Diplomarbeit

RWTH AACHEN UNIVERSITY

Eduard Ramon Maldonado

Algorithms for B waves detection

LEHRSTUHL FÜR MEDIZINISCHE INFORMATIONSTECHNIK
Univ.-Prof. Dr.-Ing. Dr. med. Steffen Leonhardt
Betreuer-/in: Dipl.-Ing. Inga Elixmann
Datum: April 9, 2013

mediT

Acknowledgment

First of all, I would like to thank my parents, who always offer me their unconditional support in everything I do and in every decision I take even when they are far away.

I also would like to express my gratitude to my supervisor of the Thesis, Inga Elixmann, who offered me the opportunity of doing this project in the RWTH Aachen and who spent her time in advising and encouraging me during this last year.

Finally, thanks to my colleagues in Aachen, who made from those 6 months an experience that I will never forget.

Erklärung

Ich versichere hiermit, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten und nicht veröffentlichten Schriften entnommen sind, wurden als solche kenntlich gemacht.

Ort, Datum	Unterschrift

Abstract

The objective of this Master Thesis was to develop algorithms for B waves detection in ICP. This goal was approached by two different methods that depend basically in the resolution of the acquired ICP. Then, both methods were adapted to work in an ultra-low power microcontroller.

The first method works using ICP recorded at 1 Hz and it is based on the Lundberg's definition of B wave. A plus of this algorithm is that reduces to the minimum the number of samples per block to classify. The results obtained after testing it using long records of ICP from 27 patients were an accuracy of 89,59%, a specificity 89,71% and a sensitivity of 89,16%. These results did not change when the code was adapted to the microcontroller.

The second method requires ICP obtained with a sampling rate of 100 Hz. It is based on the morphology of the pulse waves present in the ICP and caused by the change of blood volume inside the skull with every heartbeat. A total of 1430 blocks of ICP (864 for lack of B wave and 566 for presence of B wave), everyone with duration of 41 seconds, were used to extract 21 features from each one. Then a MLP classifier and a SVM classifier were tested and compared. The best results were obtained by the SVM classifier, reaching an accuracy of 86,37%, a specificity of 88,09% and a sensitivity of 83,74% when all features were used. After adapting the algorithm to the microcontroller the results were nearly the same.

Contents

Symbols

Medicine terms

ICP	Intracranial Pressure
CBF	Cerebral Blood Flow
CSF	Cerebrospinal fluid
ABP	Arterial Blood Pressure
CPP	Cerebral Perfusion Pressure
MRI	Magnetic Resonance Imaging
CT	Computed Tomography
NPH	Normal Pressure Hydrocephalus

Physical Terms

p	Pressure	$mmHg$
t	Time	$second$
f	Frequency	Hz

Mathematical Terms

pdf	Probability Density Function
DFT	Discrete Fourier Transfrom
FFT	Fast Fourier Transfrom
ARMA	Autoregressive Moving Average Model
MAF	Moving Average Filter
$\mathcal{L}$	Lagrangian
∇	Gradient

Algorithms

MOCAIP	Morphological Clustering Analysis of Intracranial Pressure Pulses
PCA	Principal Component Analysis
ANN	Artificial Neural Network
MLP	Multilayer Perceptron
SVM	Support Vector Machines
kNN	K-Nearest Neighbor Algorithm
PLR	Perceptron Learning Rule
BP	Back Propagation Algorithm

1 Introduction

Hydrocephalus is a disease that affects mainly children and old people and causes a harmful increase of the intracranial pressure. Statistics revealed that this disorder occurs in 1 out of 500 children [1] and in 1 out of 200 adults over the age of 55 [2]. Moreover, the growing age of the population is increasing the number of positive hydrocephalus diagnoses.

For this reason, a number of studies on hydrocephalus has been executed the past years, both in the medical and technically case. In addition to investigating the origin of hydrocephalus, there has also been done research on how to improve patient's quality of life through more efficient treatments.

Despite the effort of the community of scientists in the past decades, many questions about hydrocephalus remain unanswered. It is true that the procedures for diagnosis and treatment for hydrocephalus have been standardized, but the methods to carry it out remain unchanged.

Nowadays, the most common method used is implanting a shunting system, normally ventriculoperitoneal, by surgery. The main element of this system is a valve that controls the intracranial pressure (ICP) and regulates the pressure to remain in a safe region established by the doctor. The drawback of this system is that the only parameter that controls the valve is the intracranial pressure. This can create complications from overdraining, poor drainage, or just unnecessary drainage. For this and other reasons, these systems require regular follow up and surveillance from the doctor, fact that provides poor quality of life to the patient.

By intracranial pressure monitoring it has been observed that there are some factors directly or indirectly linked with autoregulation mechanisms and with the drainage efficiency [3] [4]. These factors provide a lot of information so they must be checked before shunting. For this reason we are developing a mechatronic implant called iShunt. Our device follows the same philosophy as the others, but having the benefit of taking into account many other criterions than only the intracranial pressure. This fact increases the probability of obtaining a successful drainage and minimizes the number of possible complications.

One of the criterions used by iShunt is the presence or lack of B waves (or Lundberg waves) in the ICP recordings. Many studies claim that a high percentage of incidence of B waves predicts a more efficient drainage and therefore it is essential to detect them in order to achieve a successful treatment [5]. For this reason, in this master thesis a number of new algorithms for automatic B waves detection have been studied and created.

2 Biomedical Background

2.1 Intracranial Pressure: Hypothesis of Monro-Kellie

Alexander Monro was the person that in 1783 established the bases for understanding the compliance between the different liquors inside the skull. His studies revealed that [6] [7]:

1. The brain was inside a rigid structure that could not be expanded.

2. The brain was nearly incompressible.

3. Then, the total volume of other substances inside the skull was almost constant.

Some years on, Kellie de Leith obtained more results that helped to support and correct some points of the theory proposed by Monro. According to the Monro-Kellie doctrine there are three intracranial components, brain tissue, blood, and CSF [8]. Every component contributes to the global volume with 1400, 140 and 140 cm^3 respectively, that is to say 80%, 10% and 10%. Mathematically, Monro-Kellie doctrine can be described with the expression below:

$$V_{brain} + V_{cpp} + V_{blood} = Constant \tag{2.1}$$

If one of the three components increases, the other two must decrease in order to maintain the ICP constant. In general, a discrete increase in the intracranial volume does not mean an increase of the ICP. This happens thanks to the compensatory mechanisms. In contrast, if an intracranial volume keeps on rising, the ICP starts increasing. The illustration 2.1 shows the relation between the intracranial volume and the ICP.

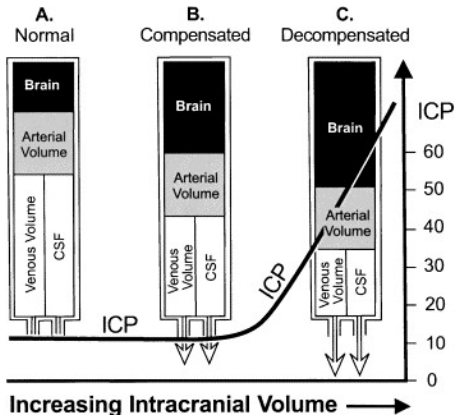


Figure 2.1: Intracranial compensation for increasing brain swelling. [9]

2.2 Hydrocephalus

Hydrocephalus is a disorder that causes an excess of CSF in the brain [1], more specifically in the ventricles. This excess of liquid causes a dilatation of the ventricles and at the same time the brain is harmfully compressed. There is not only one definition for this disorder, these two are the most extended ones [10].

1. *Hydrocephalus is a condition characterized by a dynamic imbalance between the formation (production) and absorption of spinal fluid resulting in an increase in the size of the fluid cavities (ventricles) within the brain.*

2. *Hydrocephalus is a condition characterized by a dynamic imbalance between the formation (production) and absorption of spinal fluid that results in an increase in the size of the fluid cavities within the brain and, in some situations, in an expansion of the spaces outside the brain, with or without an increase in the size of the ventricles.*

In the following illustrations, two MRI scans show the difference between ventricles when hydrocephalus and when not:

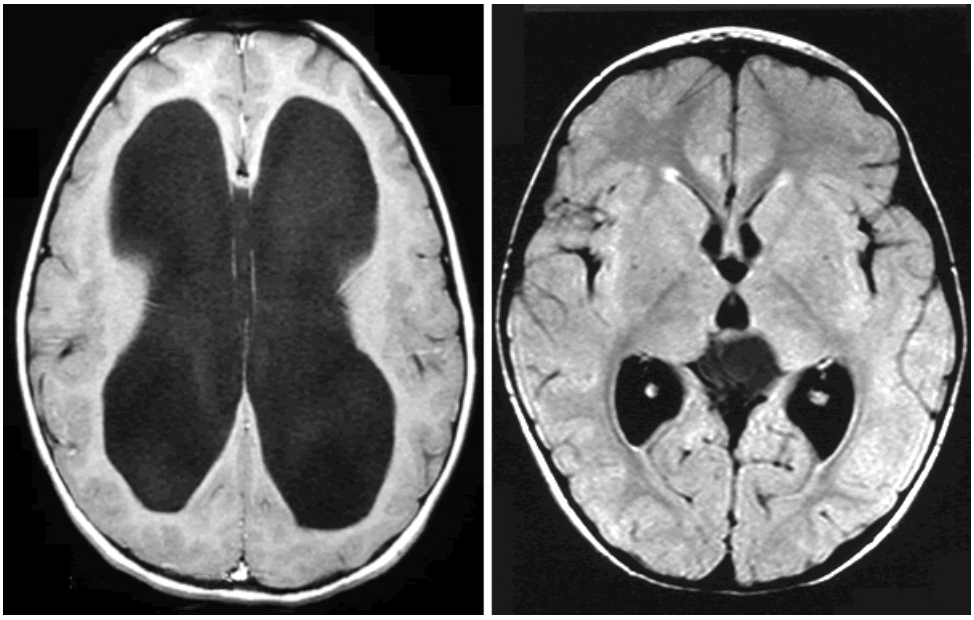


Figure 2.2: An MRI scan of patient with hydrocephalus (left) and a normal MRI scan (right) The large dark area on the left is the ventricles, made bigger by a build-up of CSF [11].

Depending on the flow of the cerebrospinal fluid between the ventricles, hydrocephalus is classified in communicating and non-communicating. If the liquid is blocked after leaving the ventricles, hydrocephalus is communicating since the liquid can still flow between ventricles. Non-communicating hydrocephalus occurs when one or more of the passages between ventricles are blocked so the liquid cannot go through.

The symptoms of hydrocephalus depend on the age, how long the disorder is present in the patient and the tolerance of the patient to the increased cerebrospinal fluid. In babies, the clearest symptom is the fast elongation of head curvature.

Figure 2.3: African kid with Hydrocephalus and elongation of the head curvature [12].

Early symptoms may also include [13]:

- Eyes that appear to gaze downward.
- Irritability
- Seizures
- Separate sutures.
- Sleepiness
- Vomiting

In older children and adults, these symptoms can include [13]:

- Brief, shrill, high-pitched cry.
- Changes in personality, memory and ability to reason or think.
- Changes in facial appearance and eye spacing.
- Crossed eyes or uncontrolled eye movements.
- Difficulty feeding.

- Excessive sleepiness.
- Headache.
- Irritability, poor temper control.
- Loss of coordination and trouble walking.
- Muscle spasticity.
- Slow growth.
- Slow or restricted movement.
- Vomiting.

In old people, the most common kind of hydrocephalus is called Normal Pressure Hydrocephalus and its symptoms are:

- Progressive mental deterioration and dementia.
- Difficulties for walking and loss of bladder control.

These last symptoms belong to other common diseases like Alzheimer or Parkinson as well. This fact often drives doctors to a wrong diagnoses and, consequently a bad treatment.

