ELEC0130 – Internet of things

Design and implementation of an IoT solution for detecting stress levels and underlying conditions of patients in care homes.

Student number: 19162034

MSc. in Integrated Machine Learning Systems

25/4/2021

Contents

1. EXECUTIVE SUMMARY

The scope of this project will take an existing dataset and apply it to a certain scenario. In the dataset selected there are 15 instances of subjects that a device was tested on and the values of parameters that were measured by the sensors. The sensors are an electrocardiography, electromyography, electrodermal activity, temperature, respiration and an accelerometer. They are explained with detail in section 2.1. The purpose of the device would be to detect the state that a person is in (stressed, neutral, amused). However, this is a device which could be uncomfortable to wear daily, so an application that was considered for is the monitoring of the elderly. Many care homes are understaffed, and while they are able to monitor the health and state of the patients, having a device that could transmit some of their vitals and predict the state they are in while also keeping an eye on their health could prove immensely valuable. Therefore, the scenario proposed is a care home with patients with limited mobility that are not in critical condition, so constant monitoring with expensive apparatus such as a vital signs monitor is unnecessary. This device however could help estimate their state and observe their vitals.

The dataset is separated in 15 pickle files with a variables number of instances for every case. The sensors are sampled at 700Hz, so there will be 700 instances per second. There is a ground truth as to the state the person is based on questionnaires that they completed when performing the study. This would have to be replicated with the patients in the care home to be able to train the model, and repeat it every certain amount of time to keep the model updated. Every person is considered a node, in which some preprocessing of the signals will be done to avoid sending a massive amount of data to the gateway, which will have in it a trained model to predict the state of people. The data will periodically be sent to the cloud for additional processing and to train the model. Figure 1 shows the simplified overall system architecture.

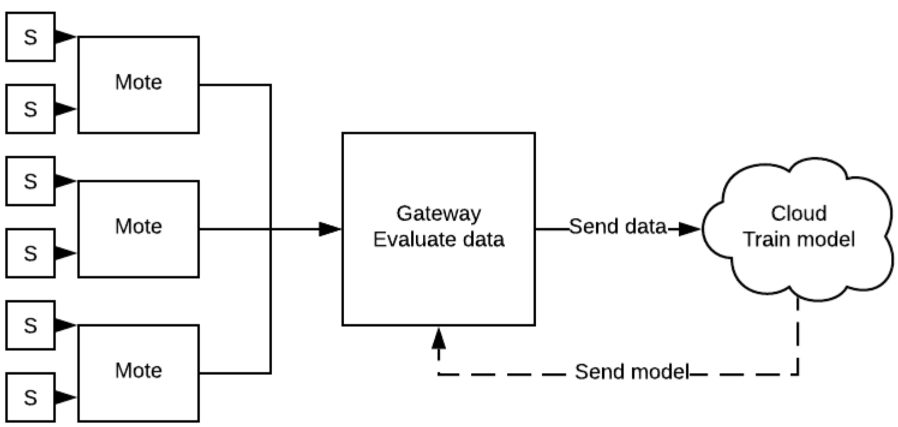


Figure 1: Architecture

Link to dataset

2. DESIGN AND IMPLEMENTATION

2.1. Sensors

All sensors are included in the same piece of equipment called the RespiBAN [1]. This is a wearable that is placed on the chest of the subjects. However, for the sake of designing the edge system, each sensor will be studied and selected individually. RespiBAN costs 2000€ without tax and weighs about 500g. These values will be used to specify the type of sensor that will be chosen, as weight and price should be considered when building an IoT device. They will however be used simply to orientate, not constrain. There are three variables that will be considered when choosing sensors: price, weight and quality of data. The first two will be quantitative, and in many cases, one will have to be prioritized over the other. The quality of data will be qualitative, either the data will be satisfactory or not. There are cheaper and lighter versions for all sensors used, but their data quality is suboptimal. Quality of data comprises accuracy, resolution, response time, range… etc. The way these sensors are connected to the board will be evaluated in section 2.2.4.

2.1.1. Electrocardiography sensor (ECG).

This sensor collects electrical signals generated by the heart, which if plotted represent your heart rate. The chosen one has a range of -1.5 to 1.5mV, manufactured by Shimmer [2], with a price point of 448€. Its weight is 31g. There are other sensors, mainly one manufactured by Plux [3], that come at a lower price but are much heavier. Weight is given more importance than price because, as will be seen in Table 1, the price ends up being lower than 2000€, but the weight is quite high.

2.1.2. Electromyography sensor (EMG).

This sensor measures the electrical activity of muscles. It can be used for identifying neuromuscular diseases, muscle pains, evaluate kinesiology (study of the human movement) … etc. The sensor must have a range of -1.5 to 1.5mV, same as the ECG. The chosen sensor is manufactured by Plux [4], at a price of 95€ and a weight of 120g. There are more sensors on the market at lower prices and weights, but the quality of data should be guaranteed. The chosen sensor can obtain precise data at the most extreme conditions with low noise.

2.1.3. Electrodermal activity sensor (EDA).

This sensor measures the variation of electrical conductance of the skin in response to sweat secretion. This is done by applying a very low voltage to the skin and measuring how the conductance varies. The sensor is manufactured by Plux [5] as well, at a price of 95€ and 150g. As before, there are other sensors available at lower prices, but the low-noise circuit design proves optimal for this application.

2.1.4. Temperature sensor (NTC).

Used to measure the temperature of the body, and as such, the range does not need to be very big. The chosen sensor is manufactured by Amphenol Advanced Sensors [6], with a range of 0 to 50ºC. The price is very low, at about 5€, with a weight of less than 4g. The price might seem too low, but there is a reasoning behind that. The other sensors chosen are fully integrated and ready for use, which makes them have a higher price point, while this one will need some assembly to be able to function in the field, so the labor to fully prepare it is not included in the price.

