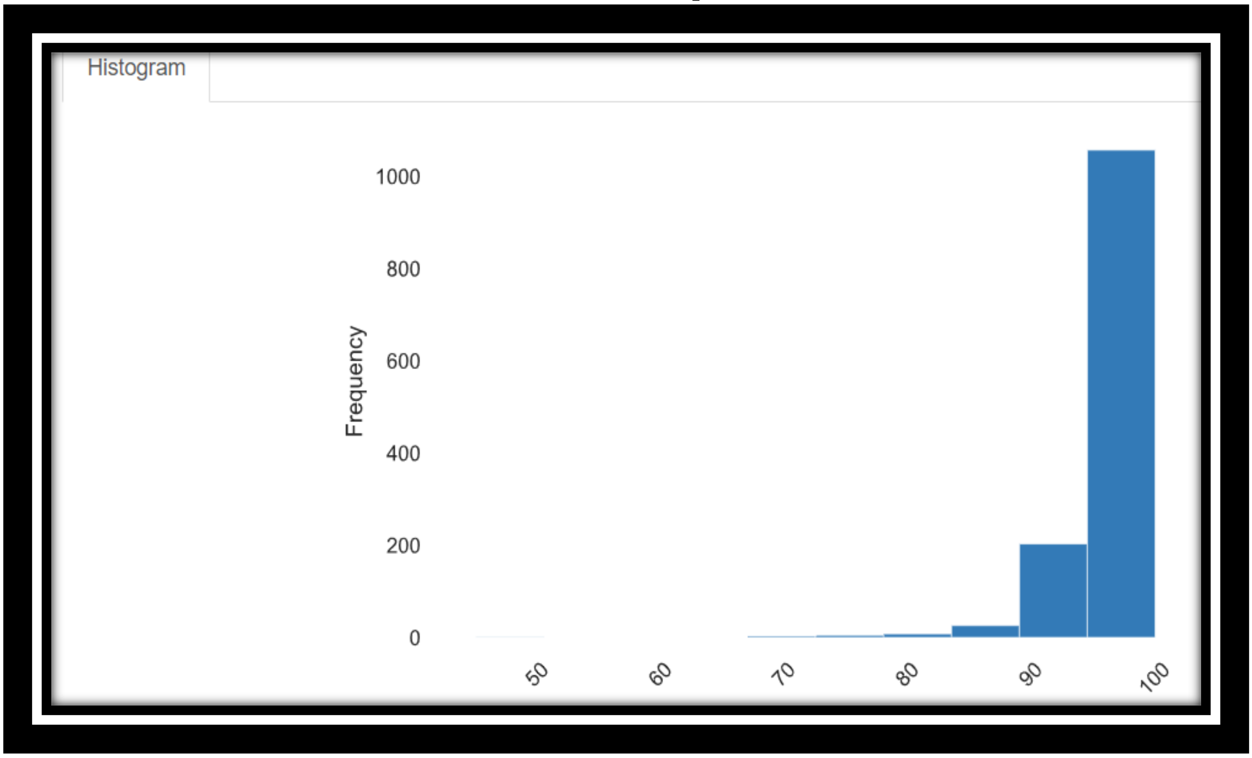
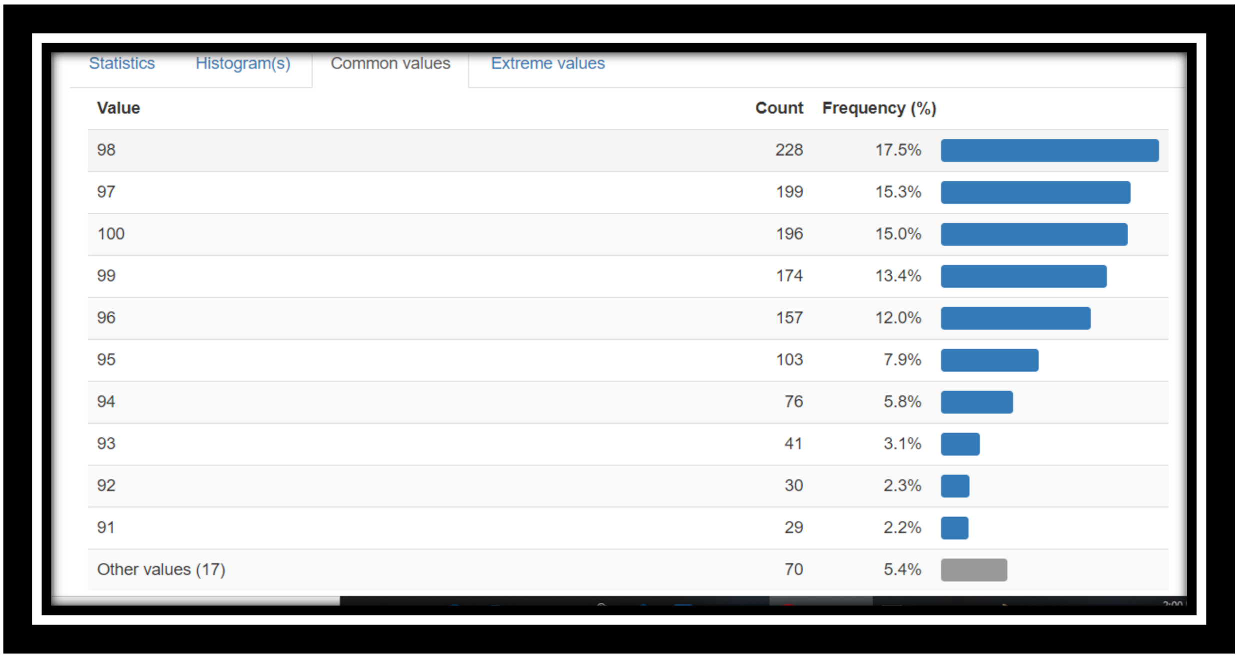


Fig 6.5 Spo2 Histogram

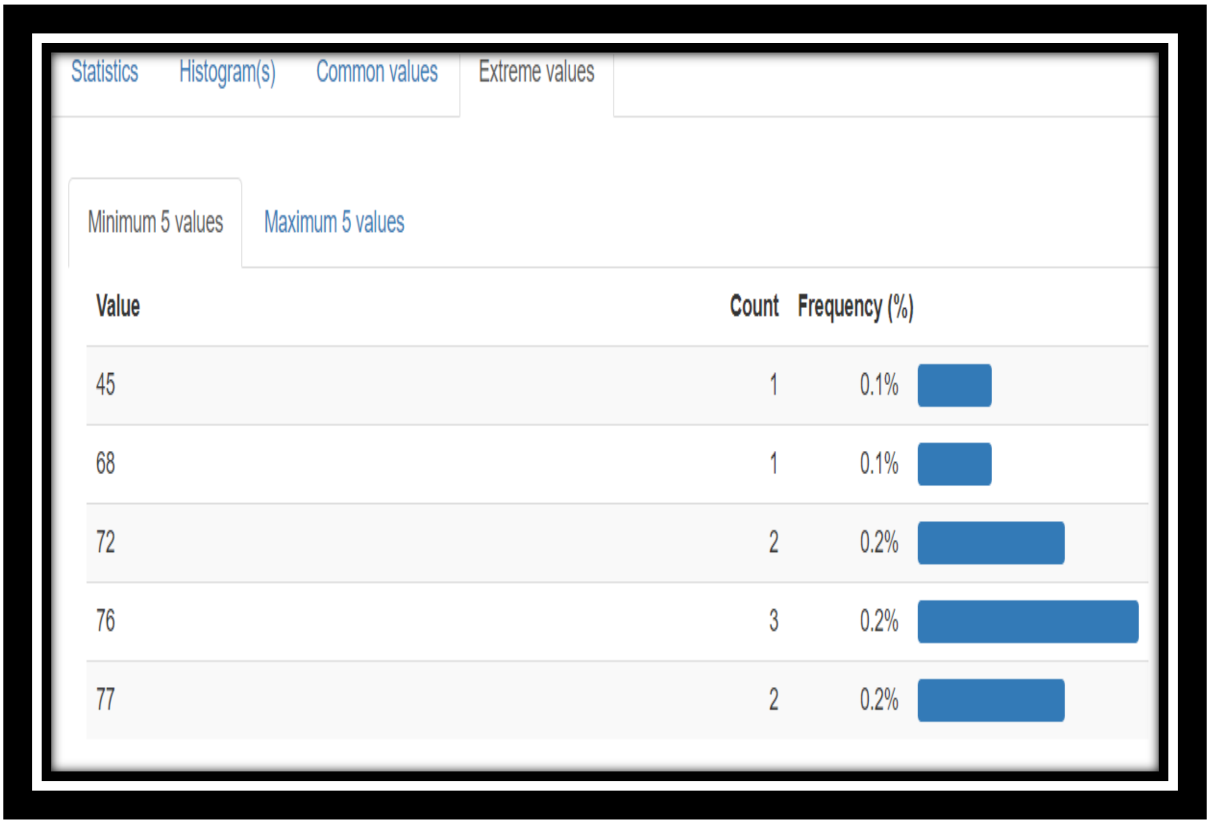
Fig 6.6 shows common values of Spo2 with its percentage.