The first guidelines for diagnosis and treatment for Normal Pressure Hydrocephalus appeared in Japan in 2004 [5] due to the aging of Japanese society, which in turn made the number of patients with this disorder increase. For the diagnosis there are many available technics. They are summarized in the following lines [14]:

Brain diagnosis images: Computed tomographies (CT) and magnetic resonance imaging (MRI) are used to detect if there is enlargement of the ventricles as well as evaluate the CSF flow and provide information about the surrounding brain tissues.

Neuropsychological Test: Detects loss of brain function.

Lumbar puncture: Provides information about different aspects of the CSF like the pressure.

Infusion test: Consists in introducing liquid inside brain and observe the capacity of the body to absorb the excess of liquid and bring the intracranial pressure again into normal values

Intracranial pressure monitoring: A small pressure sensor is inserted through the skull into the brain or ventricles to measure the ICP. Although ICP is not always high, concrete kinds of patterns in the signal can reveal useful information about the patient condition.

Regarding to the treatment, the most common technic is shunting surgery. In general terms, this method consists in draining the excess of the liquid from the brain to another place of the body where CSF is not dangerous and can be absorbed. To achieve this, a catheter is used and basically one end of the catheter is placed where there is the exceed of CSF, normally the ventricles or the spinal chord, and the other end is placed where the liquid will be shunted, for example the stomach or a chamber of the heart. The flow of the CSF along the catheter is controlled by a valve that can be regulated according to the patient condition [15].

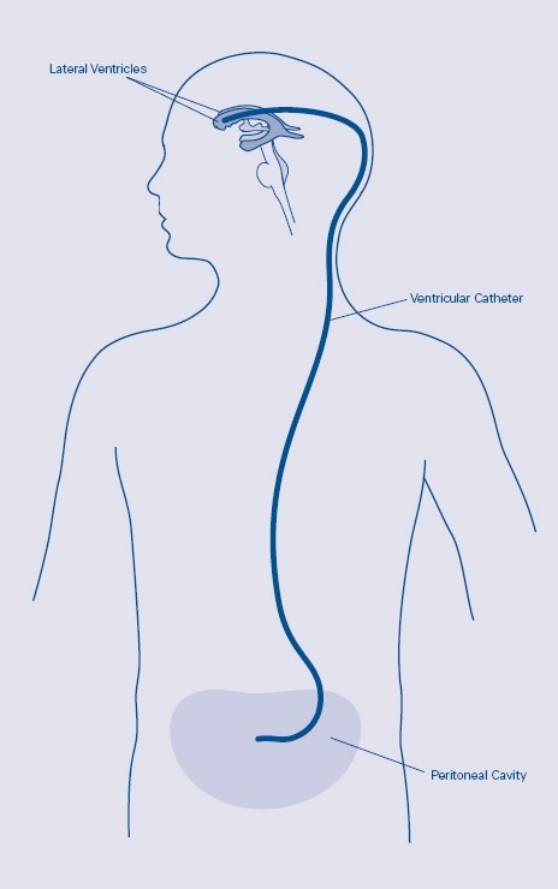


Figure 2.4: Ventriculoperitoneal shunt system [16].

The flow graph on the next page has been extracted from the Japanese guidelines and shows how different technics can be used to diagnose Normal Pressure Hydrocephalus.

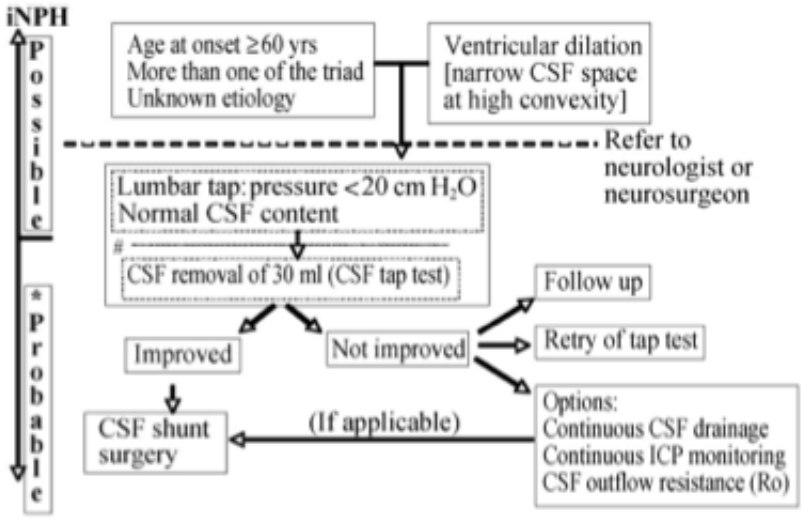


Figure 2.5: Japanese Guidelines for diagnoses and treatment of Hydrocephalus [5].

As it can be observed in the illustration above, not in all cases shunt surgery is directly carried out. A previous study is required to understand if this process is necessary as well as if it will be efficient.

3 Technical Background

3.1 iShunt

Nowadays, devices used for the treatment of Hydrocephalus consist basically in valves that keep the ICP in a safe range of values. The newest valves, which are still in research, allow the doctor to regulate the safe range depending on the requirements of the patient. Moreover, they include acceleration sensors so they can work independently of the position of the patient.

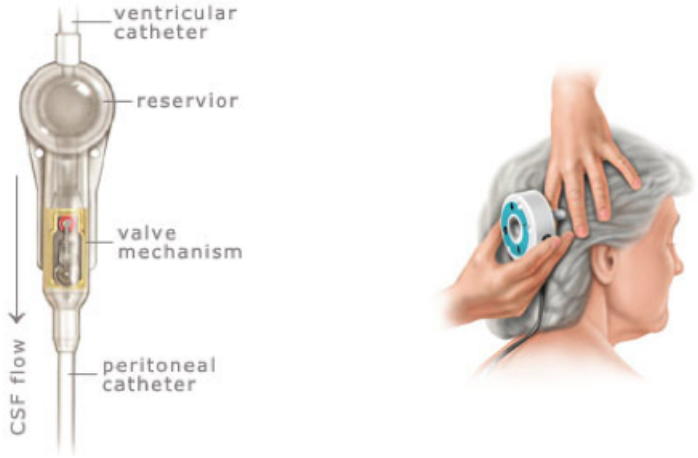


Figure 3.1: Valve of a conventional Shunting system (left). Programable Shunting system (right) [14].

Adjustable valves have meant a breakthrough in front of the valves with fixed range. Even so, problems from overdrainage or underdrainage still appear in some patients. That is the reason why scientists and technics research in new shunting methods that perform a more accurate drainage, depending on the person and on the state of the person in a particular time.

A new and more advance kind of adjustable valves is called *programmable shunt*. These valves offer the physician the possibility of change the opening pressure without the need of carry out more surgery. Moreover, these valves offer a bigger amount of different security range what allows the doctor to select a more accurate treatment.

Despite this kind of devices have been improved in the last years, the philosophy they follow is the same. They drain liquid according to the intracranial pressure. When intracranial pressure rises, the resistance of the valves gets lower, draining more fluid, and the ICP

decreases and stays again in the security range. In this process there are no other factors involved in the shunting decision.

Several studies claim that some patterns related with shunting decisions can be observed by ICP monitoring. Some behaviours of the ICP may suggest information about the compensatory mechanism state and also about shunting efficiency. Shunting systems should take into account these patterns in order to improve their efficiency and the quality of life of the patient. This is the main idea iShunt arises from. iShunt is a mechatronic implant that makes use of the information stored in the ICP to execute a more efficient drainage, adapting it to the patient and to the state of the patient in a current moment.

iShunt device is compound by three main parts: Pressure measurement system, Control unit and Shunting system. In the illustration above there is a schematic overview.

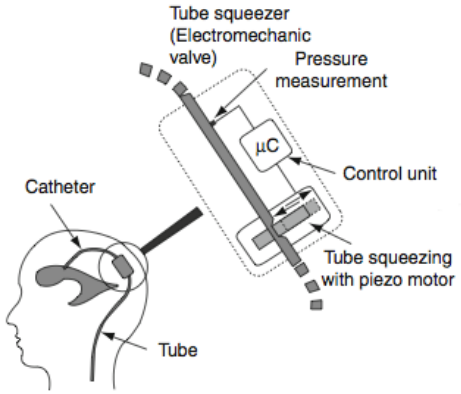


Figure 3.2: Diagram showing iShunt concept [17].

The system acquires data from intracranial pressure and stores it in a buffer. Data can be acquired by different sampling frequencies. Depending on the frequency used, some algorithms will be able to work and some will not because of the data resolution.

Control system will carry out a thorough analysis of the data, extracting these factors that influence in the shunting decision. In one hand, an analysis of the morphology of the P-waves will be done in order to obtain an idea of which is the state of the compensatory system. A low compensatory capacity with a high ICP would suggest the necessity of drain. In contrast, a high compensatory capacity with a raised ICP, maybe would suggest not to drain and wait. On the other hand, the system will detect the presence or lack of B-waves, which are related with shunt efficiency. With this and more characteristics extracted by the algorithms of the control unit, a decision will be taken and the actuator of the iShunt will be controlled accordingly.

One of the novelties of iShunt is the adaptive drainage system. The concept is completely different. Nowadays, valves are mechanic devices offers resistance (regulable resistance) to the fluid when it goes through. This resistance is inversely proportional to the pressure.

In iShunt, the drainage system is mechatronic. It is compound by a piezomotor that squeezes the tube that contain the fluid. This controls the resistance to the fluid and consequently the intracranial pressure. Using a piezomotor provides much more precision and manageability. Since it can be programed there are infinite ways to react to every kind of behaviour of the ICP. The next illustration compares the response to A-wave by using conventional valve and using an electromechanic tube squeezer valve as example.

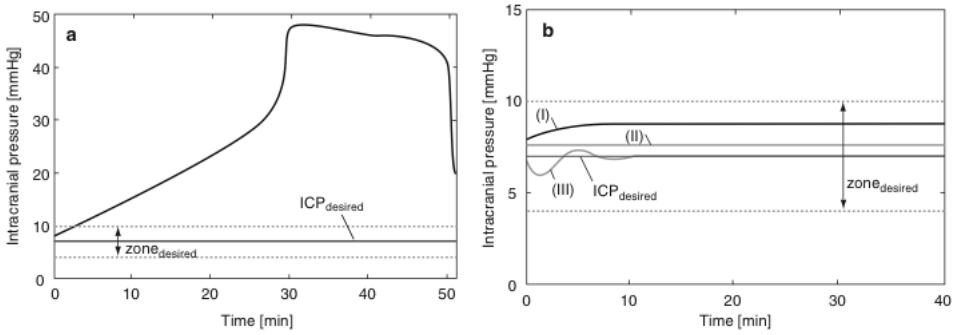


Figure 3.3: Simulation of (a) A wave, and (b) with additional ball-in-cone valve connected to a tube with inner diameters of 0.7 mm (I), 1,3 mm (II) and tube squeezer (III). ICP intracranial pressure [17].

3.2 ICP Monitoring: Slow waves and Pulse waves

ICP monitoring is a technic that allows doctors and researchers to observe and analyse the variations of the intracranial pressure. The first experiments related with this technic started around 1866 by Leyden [18]. Some years on, in 1927, Adson and Lillie introduced the ventricular puncture and catheter insertion for central ICP monitoring. Afterwards Guillaume and Janny achieved the same measurements by using a mechanic-electrical pressure transducer. But it was not until 1960 that Lundberg published his work about ICP analysis, presenting his concept of cerebrospinal compliance and establishing the bases for ICP analysis. Moreover, Lundberg contributed in the definition of 2 kinds of pressure oscillations that are called Plateau-waves and B-waves. These last are also called Lundberg waves. Furthermore, there is another type of waves called C waves that, within the other two, form the group known as Slow Waves.