2.1.5. Accelerometer (ACC).

This sensor is used to measure the acceleration in each of the three axes. Here a problem is faced, as there are two options to select from:

i. Unassembled ACC [7]:
   Manufactured by Freescale, low price (10€) and low weight (2.5g). This might seem like the optimal option, but the price is solely for the accelerometer. It must be fitted to a person and it must be comfortable for them to wear. Further designing should go into how to incorporate the sensor to the system and this would need more time, money and effort.
ii. Assembled ACC [8]:

Manufactured by Plux, this sensor has a much higher price point (95€) and weight (200g). However, this is a fully assembled and ready to use sensor that could fit into the system without any further preprocessing.

Between these two options, and in order to make the project simpler, the assembled ACC manufactured by Plux would be chosen. However, when considering the scalability of this IoT system, it might be more beneficial long term to pick the unassembled one and assemble the accelerometer from scratch.

2.1.6. Respiration sensor (PZT).

This sensor is a wearable chest belt that measures displacement variations caused by the volume changes of the chest. The chosen sensor is manufactured by ThoughtTech [9], at a price of 230€ and a weight of 30g. Other options exist at a lower price but higher weight. In the next section it will be clear why better weight was preferred over better price.

2.2. Connectivity

2.2.1. Preprocessing in the edge

For every sample of the data, the total size of it should be calculated in order to decide how much memory the board would need and how often to transmit to the cloud. The type of every variable can be found in table 1.

Sensor	ECG	EMG	EDA	NTC	ACC	PZT	Total
Bits (every cycle)	64	64	64	32	64x3	64	480

Table 1: Bitrate for every cycle

The signals are sampled at 700Hz, so the data per second would be around 330 kbps. This is a very large amount of data, and for all this to be sent to the cloud the network would need to provide a large throughput, which would impact the power consumption negatively. Therefore, some processing of the data will be carried out at the nodes. In order to know how often the data should be communicated, it is imperative to know how to combine the readings for each sensor. An external library (BioSPPy) was used for this purpose, which processes the signals to extract significant parameters. For each sensor the variable preprocessing is explained:

- ECG: the heart rate can be calculated for each second. This way, every second a 64-bit number is produced.
- EMG: the signal will be processed to detect the number of offsets in a second. The data type will be int8, so 8 bits for every second.
- EDA: the main purpose of this signal is to compute the skin conductivity responses. The most important feature is the location and amplitude of the peaks. Since the signal is sampled at 700Hz, the peak location will be approximated to the closest second. Therefore, the information to pass is: if there is a peak and the amplitude. The first is a 1-bit Boolean and the second is 64 bits.
- Resp: Similar to the EDA, there will be a reading and a Boolean to indicate if there was a reading. Again, 65 bits in total.
- Temp: The mean and standard deviation for each second is the data that will be computed. These are two 64-bit numbers.
- ACC: For each of the axis, the mean and standard deviation is the value that will be stored and transmitted. This means two 64-bit numbers for every axis.

After the processing of the signals, Table 2 shows the size of the required packet for every second.

Sensor	ECG	EMG	EDA	NTC	ACC	PZT	Total
Bits (every s)	64	8	65	128	384	65	714

Table 2: Bitrate for every second

Every second 89.25 bytes are produced, and the minimum time period when data can be sent is every second, since the heart rate needs data for one second in order to be extracted. This is a much more reasonable number for the necessary throughput, since the power consumption would have been too high. With this, the decision on what type of MCU (Micro Controller Unit) should be used must be taken. Three of them are considered: ESP8266, Arduino and Raspberry Pi. Table 3 sees the differences between them. [10][11][12]

MCU	**Power consumption**	**Connectivity**	**Clock speed**	**RAM**	**Price**
ESP8266	Very Low	Wifi	80-160MHz	32KB	10$
Arduino	Low	None	16MHz	32KB	20$
Raspberry Pi Pico	Medium	Wifi and Bluetooth	133MHz	264KB	4$
Raspberry Pi 4	High	Wifi and Bluetooth	1.5GHz	2GB	35$

Table 3: MCU parameters

Comparing these three, and the amount of processing to be done as well as the frequency at which the data is sampled, it is reasonable to choose the Raspberry Pi Pico, for its high memory and clock speed, as well as the low price point. It also lends itself to more connectivity technologies. The gateway will be a larger Raspberry Pi 4, with 1.5GHz clock speed and 2GB RAM. The reason for a larger MCU is that the data must be aggregated and sent, but also that the data evaluation is done at the edge, trying to predict the state of the subjects, which required memory and speed.

2.2.2. Topologies considered

The number of motes is set to 15, as there are 15 subjects that participated in the study. As to the number of hops for every mote, the scenario chosen is that of a care home, where all subjects are in a very close range. However, the problem that arises here is that the motes are not static, since they move with the people. Most care homes offer walks around the premises or in the neighborhood. For most of the time, the motes will all be directly connected to the manager in a star topology. However, at certain times some motes will go further away, walking around the premises, needing a second hop. Therefore, two topologies will be simulated:

1. Star topology.
2. 10 at one hop and 5 at two hops.

A regular widely commercial battery such as the L-91 AA Energizer was chosen. The charge is enough to keep the battery life very long and the price is low. With these two configurations SmarthMeshIP will be used to serve as a benchmark to compare to other connectivity technologies.

2.2.3. Connectivity technologies

Table 4 summarizes the characteristics of each technology based on the same parameters. A technology not included in the table is SigFox, since the throughput and reporting interval are unacceptable [13],[14],[15],[16],[17].