Value	Count	Frequency (%)
98	228	17.5%
97	199	15.3%
100	196	15.0%
99	174	13.4%
96	157	12.0%
95	103	7.9%
94	76	5.8%
93	41	3.1%
92	30	2.3%
91	29	2.2%
Other values (17)	70	5.4%

Fig 6.6: Spo2 Common values

Fig 6.7 shows extreme values of Spo2 with its percentage.



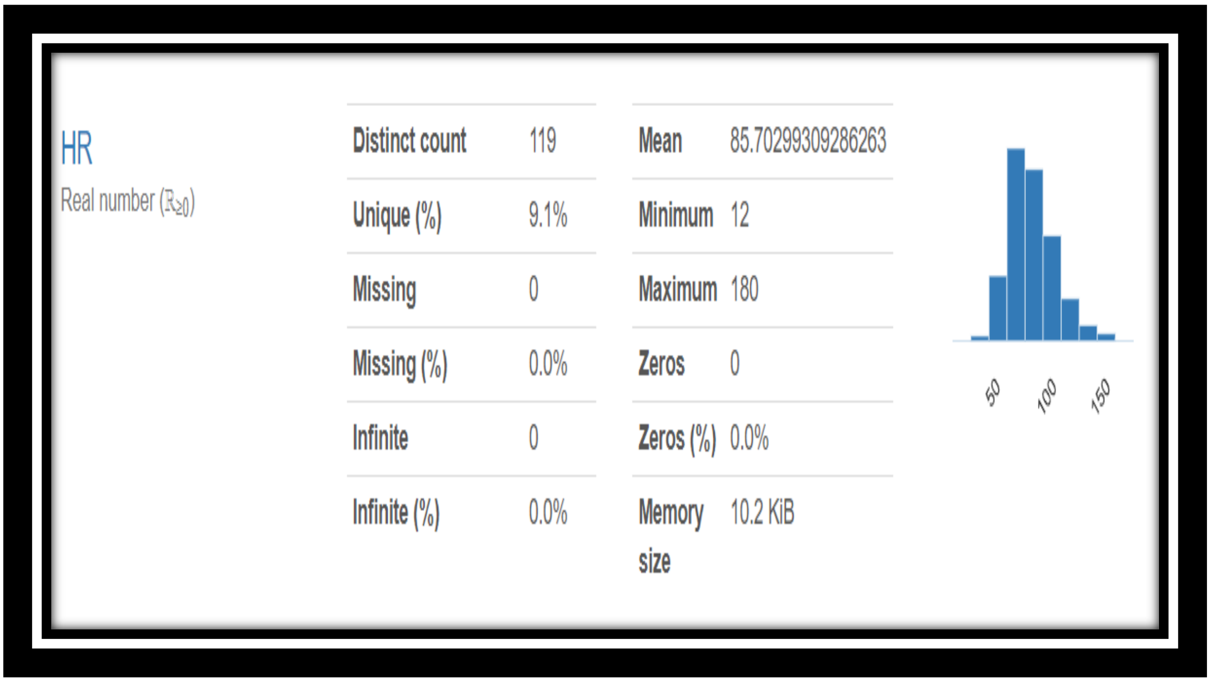
Statistics Histogram(s) Common values Extreme values

Minimum 5 values Maximum 5 values

Value	Count	Frequency (%)
45	1	0.1%
68	1	0.1%
72	2	0.2%
76	3	0.2%
77	2	0.2%

Fig 6.7:Spo2 Extreme values

Fig 6.8 shows details about HR like mean, maximum, zeros, memory Size, minimum, zeros, unique, infinite, missing etc. the graph shows that HR values on X-axis and number of samples on Y-axis.



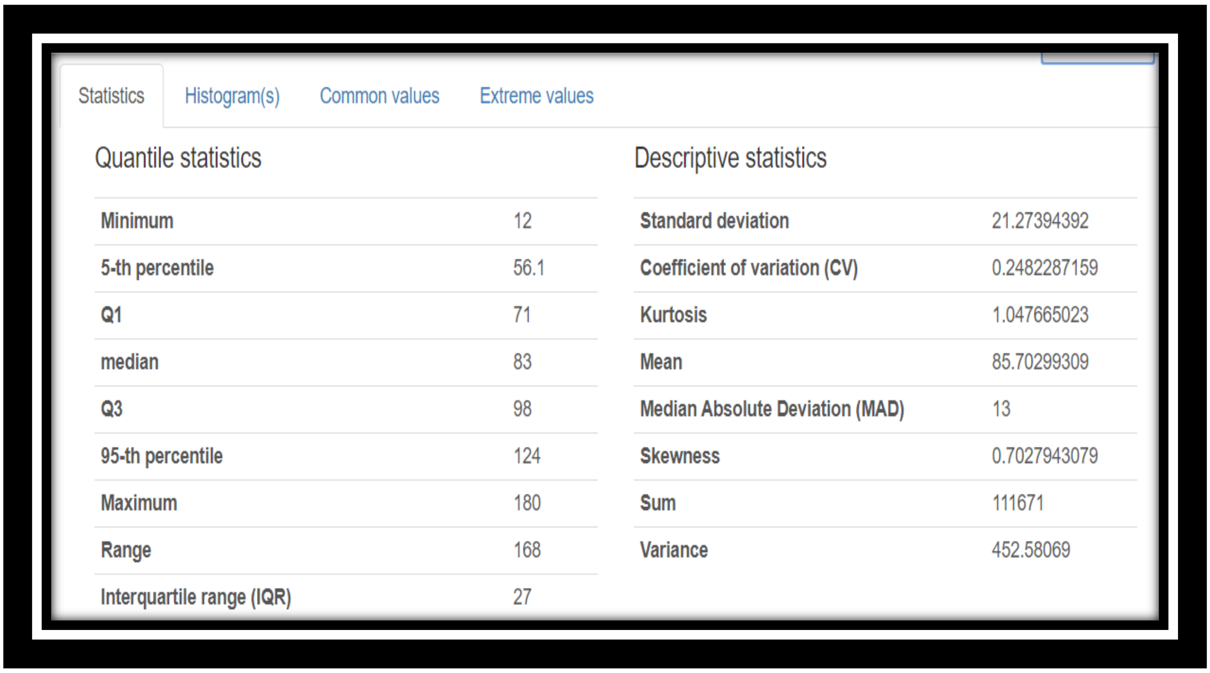
HR

Real number ($\mathbb{R}_{\geq 0}$)

Distinct count	119	Mean	85.70299309286263
Unique (%)	9.1%	Minimum	12
Missing	0	Maximum	180
Missing (%)	0.0%	Zeros	0
Infinite	0	Zeros (%)	0.0%
Infinite (%)	0.0%	Memory size	10.2 KiB

Fig 6.8: HR details

Fig 6.9 shows statistics of HR like standard deviation, mean, sum, variance, skew ness, Median absolute deviation, coefficient of variation, range, Q1, Q3, kurtosis, Interquartile range etc.



Statistics | Histogram(s) | Common values | Extreme values

Quantile statistics

Minimum	12
5-th percentile	56.1
Q1	71
median	83
Q3	98
95-th percentile	124
Maximum	180
Range	168
Interquartile range (IQR)	27

Descriptive statistics

Standard deviation	21.27394392
Coefficient of variation (CV)	0.2482287159
Kurtosis	1.047665023
Mean	85.70299309
Median Absolute Deviation (MAD)	13
Skewness	0.7027943079
Sum	111671
Variance	452.58069

Fig 6.9: HR Statistics

Fig 6.10 shows histogram of HR Values. On the X-axis Shows Spo2 values and on Y-axis shows number of samples.