The origin and regulation of slow waves is still unclear despite the years of investigation invested on this issue. They are used to appear in ICP, but they can be found in CSF and ABP as well. These oscillations contain valuable information related with the condition of the patient [19] [20] [21]

- Function of cerebral vasculature.
- Intracranial compliance.

- Autoregulation.
- Neurovegetative cardiovascular system.
- Abnormal breathing.
- Sleeping stages.

Moreover, it is crucial to determine when slow waves are pathological and when not. Many studies had been carried out in order to know more about this issue and the most relevant are summarized in table 3.1.

Authors	A Waves	B waves
Chawka et al. (1974)	occasional period: 5 to 20 min	frequency: 1 wave/min duration of trains: 5 to 30 min
Symon and Dorsch (1975)	mean daily occurrence: 1.58 (SD=1.67) mean amplitude: 18 mmHg (SD = 11.2)	occurrence>=80% of the recording
Lamas et al. (1980)	amplitude 8 mmHg to 34 mmHg period: 8 to 14 min	"very frequent"
Pickard et al (1980)		frequency: 1 wave/min minimal amplitude: 2-3 mmHg occurrence > 5% of recording
Janny et al (1981)	"low amplitude, degraded"	
Borgessen and Gjerris (1982)		frequency: 1-2 wave/min occurrence >= 50% train duration: 10 min
Godersky and Graff-Radford (1991)		occurrence > 50%
Raftopoulos et al. (1994)		amplitude > 9 mmHg

Table 3.1: Characteristics of Slow ICP Waves Considered as Pathological in Chronic Hydrocephalus [19].

It is essential to determine the incidence of B waves in long recordings basing the results in a model shared and agreed by all the scientific community. Contrary to the methodology used by Lundberg and Janny, who analysed and described slow waves visually, nowadays many methods have been designed and implemented to detect and analyse slow waves mathematically, which allow a more objective, precise, graphical and semi-automatic analysis. Although the improvements in techniques, most typical slow waves definitions used for ICP analysis are the ones provided by Lundberg and Janny many years ago. All they are described in table 3.2.

Classification	Wave	Oscillation (wave/min)	Amplitude (mmHg)	Frequency band (mHz)
Janny	type 1	6 to 12	1.5 to 2.2 (healthy subjects)	66.3 to 200
	type 2 or slow cycle "coup d'hypertension"	0.5 to 3 [dominant: 1 w/min] variable occurrence	10.3 (current)	8.33 to 50
Lundberg	C wave	4 to 8	from discernible to 20	66.33 to 133.3
	B wave	0.5 to 2 [dominant: 1 w/min]	from discernible to 50	8.33 to 33.3
	A wave	variable occurrence (current duration 5 to 20 min)	50 to 100 (current)	
Frequency	UB			50 to 200
	B			8 to 50
	IB			<8

Table 3.2: Classifications of Slow ICP Waves [19].

In table 3.2, the last classification arises from a newer frequency approach that was proposed in 1994 [22]. This model is strictly defined in the frequency domain where B waves

are taken as a reference. The other two types of oscillations (IB and UB) are defined by its relative position towards B waves:

- Ultra B (UB) waves are oscillations with a frequency higher between 50 mHz until 200 mHz.
- B waves are defined in a similar way with Janny's model. This group include all oscillations with frequencies between 8 mHz and 50 mHz.
- Infra B (IB) waves include oscillations with frequencies lower than 8 mHz.

When an Infra B wave or A wave occurs, the ICP increases suddenly to 50-100 mmHg (approximately) and remains around this value from 5 to 20 minutes. After this time, the ICP falls abruptly until levels similar to those at the beginning or even lower. These oscillations are pathological in all cases. Normally, patients in who A waves are present suffer neurological deterioration [23].

On the other hand, B waves are not always pathological [24].For example, a study revealed presence of B in healthy infants during sleep[22]. In general, they may be due to respiratory changes, more specifically they have been associated with Cheyne-Stokes respiration [23] and due to variations in CBF [25].

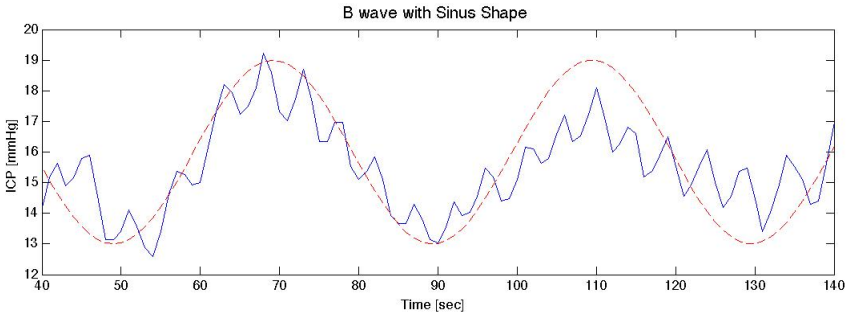


Figure 3.4: B wave with sinusoidal shape.

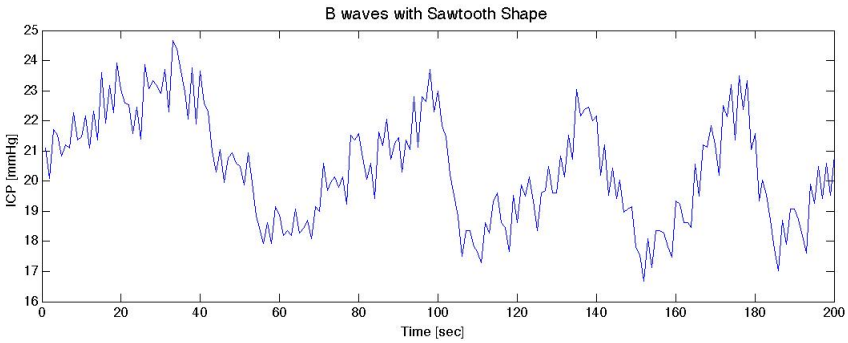


Figure 3.5: B wave with sawtooth shape.

In patients with Hydrocephalus, it has been observed in several studies that a high incidence of B waves in long records predicts a positive outcome after shunting in NPH patients with high ICP levels. iShunt will make use of it in order drain not only when ICP is raised but also when ICP the occurrence of B-waves predict a successful outcome [3] [4] [21].

B waves detection is an essential part of this project. This mechanism should achieve the objective not only with the best accuracy, but also with the lowest computational cost. So far, many studies have tried to detect B waves using different methods on low-resolution data (<20 Hz):

- FFT: A method essentially based on observing the power level in frequencies where B waves can be present and establish a relation between power and B wave occurrence [26].

- ARMA: This is a parametric spectrum estimator that has been also used to estimate the power level in frequencies where B waves are present [27].

- Wavelets: Basically, in this third method, synthetic-produced signals called wavelets are correlated in time domain with the ICP. Then, the mean squared error between these two signals is used as a measure to determine when a B wave is present or not [28].

In terms of accuracy, the results obtained by ARMA and Wavelets methods were around 70% [27][28].

In the last decade, improvements in electronics have made possible to register ICP at higher frequencies. This has provided more detail to the data acquired by the sensors and consequently other characteristics of ICP have been observed. The most important contribution has been the possibility of observing the pulses of ICP as a consequence of the entry of blood in every heartbeat. Under normal physiological conditions, this pulses occur with amplitudes from 1 to 4 mmHg or 10-30% of the mean ICP[18].

A more detailed analysis of the pulse amplitude shows different wave sub-peaks, which are consecutively termed P1-P5. Normally only P1, P2 and P3 are present. It has been observed that the morphology of the pulse waveform changes depending on the state of the patient [29].

Figure 3.6: P-wave Morphology evolution from well-being (left) to pathologic state (right) [29].

It has been observed in many infusion tests that, when a patient is in normal conditions of ICP, P1 is higher than the other sub-peaks. As intracranial pressure starts growing and the compensatory system have problems, P2 rises until it is higher than P1 and P3. When the state of the patient is very bad, P1 and P3 cannot even be distinguished [29].

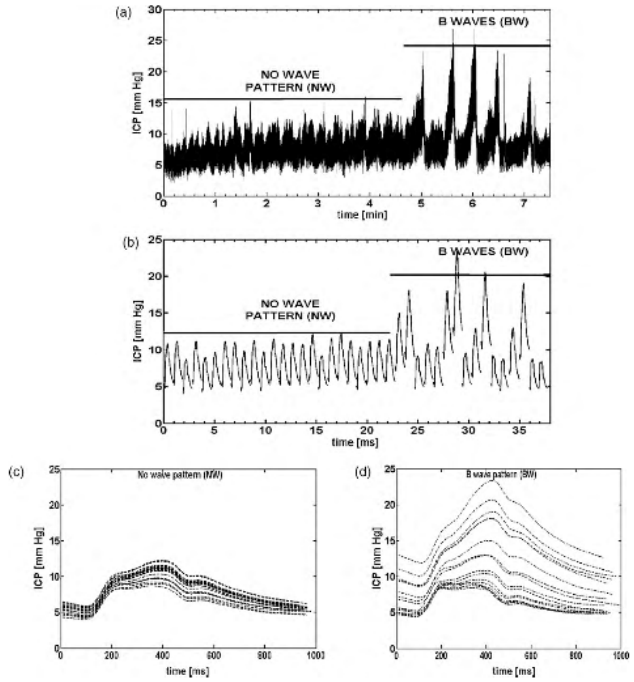


Figure 3.7: An example of nearly flat ICP recording (NW-no slow wave pattern) followed by clearly distinguishable ICP slow waves (BW). (a) Row ICP data, (b) sequence of dominant pulses calculated from 30 s of ICP signal, (c) low dispersed overlapped dominant pulses calculated for NW-. and (d) high dispersed overlapped dominant pulses calculated for BW [30].

Since B-waves are related with ICP instability, some studies have tried to detect Lundberg waves using the morphology and the changes in the morphology of the Pulse waves [30].

The method to extract the features from the pulses is called MOCAIP (Morphological Clustering and Analysis of Intracranial Pressure) [31].

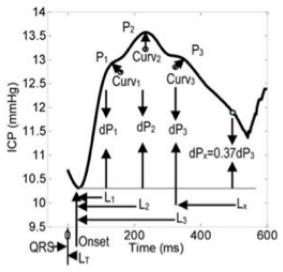


Figure 3.8: Morphological features extracted from a Pulse wave [30].

The most successful study was published as an article in Journal of Neuroscience Methods with the name *Pattern recognition of overnight intracranial pressure slow waves using morphological features of intracranial pressure pulse* in 2010. In this study, 24 features were extracted from different ICP pulses and were analysed in order to find out which of them were better for B-waves detection. The final result was a 88,9% of accuracy, a 96,3% of specificity and a 83% of sensitivity using a quadratic classifier [30].

After this, the same researchers went beyond and studied the way to identify the kind of the B-wave by looking at the morphology of the pulses [32].