Technology	**Power consumption**	**Packet loss**	**Range (max)**	**Throughput (max)**	**Cost**	**Reporting Interval**
SmartMesh IP	Low	Can be chosen	100m	17 kbps	<70$	Low
Zigbee	Low (60mW)	Can be very high	75m	0.25 Mbit/s	<15$	Low
Bluetooth	Medium (100mW)	Very low	100 m	3 Mbit/s	<10$	7.5 ms (min)

Bluetooth Low Energy (BLE)	Very low (10mW)	Very low	50m	1 Mbit/s	<5$	7.5 ms (min)
Wi-Fi	High (800mW)	Unpredictable	250m	Very high	<10$	Low
LoraWAN	Very Low	Very low	2-3km	37.5kbps	<20$	High
LTE-M	Medium	0 for under 1km	10km	1 Mbit/s	<20$	Low

Table 4: Connectivity technologies. Red cells are unacceptable and grey are undesired.

2.2.4. Sensor connectivity

Another important aspect of the connectivity layer is how to connect the sensors to the node. Two approaches appear here: simple cables to send data using I2C or implement a connectivity technology of their own, as a WBAN (Wireless Body Area Networks). The requirements for a WBAN are a short range, very low power consumption and high reliability. Standard 802.15.6 was created with this purpose in mind, with a range of up to 10m, a throughput of 10Mbps and ultralow power consumption (1mW) [18]. Table 5 summarizes the two approaches.

Cables	**WBAN**
More uncomfortable for subjects.	Higher cost
Easier to implement	More scalable
Less latency	More flexibility

Table 5: Connectivity technologies for every node

Comparing the two, it is determined that connecting with cables and using I2C is better when applied to the scenario in mind, all the while remaining open to changing this if the patients feel too much discomfort. Another reason for preferring I2C is that the weight can be decreased, since very few cables will be required.

2.3. Cloud

The main services and applications that will be implemented are for aggregating and displaying the data. It will be seen later that some of the features that are sent to the gateway and cloud are not significant to the machine learning part of the project in which the state of patients is being predicted, but they are still significant to monitor the health of the subjects. The cloud is also where the training of the model is taking place. IBM cloud was the used platform. The pipeline followed can be seen in Figure 2.

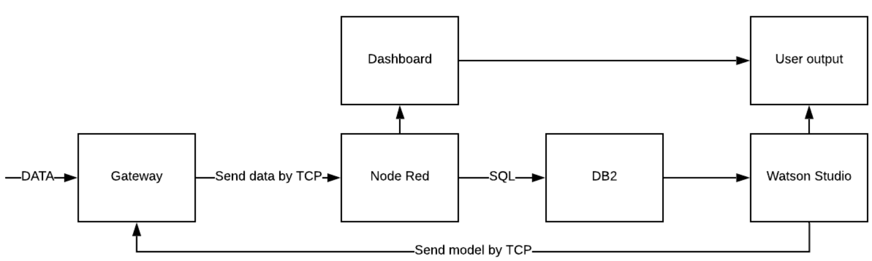


Figure 2: Cloud architecture

2.3.1. Node Red

The node red flows can be found in Appendix 2. The Node Red application will receive the data sent from the gateway and do two things with it:

- Add a timestamp to know when it was received and insert all the data into an SQL database.

- Plot the temperature and heart rate on the dashboard. These two variables are the ones that require the least processing to make sense of them. The dashboard will be displayed for the people in charge of caring for the patients. If any values are too high or too low, a message is immediately sent to them so they can rapidly react.

2.3.2. DB2

An SQL database where the data will be stored [19]. SQL was chosen instead of non-SQL due to the fact that the features are always the same when received by node red, and therefore they can be introduced into a table.

2.3.3. Watson Studio

The model is trained extracting data from the SQL database. This model is later sent to the gateway so the evaluation of the patients' state can be done locally. Apart from this, further analysis is done on the features to detect underlying health conditions, such as high blood pressure or respiratory abnormalities [20], [21]. This will not be implemented in the data analytics part, as the student does not know how to detect these problems, and an expert should be consulted. The data plotted in Watson studio is also displayed for the people that care for the patients.

2.3.4. Transfer to cloud

An important observation is the use of TCP to transfer data from and to the cloud opposed to an MQTT publish/subscribe model or an IoT cloud service.

- MQTT: Not secure enough [22], the right to privacy of the subjects impedes the usage of this model, since anyone could see the published data. Mechanisms could be put in place to make it more secure, but it would be unreasonable to go to all this trouble instead of using another protocol.
- IoT cloud service: The data is too time sensitive. Sending the data from the devices without pre-processing it would increase the latency of the system and could cause hazards to the subjects.

With this, and the fact that TCP will also have TLS running on top of it to increase the security of the transmission, it is clear that it is the best option for the transport layer of the system [23].

2.4. Data analytics

2.4.1. Data preprocessing

Already explained in section 2.2.1, reduces the size of the data to be sent to the gateway. The outliers are also found using the z-score [24] and the rows where outliers are detected are deleted.

2.4.2. EDA and model training.

A. <u>Data interpretability.</u>

16 features are created, some continuous and some discrete. it is very relevant to see if the models trained should be individual to the subject that is being treated of if they can be generalized. Therefore, two studies will be done: for one random subject and for all. The labels used are, in order: baseline, amused, stressed, relaxed.

i) One subject.