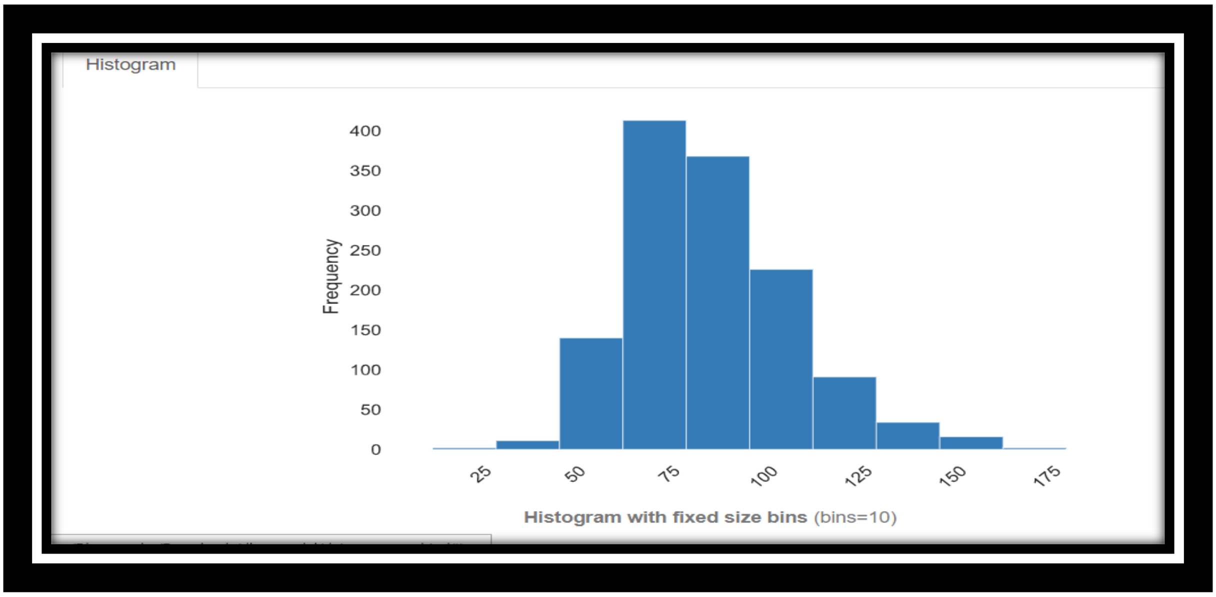


Fig 6.10: HR histogram

Fig 6.11 shows common values of HR with its percentage.

Statistics | Histogram(s) | Common values | Extreme values

Value	Count	Frequency (%)
76	41	3.1%
78	39	3.0%
90	33	2.5%
72	33	2.5%
74	32	2.5%
70	31	2.4%
82	30	2.3%
89	29	2.2%
80	27	2.1%
79	27	2.1%
Other values (109)	981	75.3%

Fig 6.11 Spo2 common values

Fig 6.12 shows extreme values of HR with its percentage.

Statistics | Histogram(s) | Common values | Extreme values

Minimum 5 values | Maximum 5 values

Value	Count	Frequency (%)
12	1	0.1%
22	1	0.1%
30	1	0.1%
36	3	0.2%
38	1	0.1%

Fig 6.12: Spo2 extreme values

Fig 6.13 shows details about HR like mean, maximum, zeros, memory Size, minimum, zeros, unique, infinite, missing etc. the graph shows that age values on X-axis and number of samples on Y-axis.

age

Real number ($\mathbb{R}_{\geq 0}$)

Distinct count	81	Mean	64.73369148119724
Unique (%)	6.2%	Minimum	18
Missing	0	Maximum	99
Missing (%)	0.0%	Zeros	0
Infinite	0	Zeros (%)	0.0%
Infinite (%)	0.0%	Memory size	10.2 KiB

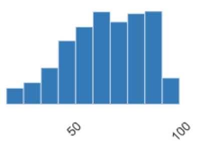
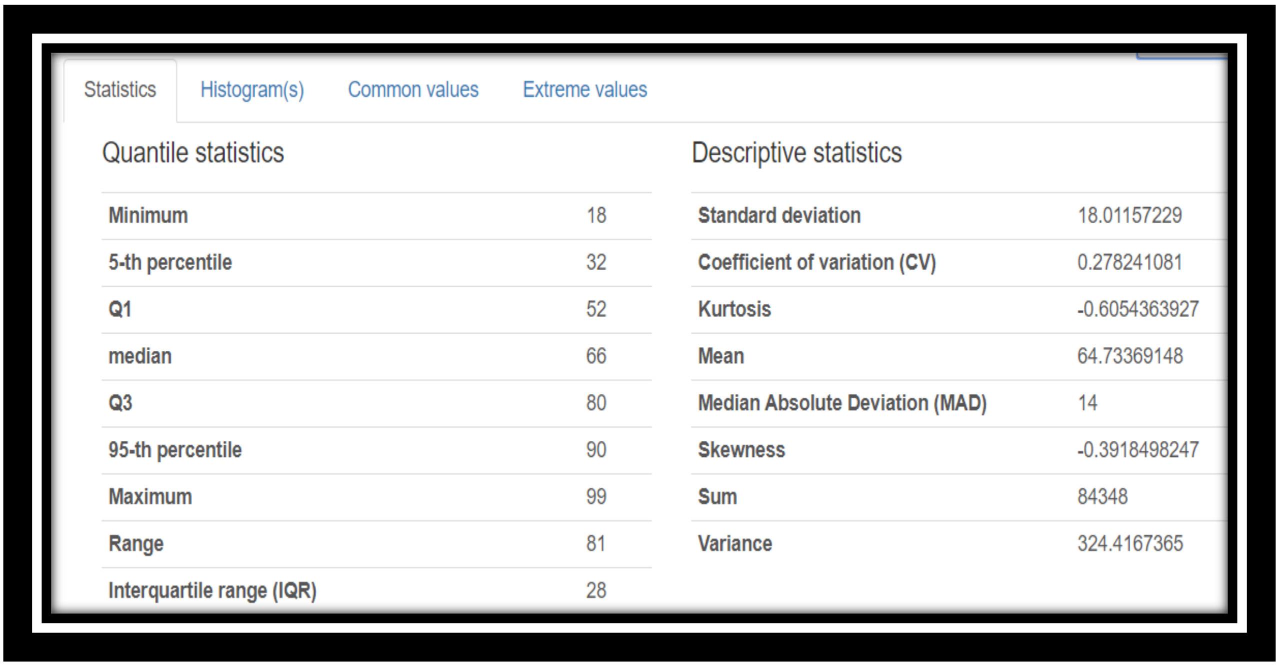


Fig 6.13: age details

Fig 6.14 shows statistics of age like standard deviation, mean, sum, variance, skew ness, Median absolute deviation, coefficient of variation, range, Q1, Q3, kurtosis, Interquartile range etc.



Statistics | Histogram(s) | Common values | Extreme values

Quantile statistics

Minimum	18
5-th percentile	32
Q1	52
median	66
Q3	80
95-th percentile	90
Maximum	99
Range	81
Interquartile range (IQR)	28

Descriptive statistics

Standard deviation	18.01157229
Coefficient of variation (CV)	0.278241081
Kurtosis	-0.6054363927
Mean	64.73369148
Median Absolute Deviation (MAD)	14
Skewness	-0.3918498247
Sum	84348
Variance	324.4167365

Fig 6.14: age statistics

Fig 6.10 shows histogram of age Values. On the X-axis Shows Spo2 values and on Y-axis shows number of samples.

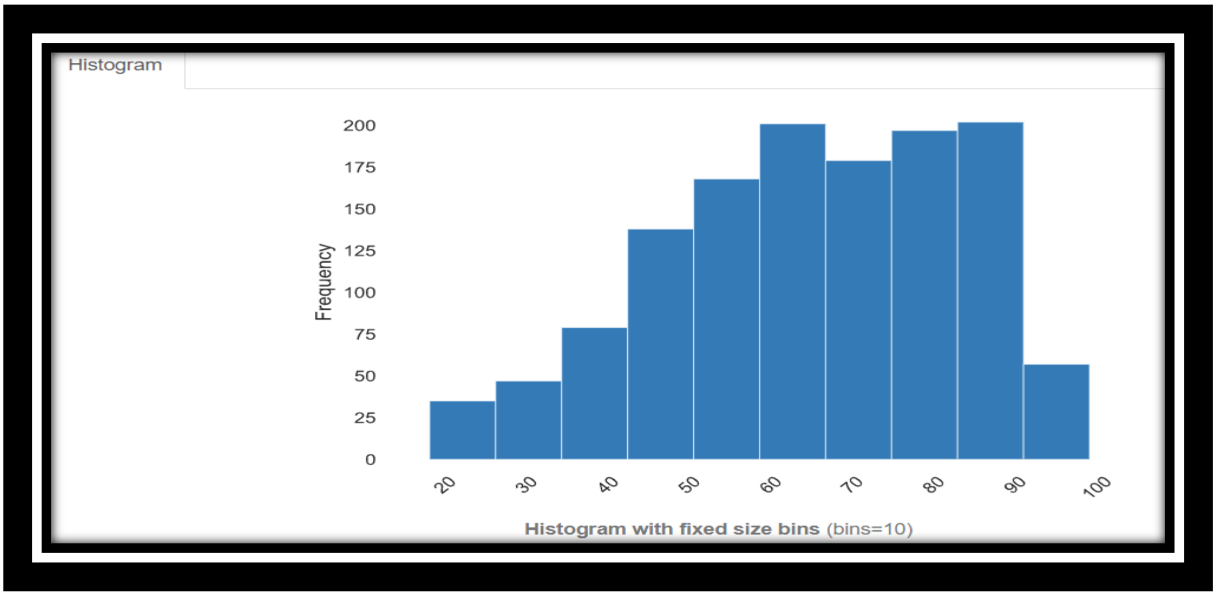


Fig 6.15: Age histogram

Fig 6.16 shows common values of Age with its percentage.