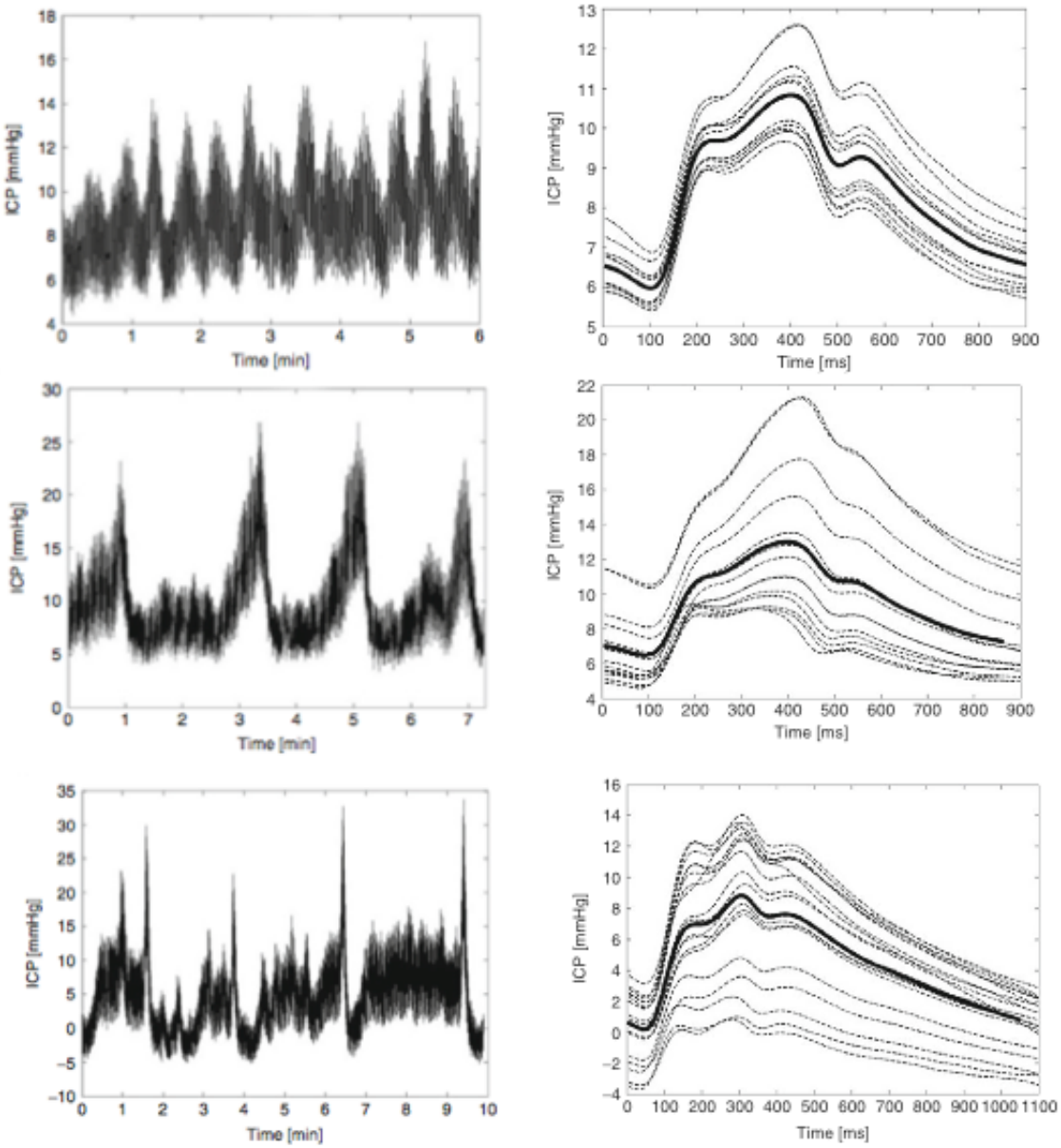


Figure 3.9: Morphology changes of P waves (right) for every kind of B waves (left) [32].

In this study, they could also bear out that, when B-waves are not present, the variances of the curves corresponding with the different pulses waveform are much lower, what confirms the relation between ICP instability and B waves occurrence:

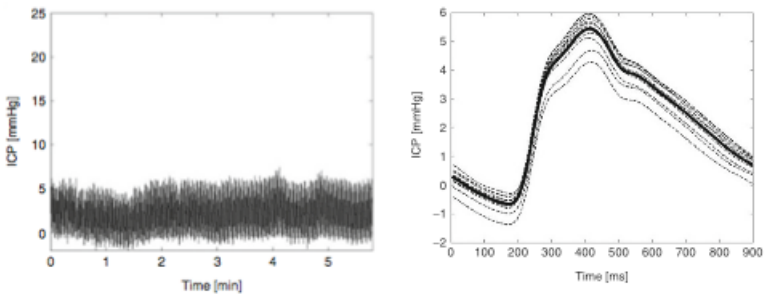


Figure 3.10: Morphology changes of P waves when B waves are not present [32].

3.3 Binary classification

Detecting B waves is reduced to a classification problem. The system must be able to classify an input signal in two different cases:

- Signal with B waves present.
- Signal with lack of B waves.

Two classifiers have been developed to work on iShunt. One of them will be used when the data acquisition system works at 1 Hz and the other when works at 100 Hz.

The first algorithm (1 Hz) uses exactly the definition of B wave to detect them. In the 3th chapter of this thesis B waves have been described as fluctuations with more than 3 mmHg of amplitude and 0.5-2 minutes of period (Lundberg). For this classification system the classifier is the B wave definition.

As a difference with other methods like spectrum estimators and wavelets, this method will not look at the shape. Analysing the spectrum of the ICP, it can be seen that in the range of frequencies of slow waves there are no other processes. According to that, in this range of frequencies there will be noise or a signal corresponding with a B-wave plus noise. It seems that there is no need to look at the shape of the signal because, physiologically, it will be always a shape corresponding with a B-wave plus noise or just noise.

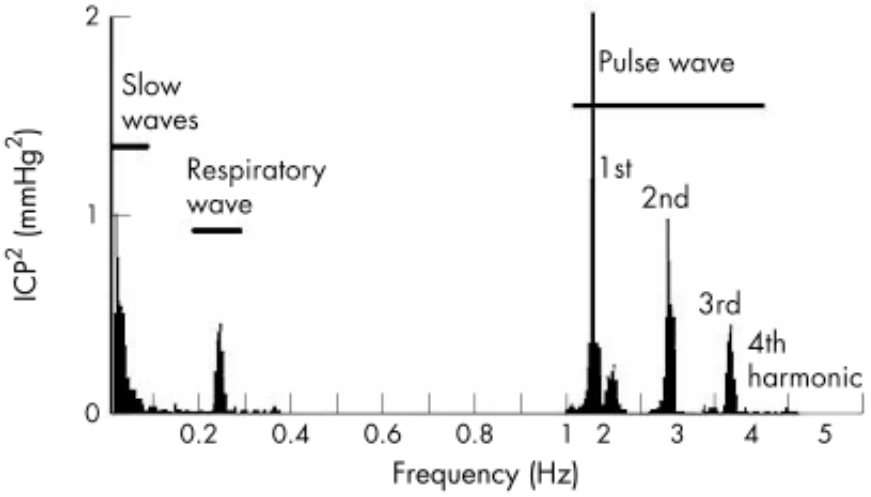


Figure 3.11: Physiological process in spectrum of ICP [33].

The other algorithm that works with 100 Hz data is based on recent studies about P waves [30][31]. These studies claim that a relation between the morphology of P-waves and the presence of B-waves exists. This classifier will be compound by a P wave features extractor and a classifier. The theory of next pages is related to the classifier for this last method.

3.3.1 Description of the classification problem

The goal of a classification system is to identify which class an element belongs to by looking at some characteristics of this element. This kind of problem has been solved since some decades by many different methods, starting by the first theories based on statistics until the last proposals related with Artificial Intelligence. Nevertheless the essence of the problem is still the same:

- Elements: Are represented by a vector with different characteristics that describes them.
- Classes: Divide the elements in different groups. Depending in how many classes are involved the problem is called binary or multiclass classification.
- Classifier: Associates every Element with a class. To validate a system the participation of an expert is usually required.

In a binary classification problem these two definitions are usually formalised as follows:

- Every element is represented by a vector $\mathbf{x} \in \Re^{\mathbf{n}}$.

- Every class is represented by y $\in \{-1, 1\}$. Process that is also called labelling.
- A binary classifier is usually described by the following function:

$$y = g(\mathbf{x}) = sign(f(\mathbf{x})) \tag{3.1}$$

- The union $\{\mathbf{x}, y\}$ is called dataset and links a class with an element. In most of cases the participation of an expert is required to build the datasets. All the datasets are stored in a database and are used to design and validate the classification system.

In the picture below, the global structure of a classification system is represented. For every input element a vector x with its characteristics is build and afterwards is judged by the system.

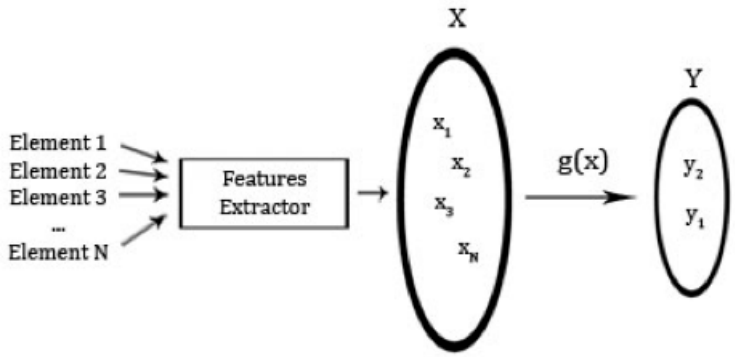


Figure 3.12: General schedule of a classification problem.

Solving a classification problem is reduced to find f(x). When a problem accepts a solution with f(x) being linear it is said that patterns are linearly separable. It can be easily exemplified with the B-wave detection problem:

Given a piece of ICP signal it is want to know if a B wave is present or not by looking at the frequency and the amplitude of this signal.

Element: Piece of signal.

Vector: Describes the element: $\mathbf{x} = [Amplitude, Frequency] \in \Re^2$.

Class: $y =$ { -1: B-wave not present, 1: B-wave present }.

If an expert (in our case a doctor) has already judged this element we have a dataset.

{x, y}: {Element, Judgement of the doctor}.

System: It will be designed and tested using datasets from the a database.

All the datasets $\{\mathbf{x}, y\}$ can be represented in a Cartesian coordinate system:

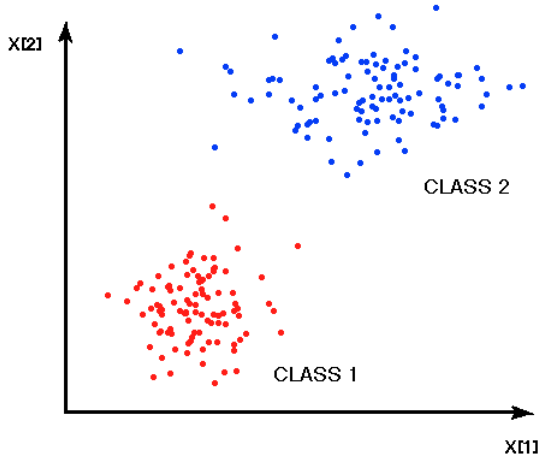


Figure 3.13: Unsolved binary classification problem.

In the picture above elements from two different classes are represented by their vectors. It is easy to see how the elements that belong to the same class follow a similar pattern. The optimal solution for the classification is the line that separates all the examples with the minimum number of errors and makes fewer mistakes with future examples. This last ability is called generalization and is a key factor in the evaluation of the classifier. In the next illustration it can be seen that the number of possible solutions can be infinite.

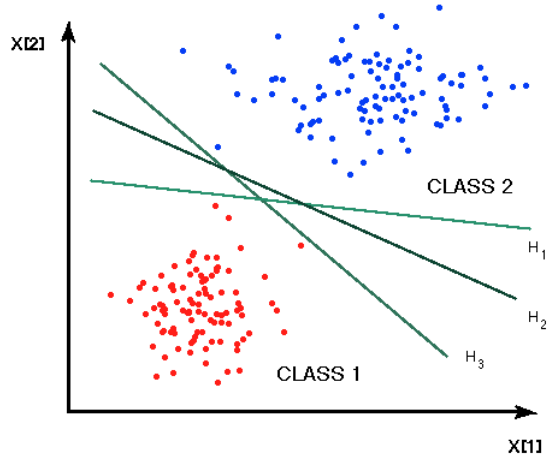


Figure 3.14: Different solutions for a binary classification problem.