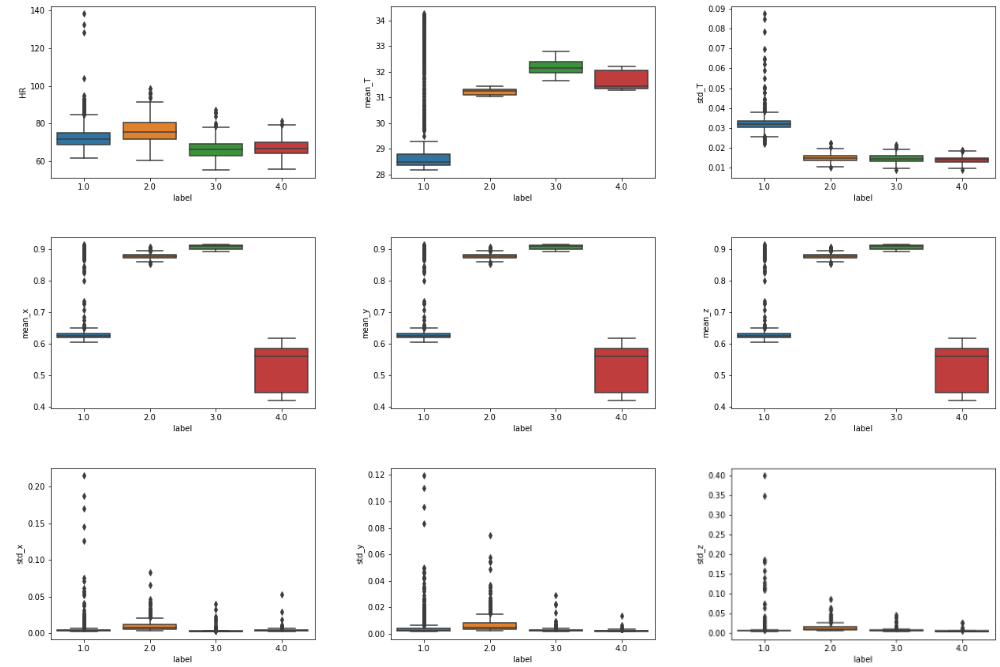


Figure 3: Boxplots for one subject

The boxplots in Figure 3 offer a lot of information as to the significance of the features. For example, the standard deviations for the accelerometer values are not very significant, since for all labels they have approximately the same values. The means, however, have very different values for each label, which leads to believe that they will have more effect on the algorithm. The same goes for both temperature features, and less so for the heart rate. Figure 4 sees the importance of the discrete features.

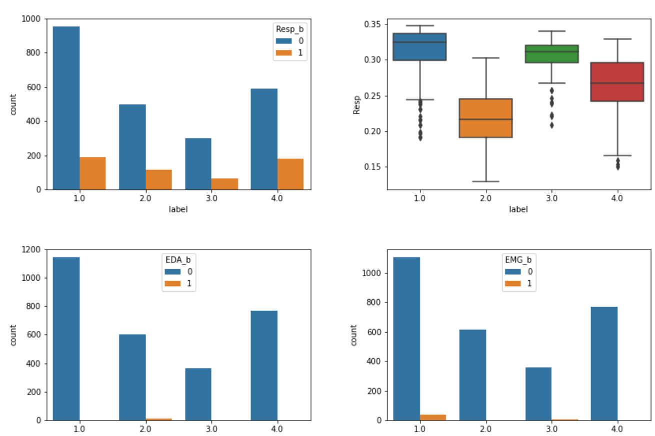


Figure 4: Bar charts for one subject

Resp has many values equal to zero, so it was considered that a binary classifier should be enough to evaluate importance, but it was not, as it can be seen in the first graph, many values exist for every label. The boxplot represents the non-zero values, and it can be seen the ranges are quite different, so this feature could have an importance when it is different than zero. As for EDA and EMG, a simple binary classifier is enough.

ii) All subjects

The same boxplots but for all subjects can be seen in Figure 5.

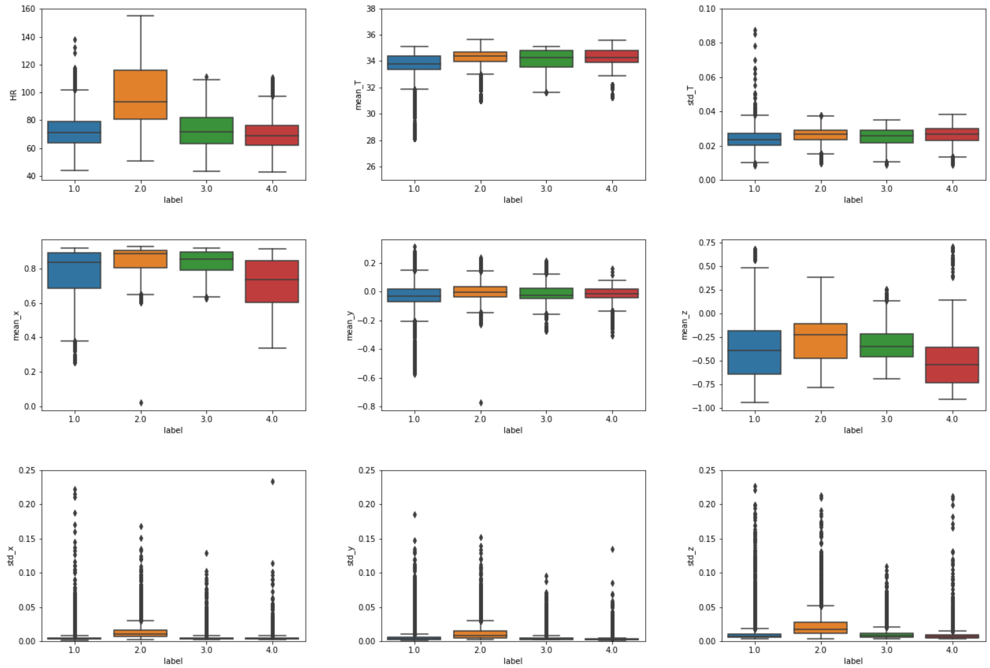


Figure 5: Boxplots for all subjects

Clearly, the variables are not distinguished enough between them. In some cases, they could indicate the class (mean of accelerometer), but in most it will be difficult to predict the label. The discrete features can be seen in Figure 6.