Statistics | Histogram(s) | Common values | Extreme values

Value	Count	Frequency (%)
60	38	2.9%
85	35	2.7%
84	34	2.6%
80	31	2.4%
74	30	2.3%
79	29	2.2%
66	28	2.1%
56	28	2.1%
83	28	2.1%
63	27	2.1%
Other values (71)	995	76.4%

Fig 6.16: Age common values

Fig 6.17 shows common values of Age with its percentage

Statistics | Histogram(s) | Common values | Extreme values

Minimum 5 values | Maximum 5 values

Value	Count	Frequency (%)
18	3	0.2%
19	2	0.2%
20	2	0.2%
21	6	0.5%
22	1	0.1%

Fig 6.17: Age extreme values

Fig 6.18 shows details about health like mean, maximum, zeros, memory Size, minimum, zeros, unique, infinite, missing etc. the graph shows that HR values on X-axis and number of samples on Y-axis.

health
Real number ($\mathbb{R}_{\geq 0}$)

Distinct count	9	Mean	6.85111281657713
Unique (%)	0.7%	Minimum	1
Missing	0	Maximum	9
Missing (%)	0.0%	Zeros	0
Infinite	0	Zeros (%)	0.0%
Infinite (%)	0.0%	Memory size	10.2 KiB

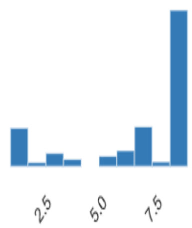


Fig 6.18: Health details

Fig 6.14 shows statistics of health like standard deviation, mean, sum, variance, skew ness, Median absolute deviation, coefficient of variation, range, Q1, Q3, kurtosis, Interquartile range etc.

Statistics | Histogram(s) | Common values | Extreme values

Quantile statistics

Minimum	1
5-th percentile	1
Q1	5
median	9
Q3	9
95-th percentile	9
Maximum	9
Range	8
Interquartile range (IQR)	4

Descriptive statistics

Standard deviation	2.903083255
Coefficient of variation (CV)	0.423738936
Kurtosis	-0.3622594609
Mean	6.851112817
Median Absolute Deviation (MAD)	0
Skewness	-1.07330021
Sum	8927
Variance	8.427892386

Fig 6.19: health statistics

Fig 6.20 shows interaction of spo2 vs Spo2.

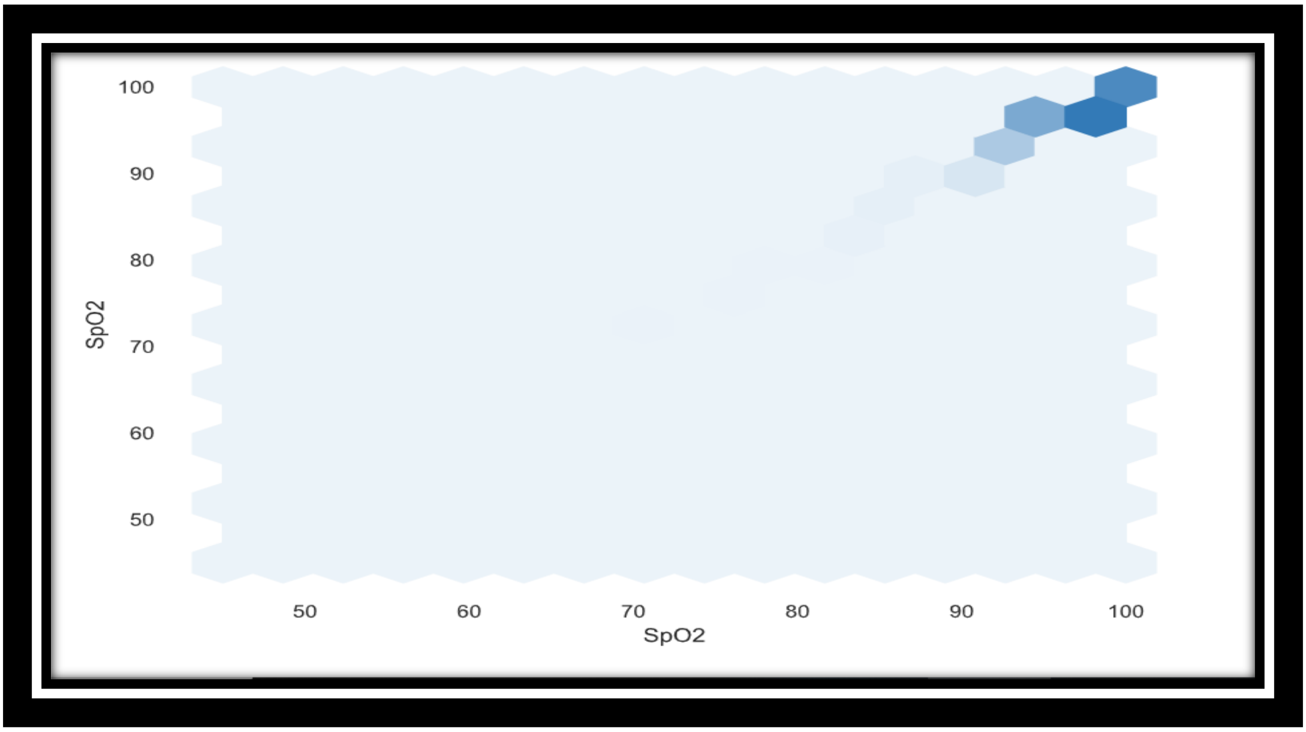


Fig 6.20:Spo2 interactions

Fig 21 show interaction Spo2 vs HR. On Y-axis shows HR values and on X-axis Shows Spo2 values.

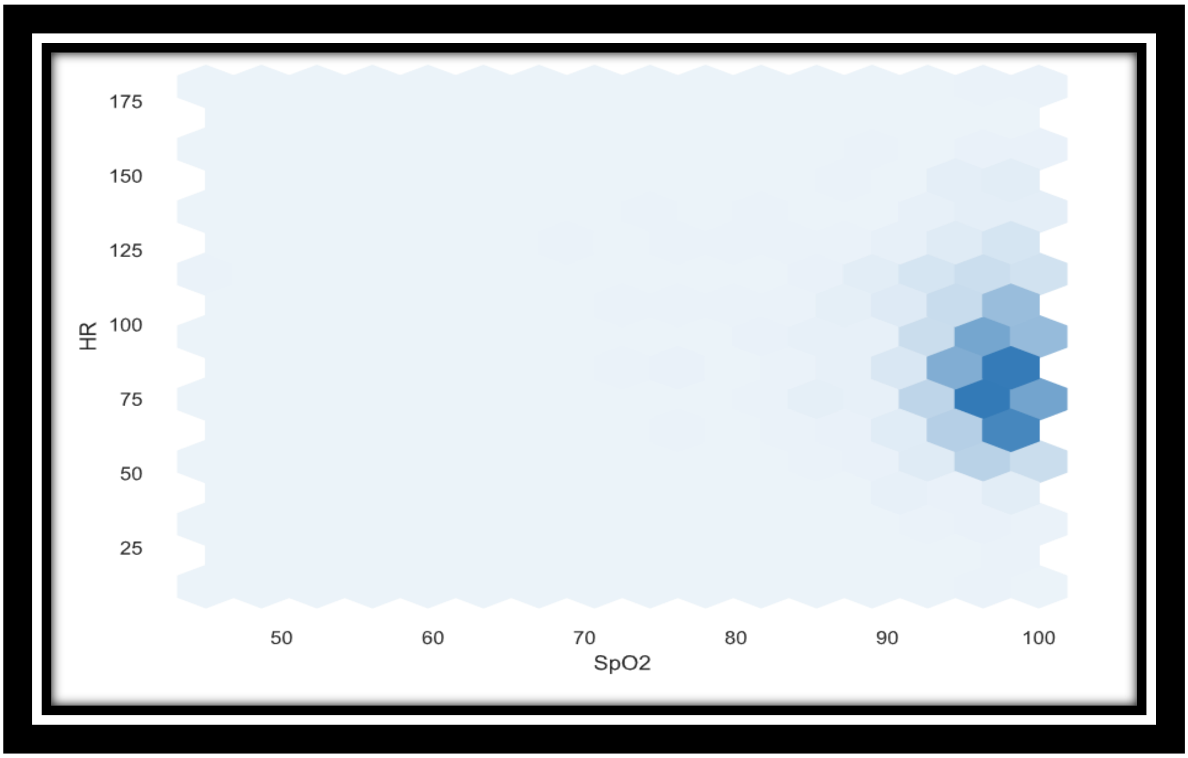


Fig 6.21:Spo2 and HR interactions

Fig 22 show interaction Spo2 vs age. On Y-axis shows age values and on X-axis Shows Spo2 values.