The lines H_i represent some of the different solutions for this example. In general, they can be expressed:

$$H_i = \{\mathbf{x} \in \Re^n | f(\mathbf{x}) = 0\} \tag{3.2}$$

After the relation f(x) has been found, the system is evaluated by different parameter.

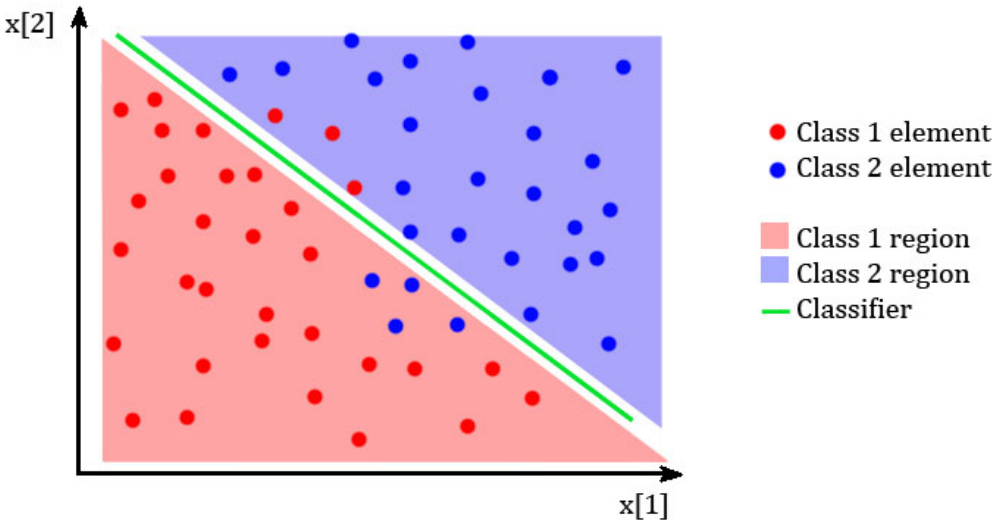


Figure 3.15: Terminology for a classification problem.

Continuing with the example of the B waves, let us suppose:

1. Class 1 element (red points): B wave present.
2. Class 2 element (blue points): Lack of B wave.
3. Class 1 region: Classifier decides that B wave present.
4. Class 2 region: Classifier decides that there is lack of B wave.

then,

Definition	**Element Class**	**Classifier decision**
True positive (TP)	B-wave present	B-wave present
False positive (FP)	B-wave not present	B-wave present
True negative (TN)	B-wave not present	B-wave not present
False negative (FN)	B-wave present	B-wave not present

Table 3.3: Parameters to evaluate the solution of binary classification problem.

Parameters of evaluation:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{3.3}$$

$$Specificity = \frac{TN}{TN + FP} \tag{3.4}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{3.5}$$

In the last pages it has been named the different elements that are involved in a classification process and also which is their main function. The success or the failure of the system depends essentially on:

Database: Contains all the datasets that will be used to design and validate the classification system. Building a data base under the correct criterions is one of the most important parts.

Classifier: It will find the relation between the elements and the classes. There is an immense amount of different classifiers. The criterions of selection will be explained in the next pages.

3.3.2 Database

The database is a matrix that contains all the datasets that will be used to design and validate the classifier. This matrix must be build according to different criterions in order to obtain the optimum results in the final system. Nevertheless, a well-designed matrix is not enough too guarantee satisfactory results.

- The first criterion is a relation between the number of datasets of the database N and the dimension of every dataset d. The structure of a database is described in the next picture:

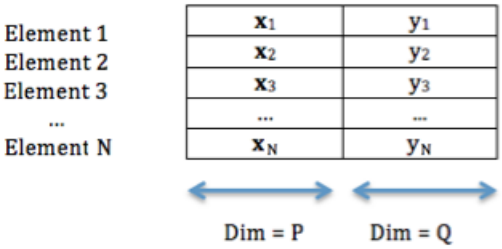


Figure 3.16: Database structure.

The intrinsic dimension can be calculated as: $d = P + Q$. In some books about pattern classification [34] [35] it can be found that the number of datasets necessary to describe the vector space of the datasets can be computed as:

$$N \approx 2^P \tag{3.6}$$

This value is a maximum. In a lot of cases it is possible to achieve satisfactory results with less than 2^P datasets in the database.

- Partitioning of the database: The database should be divided in three parts: the design or training set, validation set and the testing set. The goal of every set is explained in more detail in the next pages.

- Erasing outliers: Sometimes there are strange values that are too far from the mean value. This kind of elements may have an undesirable effect in the training process, so it's better to erase them. The parameters used to erase these values are the mean and the variance.

- Reducing dimension: The database may contain different features that provide the same information and therefore are expectable. A tool to identify these features is the covariance. Covariance describes how a variable can change depending on another variable.

$$\sigma_{xy} = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \overline{x})(y_i - \overline{y})}{N-1}} \tag{3.7}$$

This reduction can also be done using neural networks or auto-organized maps (also called Kohonen maps).

A more elegant and less drastic method for reducing the dimension of the database is PCA. This method uses the eigenvectors and the eigenvalues of the covariance matrix to project the original vectors into a new base with lower dimension but that keeps the most of the signal power.

Given a set of sample:

$$D = \{\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_N}\} \text{ with } \mathbf{x} \in \Re^d \tag{3.8}$$

It is desired an unitary matrix that approximates:

$$\mathbf{x}_k \simeq \mathbf{W}\mathbf{y}_k + \mathbf{m} \text{ with } \mathbf{m} = \frac{1}{N}\sum_{\mathbf{x}\in D}\mathbf{x} \tag{3.9}$$

Following the criterion of minimum square error:

$$J = \sum_{k=1}^{N} \|\mathbf{W}\mathbf{y}_k + \mathbf{m} - \mathbf{x}_k\|_2^2 \tag{3.10}$$

The mean is subtracted because in some situations it provides more separability.

- Scaling data: It is important that all data belongs to the same order of magnitude in order to give the same importance to every feature. There are several ways to do it:

  – Linear scaling:

  $$s_i = \frac{v_i - min(v_1...v_N)}{max(v_1...v_N) - min(v_1...v_N)} \tag{3.11}$$

  – "Mean zero and variance one" transformation:

  $$s_i = \frac{v_i - \overline{v}}{\sigma} \tag{3.12}$$

  $$\overline{v} = \frac{1}{N}\sum_{i=1}^{N} v_i \tag{3.13}$$

  $$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(v_i - \overline{v})^2}{N-1}} \tag{3.14}$$

  If the system is designed or trained using scaled data, the system will require scaled data also for work. Since the linear scaling method is lighter for the microcontroller, it will be used in the practical part of the project.

3.3.3 Classifiers

Nowadays there are many kinds of classifiers and most of them are suitable for many different problems. Some questions arise from this: Given a specific problem is there any classifier better than the others? Which one? The answer to these two questions is not clear but there are some clues that reduce the amount of options. Some of these criterions are:

- Database structure:

  - Labelled datasets: Supervised methods.
  - Non-labelled datasets: Unsupervised methods.

- Availability of previous knowledge about the features: Known pdf?
- Complexity and computational cost.
- Amount of datasets.

In order to discuss about the first criterion, first it is necessary to define what supervised and unsupervised means.

In the supervised learning, an expert provides a category label for each pattern in a training set, and we seek to reduce the sum of the costs for these patterns.

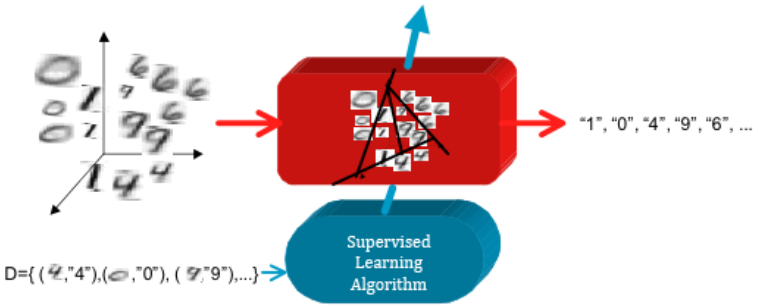


Figure 3.17: Example of supervised learning [34].

In unsupervised learning (or clustering) there is not an expert, and the system forms clusters or *natural groupings* of the input patterns. *Natural* is always defined explicitly or implicitly in the clustering system itself, and given a particular set of patterns or cost function, different clustering algorithms lead to different clusters. Often the user will set the hypothesized number of different clusters ahead of time.

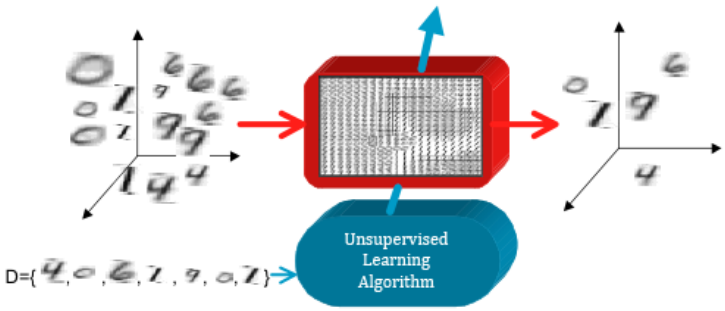


Figure 3.18: Example of unsupervised learning [34].

Since our database is labelled, we will use a supervised method. The second criterion regards on the availability of the statistical behaviour of the different features (or if there is any way to obtain it). This divides classification methods in two main groups:

- Parametric Methods: These are classifiers that assume a particular pdf for the data. Consequently, they provide a proper performance for data that belong to this pdf. Otherwise, the model doesn't fit with the actual problem and it performs a wrong classification. They are based in Bayesian Theory and are frequently applied to data with Gaussian pdf. From this definition arises the necessity of estimating the pdf of a given group of datasets:

  – Histogram: It is a first approach of the pdf of the data. The histogram provides a visual impression of which is the distribution of the data.

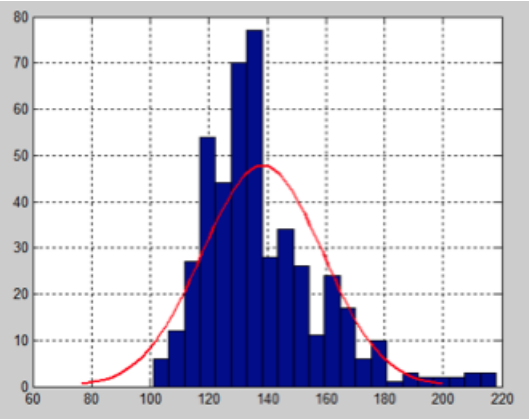


Figure 3.19: Histogram of a non-Gaussian distribution (blue bars) compared with a gaussian pdf (red line)

  – Skewness: It is a measure of symmetry, or more precisely, of lack of symmetry. A distribution, or data set, is symmetric if it looks the same to the left and right of the centre point:

$$Skewness(x) = \frac{\mu_3}{\mu_2 \cdot \sqrt{\mu_2}} = \frac{E[(x-\mu)^3]}{\sigma^3} \tag{3.15}$$

Where:

  * $E[\cdot]$ is the expected value or mathematical expectation.
  * $\mu = E[x]$.
  * $\mu_n = E[(x-\mu)^n]$.

Negatives values for the skewness indicate data are skewed left and positive values for the skewness indicate data are skewed right. Similarly, skewed right means that the right tail is heavier than the left tail.