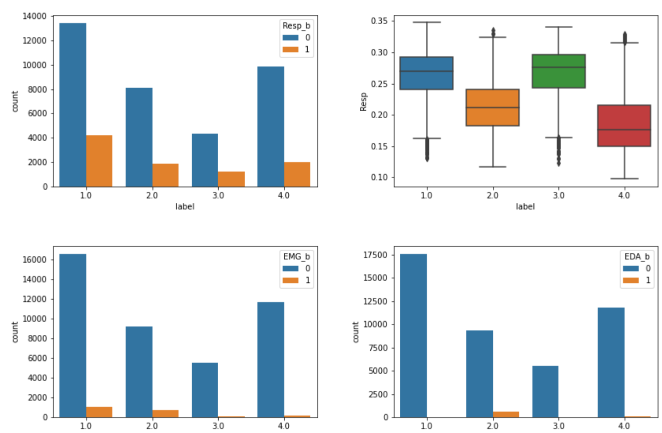


Figure 6: Bar charts for all subjects

As for the discrete features, again there is less information as to how relevant they might be, except for the EDA. This leads to believe that the results will be worse if training a single model for all people instead of training individual ones. This will however be tested in the results section.

As part of an ablation study, it is important to see which variables can be correlated and therefore could be eliminated to decrease the number of features. For this, the spearman method is used [25]. It gives a p-value (from -1 to 1, how "strong" the correlation is) and a coefficient. The initial hypothesis is that there is a correlation, and with a confidence interval of 95%, if the number is higher than 0.05, this hypothesis is rejected and there is no correlation. As for the p-value, 0-0.3 is weak, 0.3-0.5 is medium and 0.5-1 is strong. Table 6 shows the p-values found using the spearman method for all combinations of the continuous features.

	HR	Mean T	Std T	Mean x	Mean y	Mean z	Std x	Std y	Std z
HR	**X**	0.25	0.25	0.23	0.26	0.2	0.45	0.38	0.53
Mean T	0.25	**X**	0.87	0.12	0.3	0.04	0.03	0.01	0.1
Std T	0.25	0.87	**X**	0.15	0.3	0.03	0.03	-0.05	0.09
Mean x	0.23	0.12	0.15	**X**	0.17	0.7	0.13	0.22	0.3
Mean y	0.26	0.3	0.3	0.17	**X**	0.2	0.08	0.07	0.13
Mean z	0.2	0.04	0.03	0.7	0.2	**X**	0.11	0.21	0.28
Std x	0.45	0.03	0.03	0.13	0.08	0.11	**X**	0.76	0.8
Std y	0.38	0.01	-0.05	0.22	0.07	0.21	0.76	**X**	0.69
Std z	0.53	0.1	0.09	0.3	0.13	0.28	0.8	0.69	**X**

Table 6: Correlation of features. Red means no correlation, light green medium correlation and dark green strong

Two conclusions can be extracted:
1. The mean and standard deviation of temperature are very correlated, and therefore only one of them is needed to give good results.
2. The standard deviation for all axes of the accelerometer is very correlated, and therefore only one of them will be used. The heart rate also shows a strong correlation with the standard deviation in the z axis, so this is the ones that will be eliminated.

B. <u>Model interpretability</u>

A variety of models were considered, each one for its own reason. Hyper parameter selection can be found in Appendix 3.

- Logistic regression [26]: one of the simplest algorithms, plots the points and creates a regression curve. If the dataset is linearly separable, the results should be very good.
- K-nearest neighbors [27]: the points are plotted and the class of them is based on the class of the k-closest neighbors. It is a simple algorithm with only one hyper-parameter to train. The problem might be the number of features, since this algorithm struggles if there are many features.
- Decision tree [28]: computationally cheap, can offer good results is data is homogenous.
- Random forest [29]: an ensemble method that creates many decision trees and compares their results. More flexible than the previous ones, but also more computationally expensive.
- AdaBoost [30]: Combines many lower accuracy models to create one with good results. It is easy to implement and has no parameters to adjust. However, it does not deal well with outliers.

2.4.3. Gateway evaluation

The model trained in the previous section will be sent periodically to the gateway. It will use this model to make fast predictions using the data it receives. This is done at the edge instead of the cloud to avoid latency issues, since this is a time sensitive scenario.

3. RESULTS

3.1. Sensors

In Table 7 the price and weight of each sensor can be found.

Sensor	ECG	EMG	EDA	NTC	ACC	PZT	Total
Price (€)	448	95	95	5	95	230	**968**

Weight (g)	31	120	150	4	200	30	**535**

Table 7: Price and weight of sensors

A few observations can be made:

- The price is much lower than that of RespiBAN, but weight is slightly higher, although a lot of it is the weight of cables and accessories instead of the actual sensors
- Weight is more important than price when selecting some sensors due to the fact that, at the end of the day, a person will have to wear this, and it must be as comfortable and lightweight as possible.
- The weight is not a precise number, since for some sensors some extra designing and elements might be required (NTC), and for others some elements could be removed (if I2C is used, only two cables could connect all devices). In any case, 535g is an approximate measure that is quite reasonable for a person to carry, equivalent to the weight of a light hoodie.

Figure 7 shows the design of the circuit. The range is the number under the name of the sensor. The Tinkercad circuit can be found in Appendix 1.

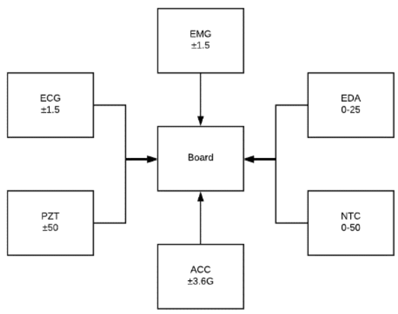


Figure 7: Sensors at node

3.2. Connectivity

The two topologies outlined in section 2.2.2. have been simulated in the SmartMesh IP estimator, obtaining the results in Figure 8.