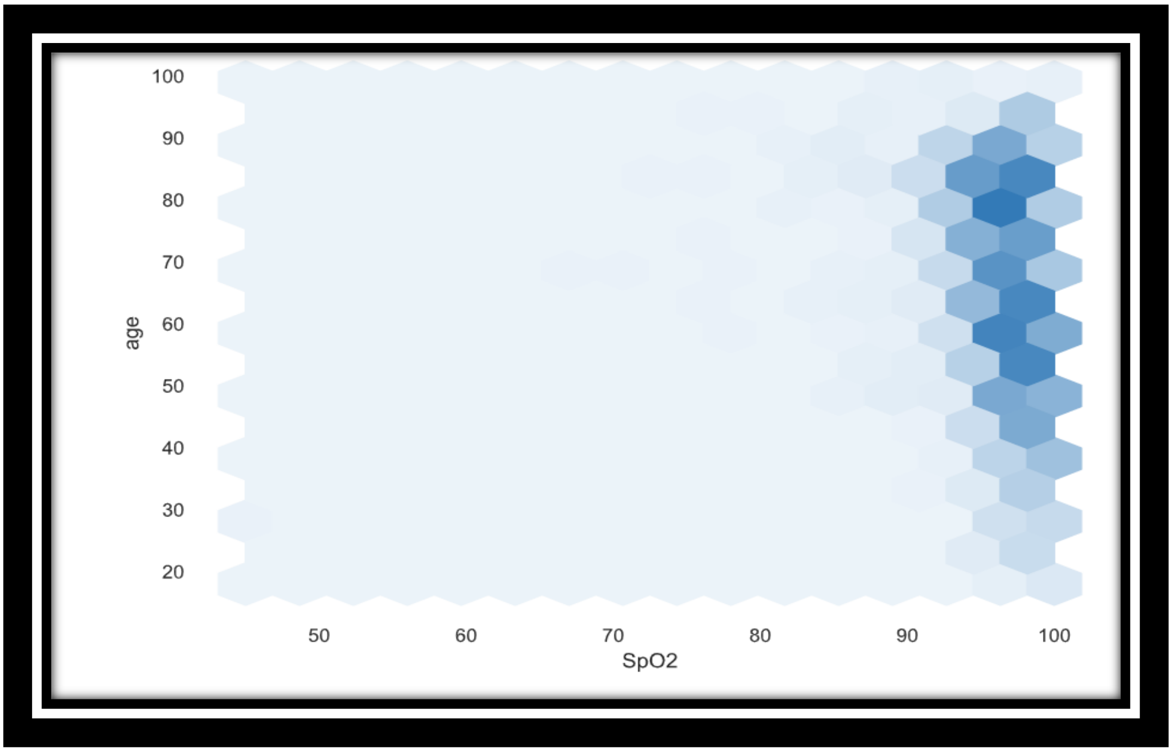


Fig 6.22: Age and Spo2 interactions

Fig 23 show interaction Spo2 vs age. On Y-axis shows health values and on X-axis Shows Spo2 values.

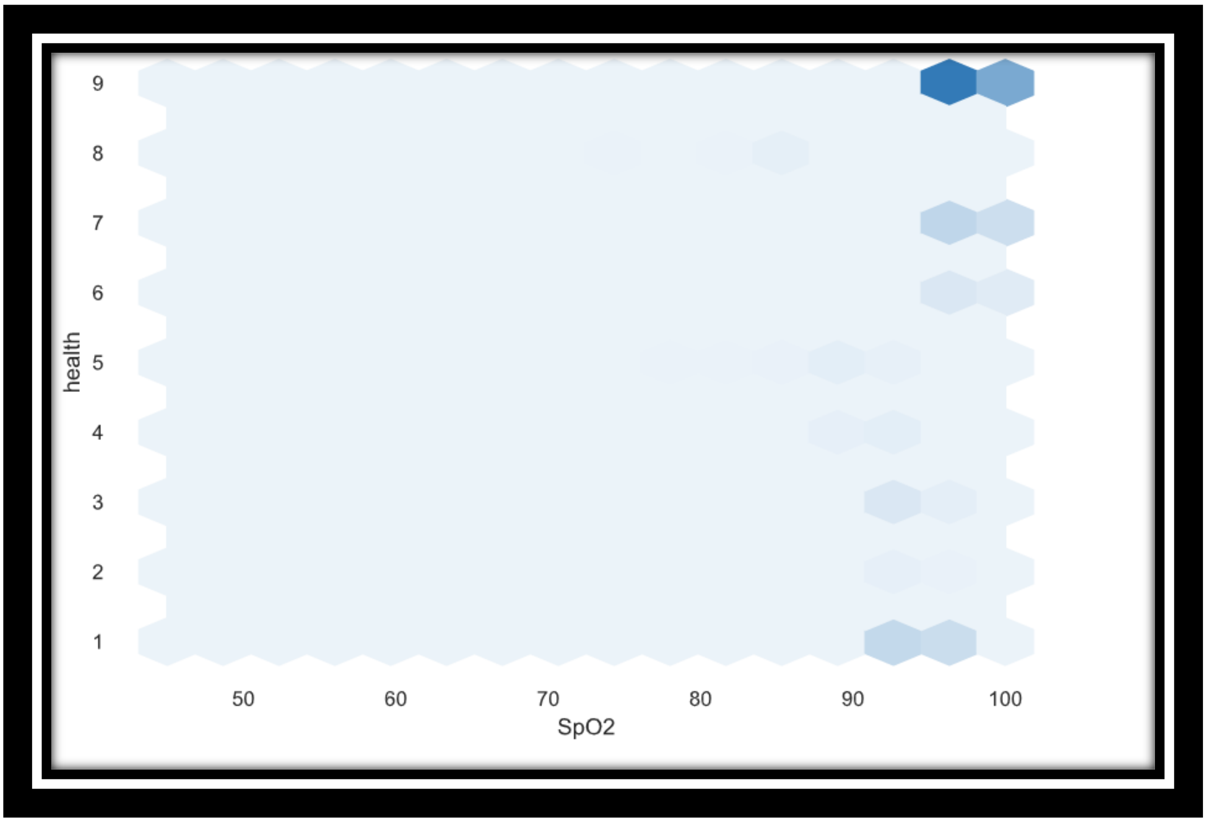


Fig 6.23: Health and Spo2 interactions

Fig 6.24: First rows values out of 1304 samples.

First rows

	gender	SpO2	HR	age	health
0	1	93	82	68	1
1	1	98	72	61	9
2	2	100	91	78	9
3	1	98	59	58	6
4	1	100	63	78	9
5	2	98	74	86	9
6	2	97	84	60	9
7	2	98	90	56	9
8	2	100	75	56	9
9	1	96	96	60	9

Fig 6.24: First rows values

Fig 6.25: Last rows values out of 1304 samples.

Last rows

	gender	SpO2	HR	age	health
1293	1	99	76	37	9
1294	2	93	130	55	3
1295	2	98	56	57	6
1296	2	97	109	63	7
1297	1	91	94	58	4
1298	2	96	70	63	9
1299	1	76	137	62	5
1300	2	72	82	81	8
1301	1	98	82	62	9
1302	2	97	76	82	9

Fig 6.25: Last rows values

Fig 6.26: Most frequent (Duplicate) values out of 1304 samples.