– Kurtosis: It is a measure of wether the data are peaked or flat relative to a normal distribution. That is, data sets with high kurtosis tend to have a distinct peak near mean, decline rapidly and have heavy tails. Datasets with low kurtosis tend to have a flat top near the mean rather than a sharp peak.

$$Kurtosis(x) = \frac{\mu_4}{\mu_2^2} - 3 = \frac{E[(x-\mu)^4]}{\sigma^4} \tag{3.16}$$

Positive kurtosis indicates a peaked distributions and negative kurtosis indicates a flat distribution.

- Non-parametric Methods: This kind of classifiers don't assume any pdf for the data, so they can be applied to any kind of data, Gaussian or not. Most common algorithms that belong to this type are:

  – K-nearest neighbours: This is a classifier that requires a very low computational cost but in contrast it needs a lot of memory. Basically, its performance consists on looking to the class of the k datasets that are the closest to the dataset that is going to be classified. It is commonly used when databases have a big number of features compared to the amount of datasets.

  – Decision trees: This classifier consists in a sequence of questions that regard to the features of the datasets. Different branches hang from the root node and connect with another nodes forming a structure of tree with different levels. This classifier is suitable for databases that contain non-numeric features. One of their problems is that they easily get overtrained and after training, another process called pruning is required to reduce the number of levels of the tree. They have low computational cost.

  – Artificial Neural Networks: ANN were inspired by the functioning of neurons. These networks are compound by perceptrons, which emulate mathematically single neurons. By using training algorithms this combination of perceptrons fits to the problem and perform a better or worst classification depending on the structure chosen. They are powerful classifiers capable to solve complex problems successfully. The main disadvantage regards on the complexity of deciding which structure is better for a given problem.

  – Support Vector Machines: SVM classifiers are a relatively new set of algorithms and they are based on maximizing the distance between the threshold and the

two classes. They are extremely powerful, complex, cumbersome to train and slow to evaluate. In addition they have large memory requirements. They work well for small data sets but the computational requirements increase dramatically with the size of the training set.

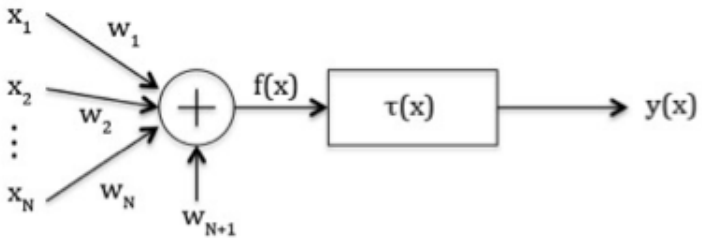
After analysing the database in section 4.2.3 it has been observed that features don't follow any specific pdf, so a non-parametric method should be used. Looking at the histograms included in the appendix A.3 it can be intuit that boundaries between the two classes will be pretty complex. For this reasons MLP and SVM will be trained and compared in section 4.2.4, since they are really powerful classifiers. Even though they are more cumbersome to run than other classifiers, they will be approximated in order to be adapted for the microcontroller of iShunt.

3.3.4 Machine Learning

Neural networks: Multilayer perceptron

Multilayer perceptron is a very common used kind of Neural Network for problems of pattern classification. The basic unit that makes up an MLP is called perceptron.

A perceptron is a mathematical model inspired by the anatomy of neurons. A perceptron's operation is very simple and it is described in picture 3.20.



$$f(x) = \sum_{n=1}^{N} x_n \cdot w_n + w_{N+1} \tag{3.17}$$

Figure 3.20: Schedule of a perceptron and its transfer function.

Ideally sign function is used as $\tau(x)$. Unfortunately, this function is non-derivative, which is a required by the optimization algorithms. As an alternative, some approximations can be used, being the most common the sigmoid function:

$$\tau(x) = \frac{1}{1 - e^{-\beta x}} \tag{3.18}$$

The input-output relationship can be written as:

$$y(x) = g(x) = \tau(f(x)) \tag{3.19}$$

Given this structure, an optimization algorithm is applied to find the weights that minimize the classification error. There are several algorithms able to obtain the weights. The most used are PLR, Adeline and Delta rule and all of them work following the gradient descent technic.

The most important limitation of the perceptron for pattern classification is that it is only able to solve problems linearly separable. In order to solve nonlinearly separable problems, a more complex structure must be build. This is called Multilayer Perceptron.

The design process of a MLP consists in three main steps.

- Choosing an initial structure.
- Training.
- Validation.

This process is repeated some times, doing small changes in the MLP structure, until a satisfactory result is obtained.

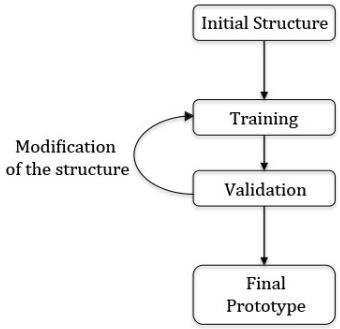


Figure 3.21: Iterative process to obtain a final MLP prototype

The general structure of a MLP is shown in the illustration below. Every circle must be interpreted as a simple perceptron as the one explained in the last page.

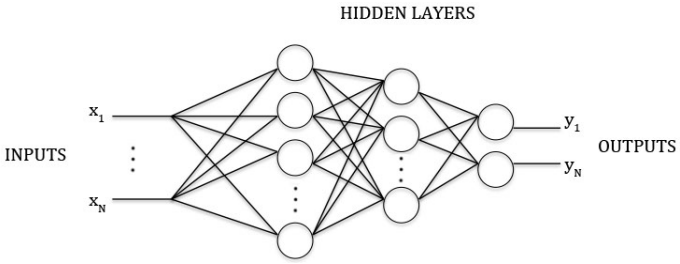


Figure 3.22: Multilayer perceptron structure.

The main problem while working with MLP is which dimensions to select for the hidden layers. There are no rules to obtain the optimum MLP for every problem. This is the reason why usually it is necessary to try different configurations of MLP to find the best solution. Nevertheless there are some criterions that are recommended to be followed when building an MLP.

- The first criterion is related with the dimensions of the database. Information theory says that the order of the number of datasets N must be in the same order with the amount of unknown parameters. In a MLP the amount parameters can be calculated as:

$$W = (I + 1)J + (J + 1)K \longrightarrow N \approx W \tag{3.20}$$

  Where:

  - I = number of inputs.
  - J = number of neurons in the hidden layer.
  - K = number of outputs.
  - W = Amount of parameters.

  If the prototype will have only one hidden layer (this is the most usual), there's a second criterion that links the number of neurons in the hidden layer with the number of inputs and outputs:

$$J = \sqrt{IK} \tag{3.21}$$

  - I = number of inputs.
  - J = number of neurons in the hidden layer.
  - K = number of outputs.

Given a structure, the algorithm used to train the MLP is called Backpropagation (BP).

Supported Vector Machine

Support Vector Machines (SVM) [35][34] is a set of algorithms developed by Vladimir Vapni and its teamwork in 1992. In the beginning, these algorithms were used for classification problems of 2 patterns linearly separable. After years some contributions made them able to classify patterns nonlinearly separable. This kind of classifiers has a lot of advantages over the other algorithms. This is the most probable reason why in the last years SVM have become very popular in a big amount of classification problems as for example:

- Temporal series prediction.
- Image classification.
- Typing recognition.

Given a classification problem of two classes linearly separable, where there's more than one solution, the following question can be formulated: Is there any of the solutions better than the others?

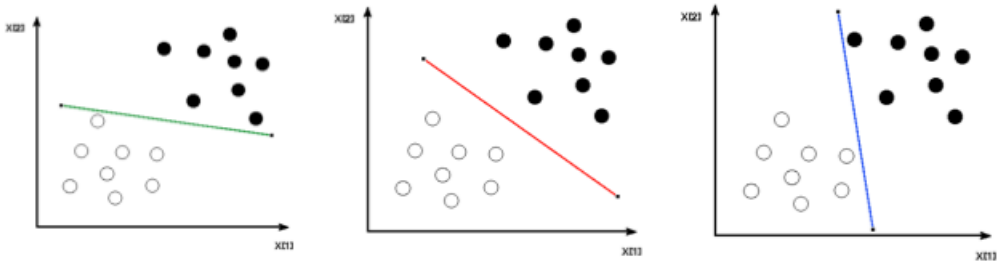


Figure 3.23: Different solutions for the same classification problem.

If we could choose a solution from the 3 options in the image, most probably we would choose the second one. It can be intuitively seen that being as far as possible from the datasets will drive to find a better solution.

The main contribution of SVM is that they are able to find the threshold or hyperplane that maximizes the distance between the hyperplane and the points that are closer to it.

The function used in SVM to pass from the input to the output space is called hyperplane and is formulated as:

$$f(x) = \mathbf{w}^T \cdot \mathbf{x} + b \tag{3.22}$$

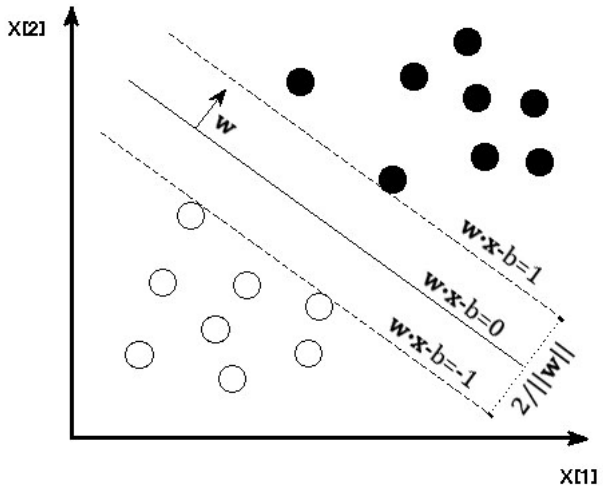


Figure 3.24: Mathematical description of the hyperplane in SVM.

Defining the output space as in section 3.3, and knowing that, by definition f(x) is linear, the threshold that separates both classes is:

$$\mathbf{w}^T \cdot \mathbf{x} + b = 0 \tag{3.23}$$

And the distance or marge between the two classes can be expressed as:

$$m = \frac{2}{\| \mathbf{w} \|} \tag{3.24}$$

Finding the hyperplane that maximizes the margin between the two classes is an optimization problem with restrictions, which is solved with Lagrange multipliers and the conditions of Karush-Kuhn-Tucker (used to solve problems with inequality constraints).