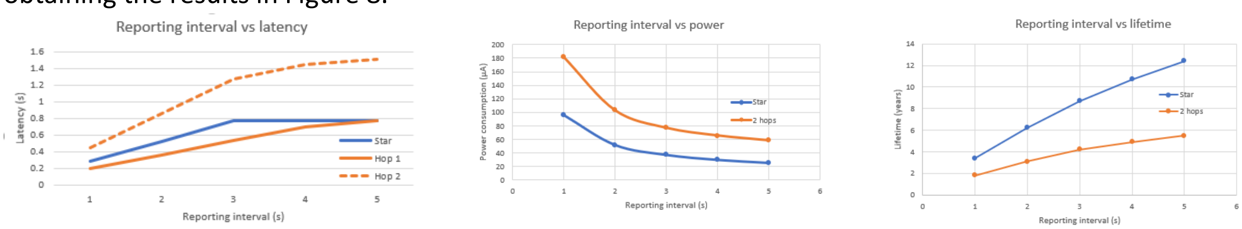


Figure 8: Connectivity topology features

Results are not shown for over 5 seconds, as it is considered that this is the maximum amount of time that the can be transmitted at. The scenario chosen is very time sensitive, as the health state of a person can change at a moments notice, and 5 seconds can make a huge difference in getting them the help the need. This being said, the power consumption decrease from reporting every one second to every two makes a big difference in the battery life, almost doubling it. Therefore, it is reasonable to choose this number as the reporting interval. If a major health risk is detected by the sensors this will be sent immediately to the manager in order to provide the care necessary. There could be a case where the health risk is not detected, and therefore sending every 2 seconds will still be enough to provide help to the person.

As to which connectivity technology to use, the ones considered are SmartMesh, Bluetooth, Bluetooth low energy (BLE) and LTE-M, since the others included in Table 5 have some unacceptable limitations. It is clear that BLE is the best option looking at power consumption, price and packet loss, but the limitation might be the range. While technically the max range is around 50m, in practice more than 10 or 20m is challenging, which greatly limits the flexibility of the system. For this reason, and although at a higher price point, SmartMesh is determined to be the most appropriate connectivity technology for the scenario in mind. If the trade-off is between price and flexibility, the latter is much more important when dealing with healthcare issues. For this reason, the topology chosen is the one with two hops, although having more power consumption and latency.

3.3. Data analytics

As explained in section 2.4, many algorithms will be tested for a dataset containing all the subjects and also for them individually, to assess if the trained model should be individual or common to all.

i) All subjects:

Two models were tested: one with all features and one as part of the ablation study without the standard deviation of temperature and the standard deviation of the y and z axis in the accelerometer. The results can be found in Table 8 for every algorithm.

ACCURACY		**Logistic regression**	**K-nearest**	**Decision tree**	**Random forest**	**AdaBoost**
All features	*Train*	0.55	1	1	0.99	0.64
	Test	0.55	0.81	0.95	0.96	0.64
Ablation	*Train*	0.54	1	1	0.99	0.64
	Test	0.54	0.81	0.95	0.97	0.63

Table 8: Accuracy score for all subjects

The first observation to be made is that the ablation study was successful. Deleting these three features does not impact model performance, so they will not be taken into account when training the model. The best performing model taking into account accuracy and processing power is the decision tree. As part of a further ablation study, the importance of the features is shown in Figure 9.

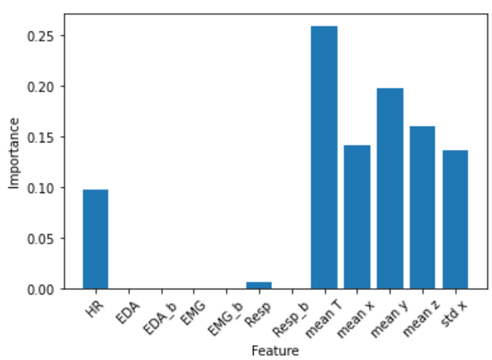


Figure 9: Feature importance

This shows that more features can be eliminated from the model. The EDA and EMG signals do not have any importance, but they are still considered valuable to monitor the health of the patients. They can be analyzed in the cloud in order to detect illnesses or other potentially harmful conditions of the patients. However, the binary classifiers can be eliminated, since they have no contribution to the model and are not helpful for monitoring the patient's health. Same goes for the binary resp feature. Therefore, the final model will only have 7 features: HR, Resp, mean temperature, the three means for accelerometer and the standard deviation for the x axis in the accelerometer. The accuracy of this model does not differ from the one in Table 7, since the only features that were deleted had no contribution.

ii) Individual subjects

Table 9 shows the test and train scores for 3 individuals who had their own model, averaged.

ACCURACY	**Logistic regression**	**K-nearest**	**Decision tree**	**Random forest**	**AdaBoost**
Train	0.95	1	1	1	0.53
Test	0.95	0.97	0.99	0.99	0.52

Table 9: Accuracy scores for one subject

No ablation study was performed on the individual subject dataset, and therefore the features that were not included in the previous section might be relevant for the individuals. In any case, the results are very good but the decision tree for all subjects provides results that don't need improving. Training a single model that will be applied to all data is much more beneficial than a 4% accuracy increase and training 15 different models. This also allows for the project to scale if more patients join. Another point where the project could scale is in trying to predict underlying illnesses of patients. If a dataset could be obtained where the subjects are affected with underlying medical conditions, this could provide labels that the algorithm could use to not only estimate the state of a person, but also their health.