Most frequent

	gender	SpO2	HR	age	health	count
0	1	91	91	81	4	2
1	1	94	82	83	1	2
2	1	96	76	79	9	2
3	1	96	78	65	9	2
4	1	97	60	61	9	2
5	1	98	71	91	9	2
6	1	98	76	55	9	2
7	1	98	106	56	7	2
8	1	99	80	64	9	2
9	1	99	89	46	9	2

Fig 6.26: Duplicate values

The Figures are given above are from the report of EDA (exploratory data analysis) from the validate data set of 1304 samples. It is also called Pandas profiling Report.

6.2 Data analysis: Algorithms Comparison

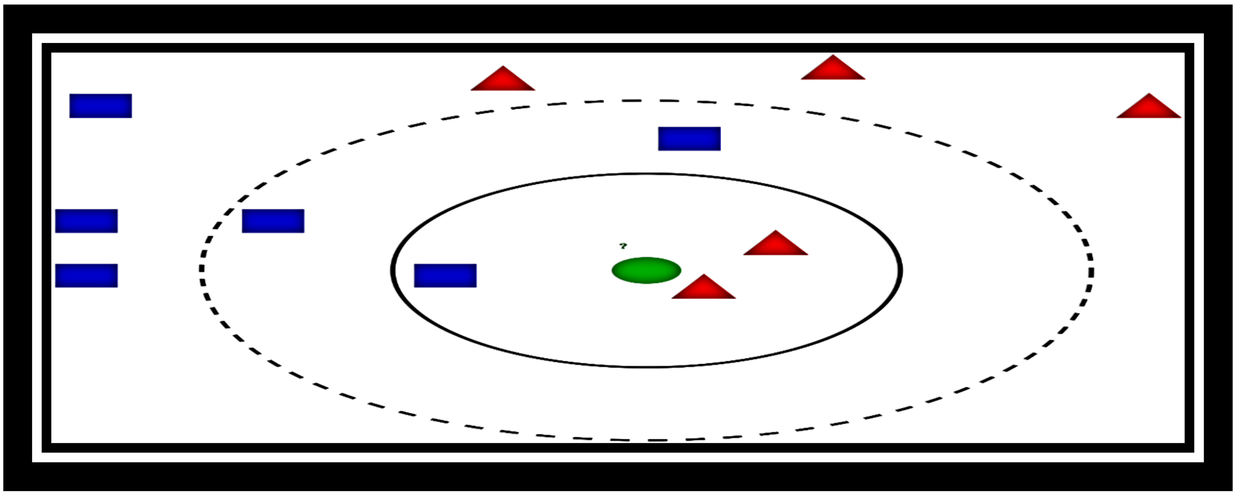
The features are in the validate data set Spo2, Gender, heart rate (HR) and age. The labels is health. The software used for performing algorithm is Jupyter Notebook. The codes are written in python language. In the validate data set 25% data was taken for testing and 75% data was taken for training. The Aim is classify the health. Three main algorithms for classification is K Nearest Neighbor, Random Forest and Decision tree. The comparison on three algorithm is shown in Table 6.1 on the parameters are MAE (mean absolute error), Average Baseline error, RMSE (root mean square error) and Accuracy.

parameters	K-nearest neighbor	Decision Tree	Random Forest
Mean absolute error	1.12	0.0	0.01
Average baseline error	57.74	57.74	57.74
Root mean square error	2.74674	0.0	0.040
Accuracy	18.46%	100%	99.81%

Table 6.1: Parameters based on Algorithms

6.3 K nearest neighbor (KNN):

KNN is also supervised learning algorithm. KNN is used for classification purpose. KNN is also known as lazy algorithm because it is only summarized learning process and do not learn or take its own decision itself. KNN basically identifies the data points that are separated into various classes to predict the classification of a given sample point. This predictions or classifications are done based on similarity measure. The similarity measure means KNN finds its nearest points based on Euclidean distance. The number of nearest points to find are defined by the value of K. suppose defined K value is 5, then KNN finds five nearest sample points from the test point and checks the labels of that five nearest points. KNN predicts or gives the classification output for the test sample based on the majority class of labels. For example, for K = 3 and for binary classification, two labels are of same class and third is of another class. Then KNN predicts the majority class label as output class. This can be understood by showing figure below.



Sample point

Class A

Class B

Fig 6.27: KNN Algorithm

```
In [23]: # The baseline predictions are the historical averages
         baseline_preds = test_features[:, feature_list.index('age')]
         # Baseline errors, and display average baseline error
         baseline_errors = abs(baseline_preds - test_labels)
         print('Average baseline error: ', round(np.mean(baseline_errors), 2))

         Average baseline error:  57.74
```

Fig 6.28: Average baseline error of KNN

```
In [34]: from sklearn.metrics import mean_squared_error
         mse = mean_squared_error(test_labels, predictions)
         rmse = np.sqrt(mse)

In [35]: rmse
Out[35]: 2.7464422049764807

In [36]: #testset = [[test_features]]
         #predictions = [[test_labels]]
         #accuracy = getAccuracy(testset, predictions)
         #print(accuracy)

In [37]: mape = 100 * (errors / test_labels)
         accuracy = 100 - np.mean(mape)
         print('Accuracy:', round(accuracy, 2),'%.')
         Accuracy: 18.46 %.
```

Fig 6.29: RMSE and Accuracy of KNN

```
In [25]: # we imported the model using sci-learn kit

         # Use the forest's predict method on the test data
         predictions = rf.predict(test_features)
         # Calculate the absolute errors
         errors = abs(predictions - test_labels)
         # Print out the mean absolute error (mae)
         print('Mean Absolute Error:', round(np.mean(errors), 2), '')
         print(errors)
         #print(n12)

         Mean Absolute Error: 1.12
         [0 0 0 0 3 0 0 0 0 0 0 0 8 0 0 0 0 0 0 0 0 3 0 0 7 0 0 0 0 1 0 8 0 0 0 0 1
          8 8 7 0 0 0 0 0 8 0 0 8 0 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 8 0 0 0 0
          0 0 8 0 0 8 0 0 0 0 0 0 0 8 8 1 8 0 0 0 8 0 0 0 0 6 0 0 0 8 0 0 2 0 0 0 0
          0 0 0 8 8 0 0 4 0 0 0 0 0 4 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 8 0 0 0 0
          0 0 0 8 0 0 0 4 0 0 0 0 0 0 0 0 0 0 0 0 3 2 0 4 4 0 4 8 0 0 0 0 8 0 4 0 0
          4 0 0 0 0 0 0 0 0 0 4 3 0 8 0 0 0 4 8 4 0 0 0 0 0 2 0 0 0 0 0 8 0 0 0 0 0
          0 0 0 0 0 0 0 0 0 0 0 0 0 4 4 0 0 0 0 0 0 0 0 0 4 0 0 0 0 1 0 0 0 8 0 0 0
          0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 4 0 8 0 0 0 0 0 0 4 0 0 0 0 0 0 8
          0 8 8 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 8 0 8 0 2 0 8 0]
```

Fig 6.30: MAE of KNN

6.4: Decision Tree (DT):

Decision tree algorithm is same as tree type structure. Decision tree algorithm is used for both the classification as well as regression problem. Decision tree algorithm mostly depends on the independent variables. Independent variables are taken as master node. This master node will have conditions over the features of the dataset. This master node has condition through which it decides on which leaf node to navigate next. Once the last leaf node is reached after satisfying all the conditions, the output is predicted. The better the sequence of conditions the better the output of algorithm. The term Information Gain is used to select the conditions in nodes. To derive the tree structure of the algorithm, a greedy, recursive based algorithm is used.as shown in FIG.6.31 the Decision will be taken based on Predicted score.