Function to maximize:

$$\text{maximize } \frac{2}{\| \mathbf{w} \|} \equiv \text{ minimize } \frac{1}{2}\mathbf{w}^T \cdot \mathbf{w} \tag{3.25}$$

Subject to:

$$y_n(\mathbf{w} \cdot \mathbf{x}_n + b) \geq 1 \text{ for n=1,2,...,N} \tag{3.26}$$

$$\mathbf{w} \in \Re^P \text{and } b \in \Re$$

Applying Lagrange, the expression to minimize is:

$$\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{n=1}^{N} \alpha_n(y_n(\mathbf{w}^T\mathbf{x}_n + b) - 1) \tag{3.27}$$

w.r.t $\mathbf{w}$ and b and maximizes w.r.t each $\alpha_n \geq 0$

Process:

$$\nabla_{\mathbf{w}}\mathcal{L} = \mathbf{w} - \sum_{n=1}^{N} \alpha_n y_n \mathbf{x_n} = 0 \tag{3.28}$$

$$\frac{\partial \mathcal{L}}{\partial b} = -\sum_{n=1}^{N} \alpha_n y_n = 0 \tag{3.29}$$

Substituting in $\mathcal{L}(\mathbf{w}, b, \alpha)$, is obtained:

$$\mathcal{L}(\alpha) = \sum_{n=1}^{N} \alpha_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} y_n y_m \alpha_n \alpha_m \mathbf{x}_n^T \mathbf{x}_m \tag{3.30}$$

maximize w.r.t α subject to $\alpha_n \geq 0$ for n=1,2,...,N

To use quadratic programing a minimization problem is required. First of all we need to convert this problem into a minimization problem.

$$\text{maximize } \sum_{n=1}^{N} \alpha_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} y_n y_m \alpha_n \alpha_m \mathbf{x}_n^T \mathbf{x}_m \tag{3.31}$$

$$\text{minimize } \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} y_n y_m \alpha_n \alpha_m \mathbf{x}_n^T \mathbf{x}_m - \sum_{n=1}^{N} \alpha_n \tag{3.32}$$

The last expression can be rewritten as:

$$\text{minimize } \tfrac{1}{2}\alpha^T \begin{pmatrix} y_1 y_1 \mathbf{x}_1^T \mathbf{x}_1 & \dots & y_1 y_1 \mathbf{x}_1^T \mathbf{x}_N \\ \vdots & \ddots & \vdots \\ y_1 y_1 \mathbf{x}_N^T \mathbf{x}_1 & \dots & y_1 y_1 \mathbf{x}_N^T \mathbf{x}_N \end{pmatrix} \alpha - \mathbf{1}^T \alpha$$

$$\text{subject to } \mathbf{y}^T\alpha = 0, \mathbf{y} \geq 0$$

This is a convex function with just one absolute minimum. It means that, once solved this system by quadratic programing, the result obtained will be the optimum. After the quadratic programming solves the problem, we get back α. Most of the values of the vector alpha will be 0 because of this condition:

$$\alpha_n(y_n(\mathbf{w}^T\mathbf{x}_n + b) - 1) = 0 \tag{3.33}$$

Looking at the equation, or α_n is 0 or $y_n(\mathbf{w}^T\mathbf{x}_n + b) - 1$ is 0. So α_n will be 0 unless some cases where x_n makes $y_n(\mathbf{w}^T\mathbf{x}_n + b) - 1$ equal to 0. These x_n are called support vectors.

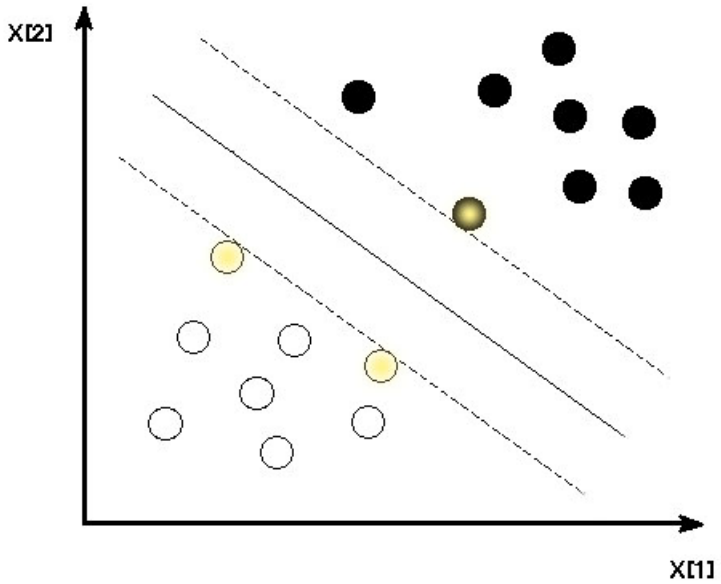


Figure 3.25: Supported vectors (yellow).

In order to obtain $\mathbf{w}$, one of the equations resulting from derivate is used:

$$\nabla_{\mathbf{w}}\mathcal{L} = \mathbf{w} - \sum_{n=1}^{N} \alpha_n y_n \mathbf{x_n} = 0 \tag{3.34}$$

$$\mathbf{w} = \sum_{n=1}^{N} \alpha_n y_n \mathbf{x_n} \tag{3.35}$$

To find b, just substitute any support vector in this equation:

$$y_n(\mathbf{w}^T\mathbf{x}_n + b) = 1 \tag{3.36}$$

All this is only useful when classes are linearly separable. When they are not there is the possibility to convert X in a bigger dimension space where the problem is linearly separable. This is commonly called the kernel trick and is exemplified in the picture below.

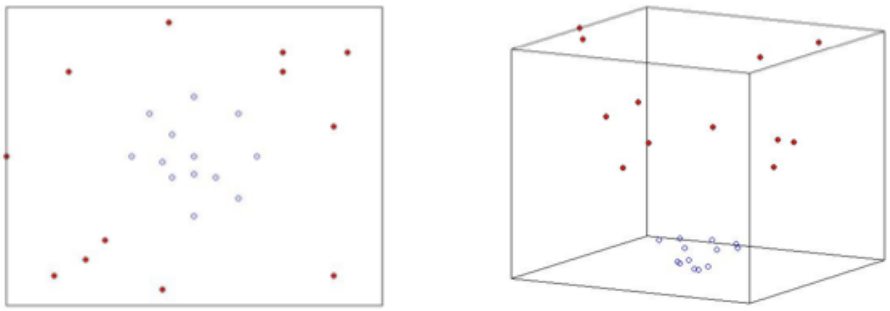


Figure 3.26: Original space X (left) where the problem is non linearly separable. New space Z where the problem is linearly separable (right).

There are different functions or kernels that transform the product $\mathbf{x}^T\mathbf{x}$ from the original space X into the product $\mathbf{z}^T\mathbf{z}$ from the new space. The most common used are:

- Polynomial:
$$K(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T\mathbf{y} + 1)^d \tag{3.37}$$

- Radial Base:
$$K(\mathbf{x}, \mathbf{y}) = e^{\frac{-\|\mathbf{x}-\mathbf{y}\|}{2\sigma^2}} \tag{3.38}$$

- Sinusoidal
$$K(\mathbf{x}, \mathbf{y}) = tanh(k\mathbf{x}^T\mathbf{y} + \theta) \tag{3.39}$$

So now, the procedure is the same but working in space Z.

$$f(\mathbf{z}) = \mathbf{w}^T\mathbf{z} + b \tag{3.40}$$

the function to maximize is:

$$\mathcal{L}(\alpha) = \sum_{n=1}^{N} \alpha_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} y_n y_m \alpha_n \alpha_m \mathbf{z}_n^T \mathbf{z}_m \tag{3.41}$$

The solutions are w and b for the Z space.

In order to express the solution in terms of the Kernel function, the initial equation $f(\mathbf{z}) = \mathbf{w}^T\mathbf{z} + b$ is expressed like:

$$f(\mathbf{z}) = \sum_{n=1}^{N} \alpha_n y_n \mathbf{z}_n^T \mathbf{z} + b \tag{3.42}$$

and now the equation depends again from the original space:

$$f(\mathbf{z}) = \sum_{n=1}^{N} \alpha_n y_n K(\mathbf{x}_n, \mathbf{x}) + b \tag{3.43}$$

and finally, the classifier is:

$$g(\mathbf{x}) = sign(f(\mathbf{x})) = sign(\sum_{n=1}^{N} \alpha_n y_n K(\mathbf{x}_n, \mathbf{x}) + b) \tag{3.44}$$

4 Algorithms for B waves detection

4.1 Algorithm based on Frequency and Amplitude

4.1.1 Introduction

By using this algorithm B waves can be detected when they appear in an ICP signal sampled at 1 Hz rate. It has been designed by considering the limitations that are accompanied by the ultra-low energy microcontroller used in the project.

The global structure of this algorithm can be easily explained by the following diagram:

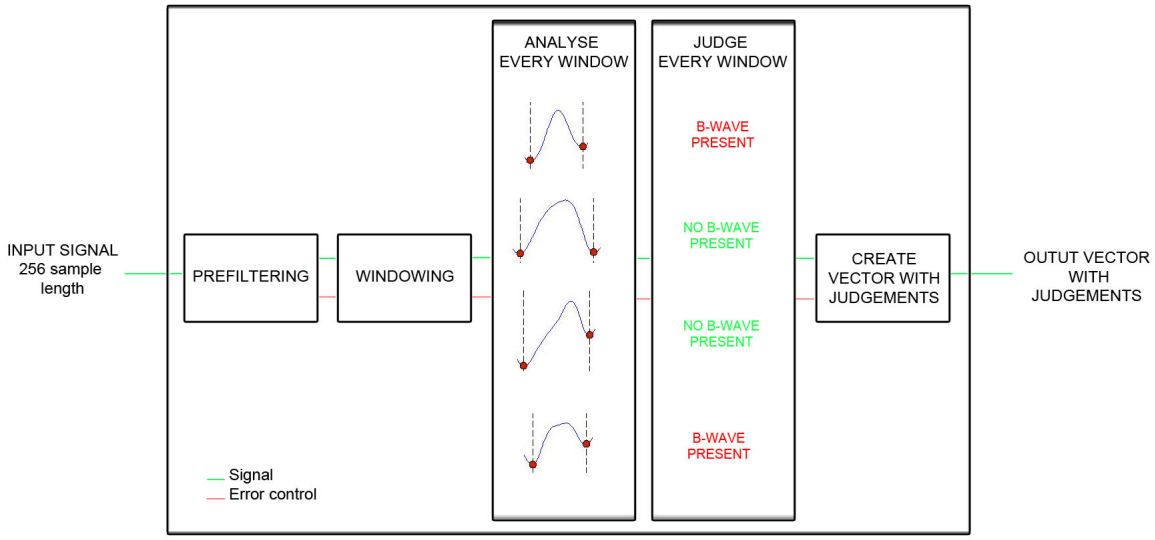


Figure 4.1: Global classification system schedule for data at 1 Hz.

The algorithm is divided in three main parts:

- Erasing respiration: In this part the signal is processed in order to maintain just the frequencies where B waves can be present.
- Windowing: During this procedure the filtered signal is separated in different parts. Every part contains one period of the signal.
- Extracting features and judging: This module analyses every window, extracting the frequency and the amplitude and deciding whether a B-wave is present in this part

of the signal or not.

After every window has been analysed, a vector with the judgments is build and sent out as the output of the system.

4.1.2 Erase respiration

The respiration process can be observed in the ICP signal. The peaks and the frequency of this oscillation change depending on the person and on the moment as well. Nevertheless in adults the frequency of the respiration usually fluctuates between 12 and 20 times per minute (some old books also include values between 8 and 12 times per minute). An example can be seen below:

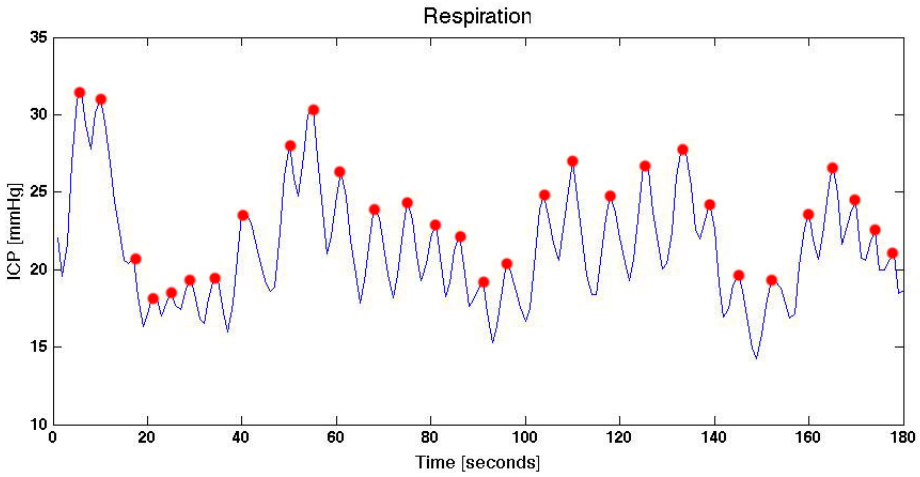


Figure 4.2: Respiration (oscillation with red points) on a B wave.