4. TRADE-OFFS

Some of the trade-offs considered are detailed here:

- Price vs weight: Price is lower than for the original device, but the weight is quite high, so when possible it was prioritized to lower it.
- Speed vs elasticity: Speed is much more important, as the scenario is time-sensitive. When it comes to elasticity, the volume of data should remain constant, since the size of the data transmitted is always the same.
- Local vs cloud processing: Some of the processing is done at the edge to reduce the volume of data to be sent, but the bulk of it is done in the cloud.
- Range vs power consumption: range is much more important, as it is important for the data to reach the gateway.
- Latency vs accuracy: Latency in the cloud is not as relevant, but it is crucial at the edge, since the data should reach the gateway fast to be evaluated.
- Performance vs security: Implementing TLS on top of TCP is vital, as the data from the patients is confidential and must be protected.
- Generic vs bespoke hardware: Some hardware can be generic, like the temperature sensor, but since the measurements have to be precise and specific, bespoke hardware is generally preferred.
- Power consumption vs data rate vs operating range: Between these three, data rate is the least important, since the size of the packets to send was decreased in the preprocessing at the node. Operating range is vital, whereas power consumption is attempted to be minimized, but is given less importance.
- Functionality vs usability: Functionality is more important, as the people evaluating the results will be trained professionals, not very concerned with usability.

REFERENCES

[1] BioSignalsPlux, “Biosignal acquisition tool-kit for advanced research applications”, 01/02/2018

[2] Shimmer, “ECG User Guide”, 2018

[3] BioSignalsPlux, “Electrocardiography (ECG) Sensor Datasheet”, ECG 10082020, 2020

[4] BioSignalsPlux, “Electromyography (EMG) Sensor Data Sheet”, EMG 26012018, 2017

[5] BioSignalsPlux, “Electrodermal Activity (EDA) Sensor Datasheet” EMG 03092020, 2020

[6] Amphenol Advanced Sensors, “Biomedical Chip NTC Thermistors”, 3/2020

[7] Freescale, “±1.5g, ±6g Three Axis Low-g Micromachined Accelerometer”, MMA7361L, 2008

[8] BioSignalsPlux, “Accelerometer (ACC) Sensor Data Sheet”, ACC 11082020, 2018

[9] Though Technology “360 Suite Reference Manual”, 2018

[10] Parihar, Yogendra Singh. (2019). Internet of Things and Nodemcu A review of use of Nodemcu ESP8266 in IoT products. 6. 1085.

[11] Maksimovic, Mirjana & Vujovic, Vladimir & Davidović, Nikola & Milosevic, Vladimir & Perisic, Branko. (2014). Raspberry Pi as Internet of Things hardware: Performances and Constraints.

[12] Louis, Leo. (2018). Working Principle of Arduino and Using it as a Tool for Study and Research. International Journal of Control, Automation, Communication and Systems. 1. 10.5121/ijcacs.2016.1203.

[13] M. U. Memon, L. X. Zhang and B. Shaikh, "Packet loss ratio evaluation of the impact of interference on zigbee network caused by Wi-Fi (IEEE 802.11b/g) in e-health environment," 2012 IEEE 14th International Conference on e-Health Networking, Applications and Services (Healthcom), 2012, pp. 462-465, doi: 10.1109/HealthCom.2012.6379462.

[14] S. Dawaliby, A. Bradai and Y. Pousset, "In depth performance evaluation of LTE-M for M2M communications," 2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2016, pp. 1-8, doi: 10.1109/WiMOB.2016.7763264.

[15] J. Skog Andersen and J. Eriksson, ‘Investigating the practical performance of the LoRaWAN technology’, Dissertation, 2017.

[16] Thomas Watteyne, Lance Doherty, Jonathan Simon, and Kris Pister. 2013. Technical Overview of SmartMesh IP. In Proceedings of the 2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS '13). IEEE Computer Society, USA, 547–551

[17] Bisdikian, Chatschik. (2002). An Overview of the Bluetooth Wireless Technology. Communications Magazine, IEEE. 39. 86 - 94. 10.1109/35.968817.

[18] Addo, Ernest & Kommey, Benjamin & Agbemenu, Andrew. (2019). Wearable Networks: Requirements, Technologies, and Research Trends.

[19] Li, Yishan & Manoharan, Sathiamoorthy. (2013). A performance comparison of SQL and NoSQL databases. IEEE Pacific RIM Conference on Communications, Computers, and Signal Processing - Proceedings. 15-19. 10.1109/PACRIM.2013.6625441.

[20] Plácido da Silva, Hugo & Lourenco, Andre & Canento, Filipe & Fred, Ana & Raposo, Nuno. (2013). ECG Biometrics: Principles and Applications. BIOSIGNALS 2013 - Proceedings of the International Conference on Bio-Inspired Systems and Signal Processing. 10.5220/0004243202150220.

[21] Wu T., Wang SX. (2011) Application of EDA Technology in the Education of Automatic Control System Design. In: Lin S., Huang X. (eds) Advances in Computer Science, Environment, Ecoinformatics, and Education. CSEE 2011. Communications in Computer and Information Science, vol 218. Springer, Berlin, Heidelberg.

[22] Anthraper, Joseph & Kotak, Jaidip. (2019). Security, Privacy and Forensic Concern of MQTT Protocol. SSRN Electronic Journal. 10.2139/ssrn.3355193.

[23] Satapathy, Ashutosh & Livingston, Jenila. (2016). A Comprehensive Survey on SSL/ TLS and their Vulnerabilities. International Journal of Computer Applications. 153. 31-38. 10.5120/ijca2016912063.

[24] Cousineau, Denis & Chartier, Sylvain. (2010). Outliers detection and treatment: A review. International Journal of Psychological Research. 3. 10.21500/20112084.844.