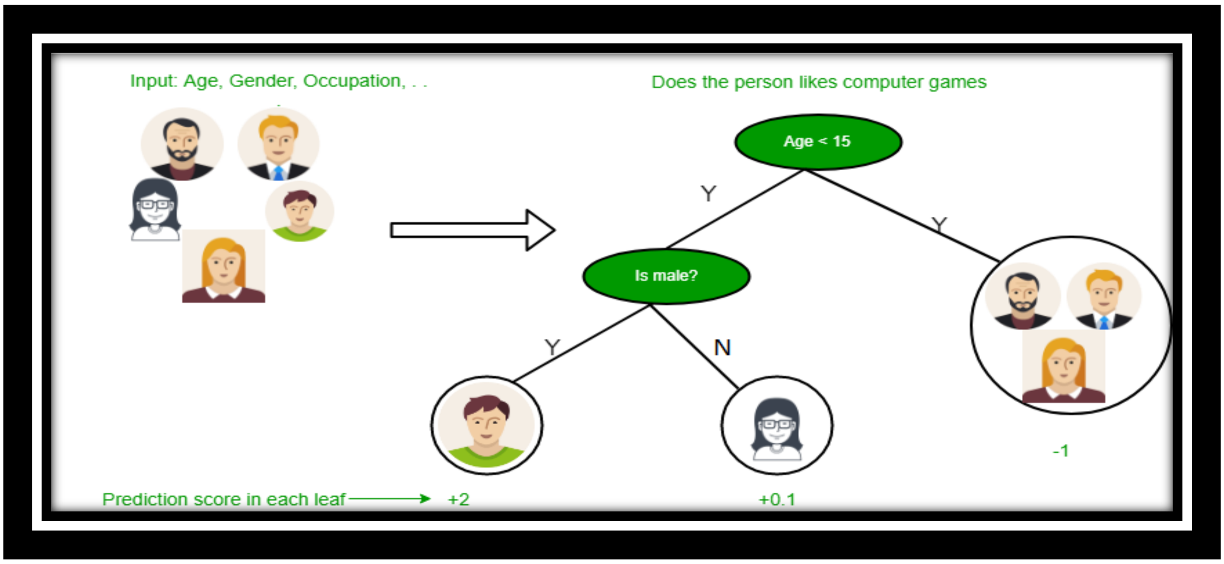


Fig 6.31: Decision Tree Algorithm

```
In [23]: # The baseline predictions are the historical averages
         baseline_preds = test_features[:, feature_list.index('age')]
         # Baseline errors, and display average baseline error
         baseline_errors = abs(baseline_preds - test_labels)
         print('Average baseline error: ', round(np.mean(baseline_errors), 2))

         Average baseline error:  57.74
```

Fig 6.32: Average baseline error of Decision Tree

```
In [34]: from sklearn.metrics import mean_squared_error
         mse = mean_squared_error(test_labels, predictions)
         rmse = np.sqrt(mse)

In [35]: rmse

Out[35]: 0.0

In [36]: #testset = [[test_features]]
         #predictions = [[test_labels]]
         #accuracy = getAccuracy(testset, predictions)
         #print(accuracy)

In [37]: mape = 100 * (errors / test_labels)
         accuracy = 100 - np.mean(mape)
         print('Accuracy:', round(accuracy, 2),'%.')

         Accuracy: 100.0 %.
```

Fig 6.33: RMSE and Accuracy of Decision Tree

```
In [25]: # we imported the model using sci-learn kit

         # Use the forest's predict method on the test data
         predictions = rf.predict(test_features)
         # Calculate the absolute errors
         errors = abs(predictions - test_labels)
         # Print out the mean absolute error (mae)
         print('Mean Absolute Error:', round(np.mean(errors), 2), '')
         print(errors)
         #print(n12)

         Mean Absolute Error: 0.0
         [0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
          0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
```

Fig 6.34: MAE of Decision Tree

6.5: Random Forest (RF):

Random Forest is the Machine learning algorithm that can be operate by constructing multiple decision trees. The final decision is made based on majority of the trees as shown in FIG 6.34. A decision tree is a tree-shaped diagram used to determine a course of action. Each branch of the tree represents a possible decision, occurrence, or reaction.[12]

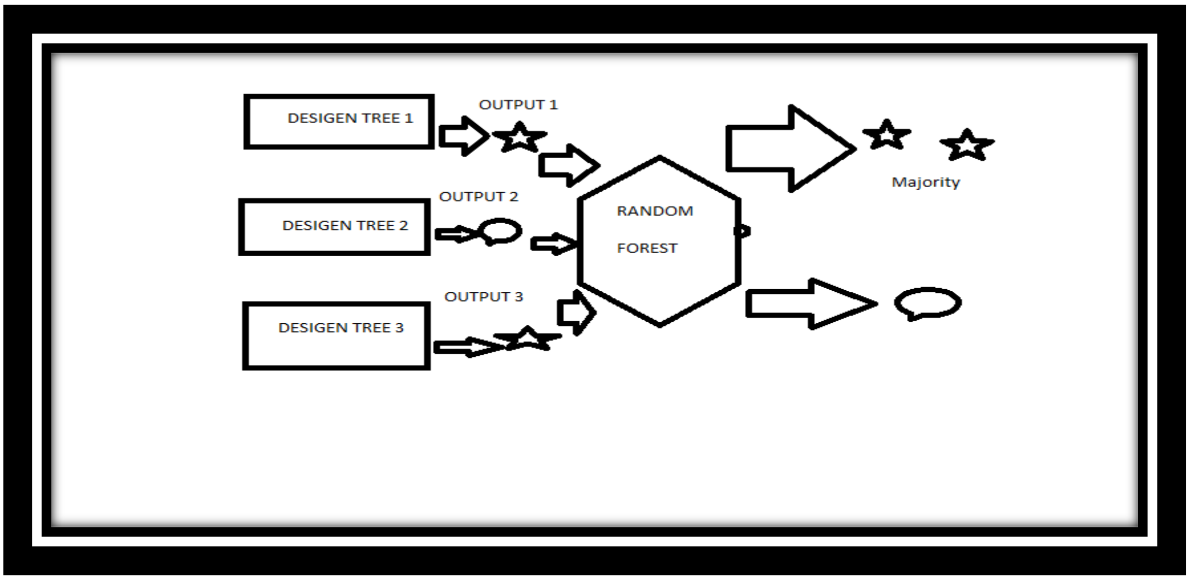


Fig 6.35: Random Forest Algorithm

```
In [23]: # The baseline predictions are the historical averages
         baseline_preds = test_features[:, feature_list.index('age')]
         # Baseline errors, and display average baseline error
         baseline_errors = abs(baseline_preds - test_labels)
         print('Average baseline error: ', round(np.mean(baseline_errors), 2))

         Average baseline error:  57.74
```

Fig 6.36: Average baseline error of Random forest

```
In [34]: from sklearn.metrics import mean_squared_error
         mse = mean_squared_error(test_labels, predictions)
         rmse = np.sqrt(mse)

In [35]: rmse

Out[35]: 0.04017416072781169

In [36]: #testset = [[test_features]]
         #predictions = [[test_labels]]
         #accuracy = getAccuracy(testset, predictions)
         #print(accuracy)

In [37]: mape = 100 * (errors / test_labels)
         accuracy = 100 - np.mean(mape)
         print('Accuracy:', round(accuracy, 2),'%.')

         Accuracy: 99.81 %.
```

Fig 6.37: RMSE and Accuracy of Random forest

```
In [25]: # we imported the model using sci-learn kit

         # Use the forest's predict method on the test data
         predictions = rf.predict(test_features)
         # Calculate the absolute errors
         errors = abs(predictions - test_labels)
         # Print out the mean absolute error (mae)
         print('Mean Absolute Error:', round(np.mean(errors), 2), '')
         print(errors)
         #print(n12)

         Mean Absolute Error: 0.01
```

Fig 6.38: MAE of Random forest

Chapter 7

Results and Conclusion

Results:

SPo2 & Heart rate samples of 41 different age patients have been taken for machine learning algorithm. The data has been trained under machine learning algorithm of decision tree(Jupyter Notebook) and out 41 samples , six people's health condition has been analyzed as reference as per below mentioned Table7.1 Total six Person's samples were taken with mean values of Spo2 and HR (heart rate).

The result shows the different health condition predictions after the machine learning algorithm.

Gender	Age	Spo2	HR	Health
1(MALE)	26	96	77	Normal(9)
1(MALE)	37	89	78	COPD(4)
2(FEMALE)	64	90	52	Critical condition(5)
2(FEMALE)	35	97	82	Normal(9)
1(MALE)	68	96	105	Tachycardia(7)
2(FEMALE)	52	97	54	Bradycardia(6)

TABLE 7.1: Results of Decision Tree Algorithm on Jupyter Notebook

Conclusion:

As Parameter shown in Comparison Table 6.1 Decision Tree Algorithm can give Better Output Then KNN (K nearest Neighbor) And Random Forest Algorithms. This device is useful in Clinical & non-clinical environment where doctor & patient can predict the health condition using the above device and methodology. This project can be enhanced in future for adding more medical Parameters.