[25] Spearman, C. (2010). The proof and measurement of association between two things. International journal of epidemiology, 39 5, 1137-50

[26] Bewick, Viv & Cheek, Liz & Ball, Jonathan. (2005). Statistics review 14: Logistic regression. Critical care (London, England). 9. 112-8. 10.1186/cc3045.

[27] Cunningham, Padraig & Delany, Sarah. (2007). k-Nearest neighbour classifiers. Mult Classif Syst.

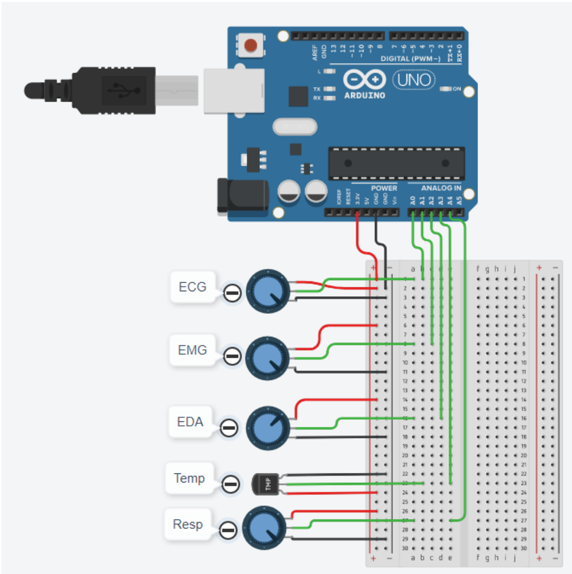
[28] Rokach, Lior & Maimon, Oded. (2005). Decision Trees. 10.1007/0-387-25465-X_9.

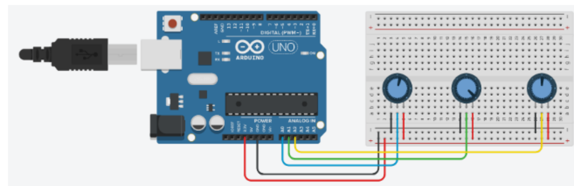
[29] Cutler, Adele & Cutler, David & Stevens, John. (2011). Random Forests. 10.1007/978-1-4419-9326-7_5.

[30] Bühlmann, Peter. (2012). Bagging, Boosting and Ensemble Methods. Handbook of Computational Statistics. 10.1007/978-3-642-21551-3_33.

APPENDICES

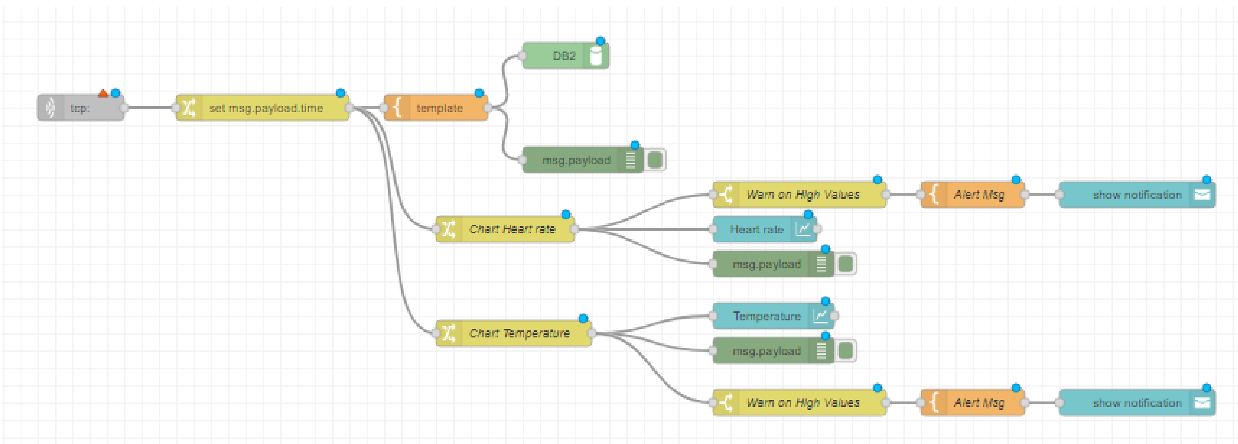
APPENDIX 1: Tinkercad Circuit





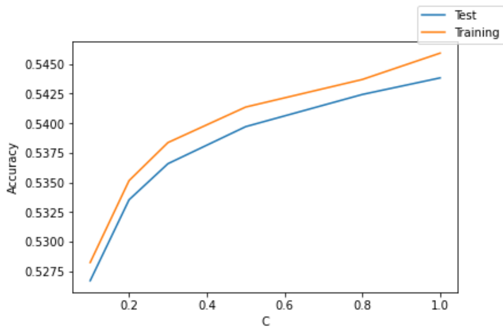
The first figure represents 5 of the sensors, while the second one is the accelerometer, with every axis represented as potentiometers.

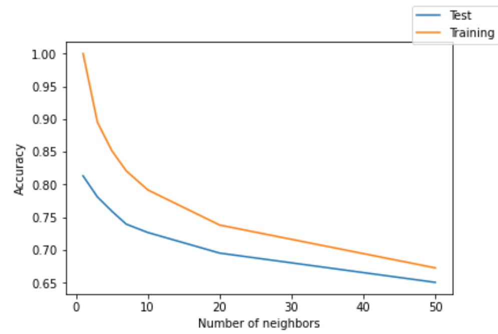
APPENDIX 2: Node red flow



The template block before the DB2 database prepares the data for inserting into it. A timestamp is added to the message and the temperature and heart rate are plotted, sending a message if any of them are outside a certain range.

APPENDIX 3: Hyper-parameter tuning





Logistic regression

K-nearest neighbors

The other algorithms did not need any hyper parameter tuning.