REFERENCES

Papers:

[3] Dhiraj sunehra and Pini Ramakrishna, “web based patient health monitoring system using raspberry pi” , 2nd international conference on contemporary computing and Informatics,2016

[4] Michael Tamayo, Andrew Westover and Ying Sun, “Micro controller Based Pulse Oximeter for Undergraduate Capstone Design, “Proceedings of the 2010 IEEE 36th Annual Northeast Bioengineering Conference (NEBEC), 2010

[5] Yuan wen hau and Mas Azalya Yusofm, “Mini Home-Based Vital Sign Monitor with Android Mobile Application (my Vital Gear)”, IEEE-EMBS Conference on Biomedical Engineering and Sciences,2018

[6] Garima P Gupta, Rajashree R Nair, R Jeyanthi, “An ANN based SpO2 Measurement for Clinical Management Systems”, Energy Procedia, 2017

[7] Swaathi Venkat,Mohamed Tanveejul Arsath PS,Preejith SP And Annamol Alex, “Machine Learning based SpO2 Computation Using Reflectance Pulse Oximetry” , 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society , 2019

Web Sites:

[1] https://en.wikipedia.org/wiki/Pulse_oximetry

[2] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5247680/

[8] https://www.amazon.in/Silicon-TechnoLabs-Alphanumeric-Display-JHD162A/dp/B00XT53RI0

[9] https://www.electronicwings.com/nodemcu/introduction-to-nodemcu

[10] https://datasheets.maximintegrated.com/en/ds/MAX30100.pdf

[11] https://robu.in/product/iici2c-serial-interface-adapter-module/?gclid=Cj0KCQjwoaz3BRDnARIsAF1RfLfsG-hIe3ohHWYhdc2RGgp3B0dLLjb1PVY6tpAeipRlSg_mp7Jz8AkaAhLxEALw_wcB

[12] https://www.simplilearn.com/tutorials/machine-learning-tutorial/random-forest-algorithm

A-Abbreviations

Acronym	Abbreviation
IOT	Internet of things
ML	Machine Learning
LR	Linear Regression
DT	Decision TREE
RF	Random Forest
ANN	Artificial Neural Network
SPO2	Saturation Percentage of oxygen
R	Ratio
IDE	Integrated Development Environment
LCD	Liquid Crystal Display
SD	Standard deviation
RMS	Root Mean Square
MSE	Mean square error
CV	Coefficient of variation
MAD	Median absolute deviation
HR	Heart rate
EDA	Exploratory data analysis

B. Review Card

Gujarat Technological University Master of Engineering (Dissertation Review Card)	Date: 12/12/19 Hall No: ____
Applicable from 2019 onwards for New Syllabus - effective from Academic year 2018-19	

Semester:	3	Subject Name & Code:	Dissertation Phase 1 - 3730003	DP-1

Student's Enrollment:	1 8 1 3 7 0 7 5 0 0 0 5		
Name of Student :	Revya Dharmendrasinh Jesingbhai		
Student's Mail ID:-	revyadharmendra@gmail.com	Branch Code	50
Students Contact No.	9714800309	Branch Name	MCNT
Theme of Title :	IOT		
Title of Thesis :	Design and Implementation of pulse oximeter to monitor and predict the SPO2 level.		

OBSERVATIONS

1. Appropriateness of title with proposal. (Yes/ No) Yes
2. Whether the selected theme is appropriate according to the title ? (Yes / No) Yes
3. Justify rational of proposed research. (Yes/ No) Yes
4. Clarity of objectives. (Yes/ No) Yes

Sr. No.	Comments given by External Examiners (DP-I) (Please write specific comments)	Modification done based on Comments
1	Due importance should be given to developing/implementing algorithms for analysing data	

Guide's Signature

Note : *Above suggestions/modifications/comments must be fulfil/accommodate by the student in IR-2*

Approved ✓ Not Approved

Please tick on any one. If approved, then put marks ≥ 50 %.

Particulars	Details of External Examiner Expert 1	Expert 2
Name:	Dr. M. V. Shah	Dr. D. D. Vyas
Institute code & Name:	028, LDCE	DIET (DSU)
Mobile No.: (optional)		
Email Id:	mihirec@gmail.com	divyang.vyas@dharmsinh...
Signature	[signature]	[signature]

Gujarat Technological University Master of Engineering (Dissertation Review Card)				Date: 03/ 05/ 20
Applicable from 2019 onwards for New Syllabus - effective from Academic year 2018-19.				
Semester:	4	Subject Name & Code:	Internal Review -2 – 3740001	IR-2

Student's Enrollment:	1	8	1	3	7	0	7	6	0	0	0	6

Name of Student :	Revar Dharmendrasinh		
Student's Mail ID:-		Branch Code	60
Students Contact No.		Branch Name	MCNT
Theme of Title :			
Title of Thesis :	Design and Implementation of Pulse oximeter to monitor and predict the SPO_2 level.		

Sr. No.	**Comments given by External Examiners (DP-I)** (Please write specific comments) i) The appropriateness of the major highlights of work done; State here itself if work can be approved with some additional changes. ii) Main reasons for approving the work. iii) Main reasons if work is not approved.	Modification done based on Comments
→	DP-1 comments	Student has
1)	Developing & Implementing algorithm for analysing data. →	developed ML algorithm for analyzed the d[illegible]
	IR-2 comments	
→	Justify & compare between atleast three ML algorithm →	Yes, he has Justified between three ML
→	Data validation require for various health condition →	Yes, student has validate data
→	Data required to be communicate on cloud platform →	student has implement communication between cloud & device
		Guide's Signature

Note : *Above suggestions/modifications/comments must be fulfil/accommodate by the student in DP-2*	Approved ✓ Not Approved	*Please tick on any one.* *If approved, then put marks ≥ 50 %.*

Particulars	Details of Internal Examiner Internal Review Panel	Internal Review Panel
Name:	Rutika P Ghariya	Dr. G.D. Makwana
Institute code & Name:	137, GTU-GSET	GTU - GSET, 137
Mobile No.:	9662556736	906[illegible]
Email Id:	ap_rutika@gtu.edu.in	aso_gordhan_makwana@gtu.e[illegible]
Signature:		

C-Paper Publication certificates


International Journal of Scientific Research and Engineering Development

ISSN : 2581 - 7175

Certificate of Publication

This is to certify that paper entitled

" Fog Node Possible Architectures and Propose the Changes in Mechanism to Architecture to Make Contiki OS Real Time "

Authored by

" Dharmendrasinh J. Revar "

has been Published in Volume 3 Issue 3 - 2020

International Journal of Scientific Research and Engineering Development

Paper ID : IJSRED-V3I3P120

IJSRED IMPACT FACTOR 2.485

Editor in Chief, IJSRED

COMPLIANCE ENGINEERING JOURNAL

An UGC-CARE Approved Group - II Journal

An ISO : 7021 - 2008 Certified Journal

ISSN NO: 0898-3577 / web: www.ijceng.com / e-mail: submitcejournal@gmail.com

Certificate of Publication

Paper ID: CEJ/3038

This is to certify that the paper entitled

Future of Fog Computing In Internet of Things

Authored by:

Dharmendrasinh J. Revar

From

Graduate school of engineering and Technology Gandhinagar-India.

Has been published in

COMPLIANCE ENGINEERING JOURNAL, VOLUME 11, ISSUE 6, JUNE - 2020.

Yasu Wang, Editor-In-Chief

CE JOURNAL

International Organization for Standardization

7021-2008

D-Plagiarism Certificate

URKUND

Document Information

Analyzed document	Final dp2-Dharmndra.pdf (D75435269)
Submitted	6/22/2020 9:23:00 AM
Submitted by	mph
Submitter email	mphcall@gtu.edu.in
Similarity	4%
Analysis address	mphcall.gtuni@analysis.urkund.com

Sources included in the report

W	URL: https://www.semanticscholar.org/paper/DESIGN-AND-IMPLEMENTATION-OF-PULSE-OXIMETER- ... Fetched: 6/22/2020 9:25:00 AM	1
W	URL: https://en.wikipedia.org/wiki/Pulse_oximetry Fetched: 6/22/2020 9:25:00 AM	1
W	URL: https://www.researchgate.net/publication/336886800_A_Low_Cost_Wearable_Medical_Dev ... Fetched: 5/31/2020 5:10:53 AM	2
W	URL: https://datasheets.maximintegrated.com/en/ds/MAX30100.pdf Fetched: 6/22/2020 9:25:00 AM	1
W	URL: https://robu.in/product/iic-i2c-serial-interface-adapter-module/?gclid=Cj0KCQjwoaz3 ... Fetched: 6/22/2020 9:25:00 AM	3
W	URL: https://www.irjet.net/archives/V6/i3/IRJET-V6I3876.pdf Fetched: 11/4/2019 11:30:59 AM	1
W	URL: https://www.simplilearn.com/tutorials/machine-learning-tutorial/random-forest-algo ... Fetched: 6/22/2020 9:25:00 AM	2