In this case the frequency is a bit lower than normally, around 10 breaths per minute, but it is still in boundaries. According to the diagram above, peaks are very sharped so a strong filtering process would be necessary to completely remove the respiration from this signal.

The first part of the algorithm consists on removing the respiration without affecting the frequencies where B waves are present. Unfortunately this is impossible when FIR filters are used but we will try to attenuate the B waves as less as possible.

Since we are using an ultra-low energy microcontroller (MSP430) the algorithm needs to be as simple as possible. Two easy operations for the microcontroller are products and divisions by a power of 2. Due to this, a good way to design the system is by the combination of Moving Average Filters with N samples, being N a power of 2.

After different proofs, it has been observed that 3 moving average filters are enough to erase even the most sharped oscillations of the respiration, with frequencies around 0,3 Hz and amplitudes of 6 mmHg.

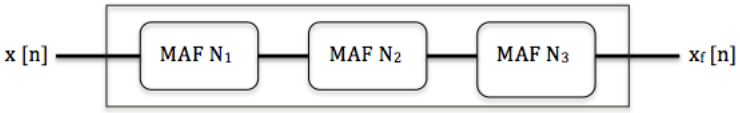


Figure 4.3: Filtering system compound by three MAF filters in cascade.

Some examples using different combinations of N are shown below.

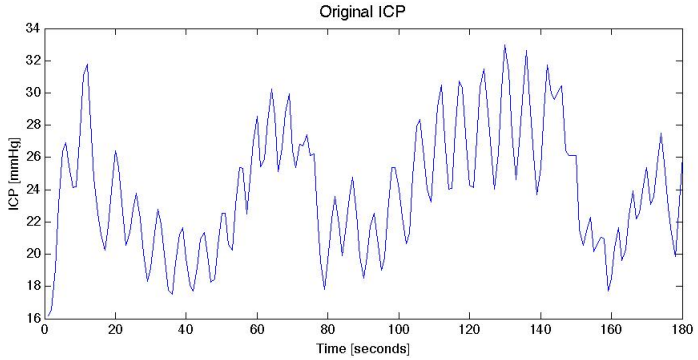


Figure 4.4: B wave before filtering in time domain.

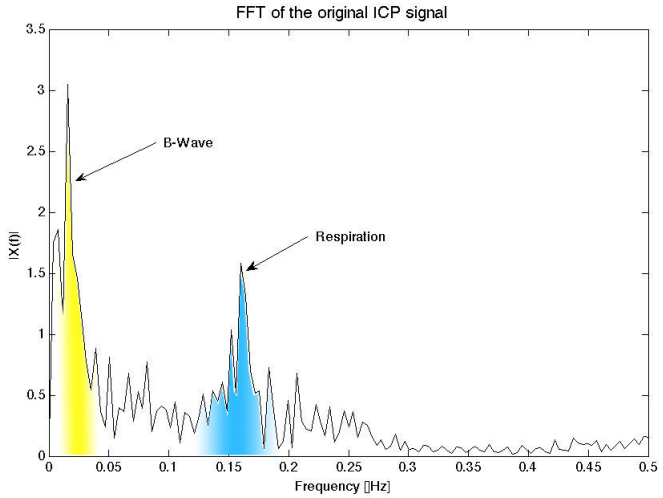


Figure 4.5: B wave before filtering in frequency domain.

Filtering with N1=8, N2=8 and N3=4:

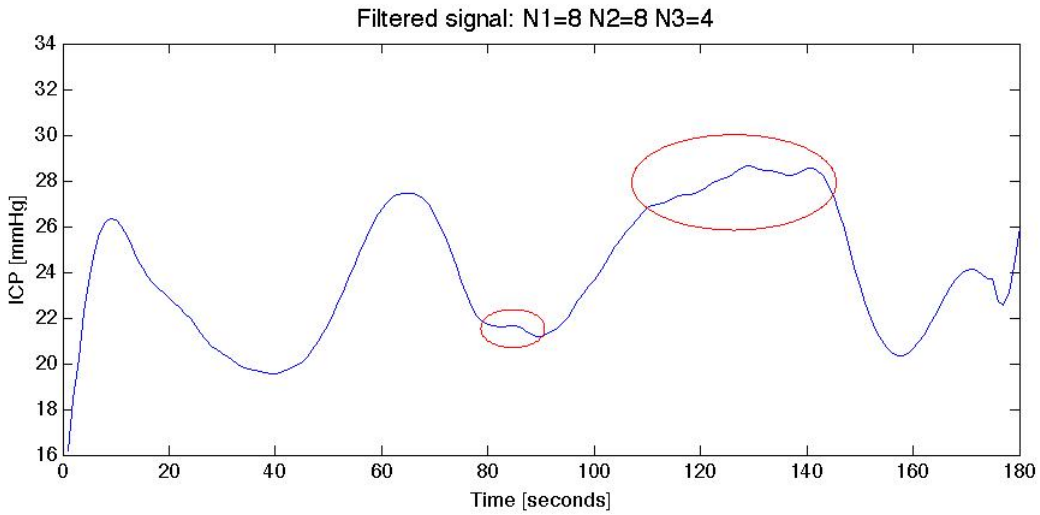


Figure 4.6: B wave after filtering using N1=8, N2=8 and N3=4.

In the picture above it can be seen that during some part of the signal (red circles) respiration is still present. A stronger filtering process is demanded.

Filtering with N1=16, N2=8 and N3=4:

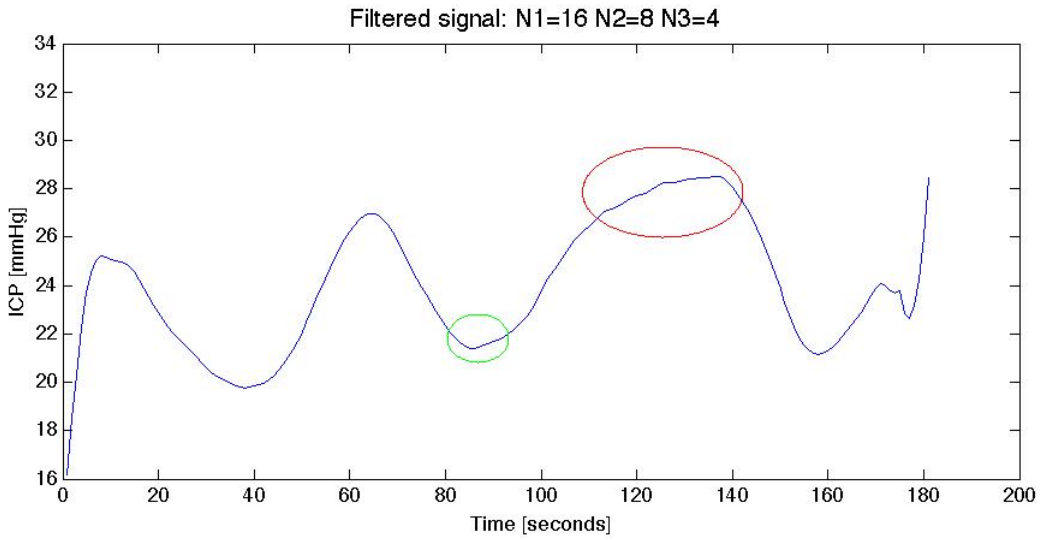


Figure 4.7: B wave after filtering using N1=16, N2=8 and N3=4.

Filtering with N1=16, N2=16 and N3=8:

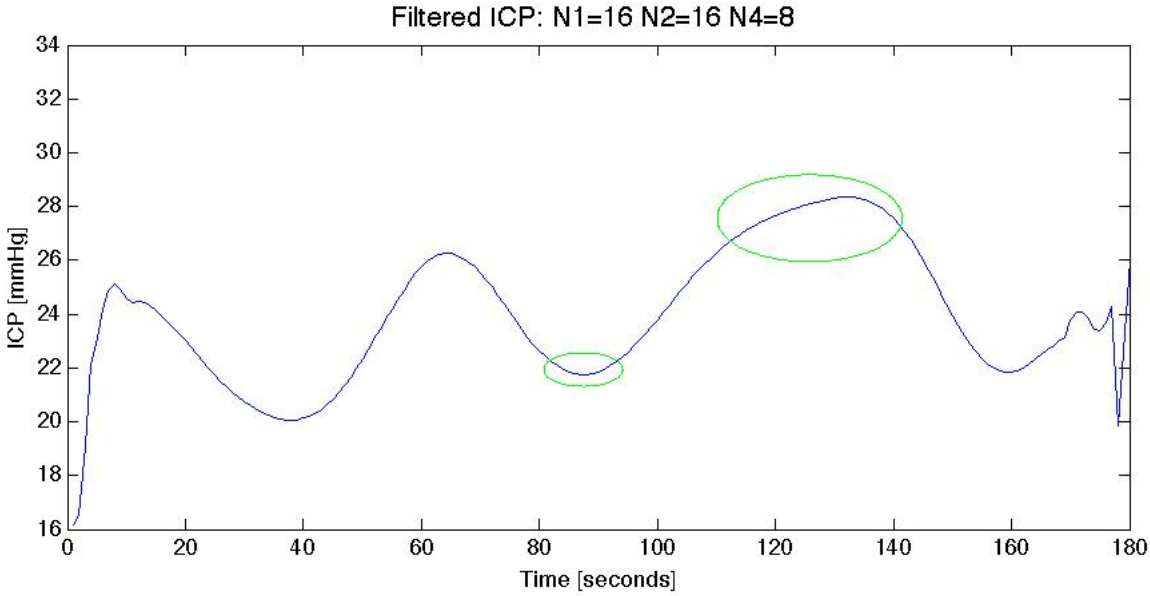


Figure 4.8: B wave after filtering using N1=16, N2=16 and N3=8.

After trying different combinations of filters it has been determined that by using N1=16, N2=16 and N3=8, respiration can be completely eliminated. As a drawback, since we are using longer filters, the permanent response is shorter than using lower values of N. Consequently the signal to be analysed is shorter. Even so, this is the filter that achieves better results using the minimum number of samples, so it will be used.

The next step is to analyse this filter in detail in order to know how it is affecting the frequencies where B waves are present. First of all, the general expression of a moving average filter will be found and then the global response of the system.

Starting with the general expression of a moving average filter, its time response is found:

$$y[n] = \frac{1}{N}\sum_{m=n}^{n+N-1} x[m] \rightarrow h[n] = \frac{1}{N}\sum_{m=0}^{N-1}\delta[n+m] \tag{4.1}$$

Then DFT definition is used to find the frequency response:

$$H(f) = \mathrm{DFT}\{h[n]\} = \sum_{n=0}^{\infty}(\frac{1}{N}\sum_{m=0}^{N-1}\delta[n+m])e^{-j\omega n} = \frac{1}{N}\sum_{m=0}^{N-1}e^{j\omega m} \tag{4.2}$$

Note that $\sum_{n=A}^{B} a^n = \frac{a^A - a^{B+1}}{1-a}$

$$H(f) = \frac{1}{N}\frac{1-e^{j\omega N}}{1-e^{j\omega}} = \frac{e^{j\omega\frac{N}{2}}}{Ne^{j\frac{\omega}{2}}}\frac{e^{j\omega\frac{N}{2}}-e^{-j\omega\frac{N}{2}}}{e^{j\frac{\omega}{2}}-e^{j\frac{\omega}{2}}} = \frac{e^{j\omega\frac{N-1}{2}}}{N}\frac{sin(N\frac{\omega}{2})}{sin(\frac{\omega}{2})} \tag{4.3}$$

Finally, the moving average filter frequency response is obtained:

$$|H(f)| = \frac{1}{N}|\frac{sin(N\pi f)}{sin(\pi f)}| \tag{4.4}$$

The next image represents the frequency response of a moving average filter depending on the number of samples used N and the sample rate used to measure the ICP from the skull. In this case, Fm=1Hz.

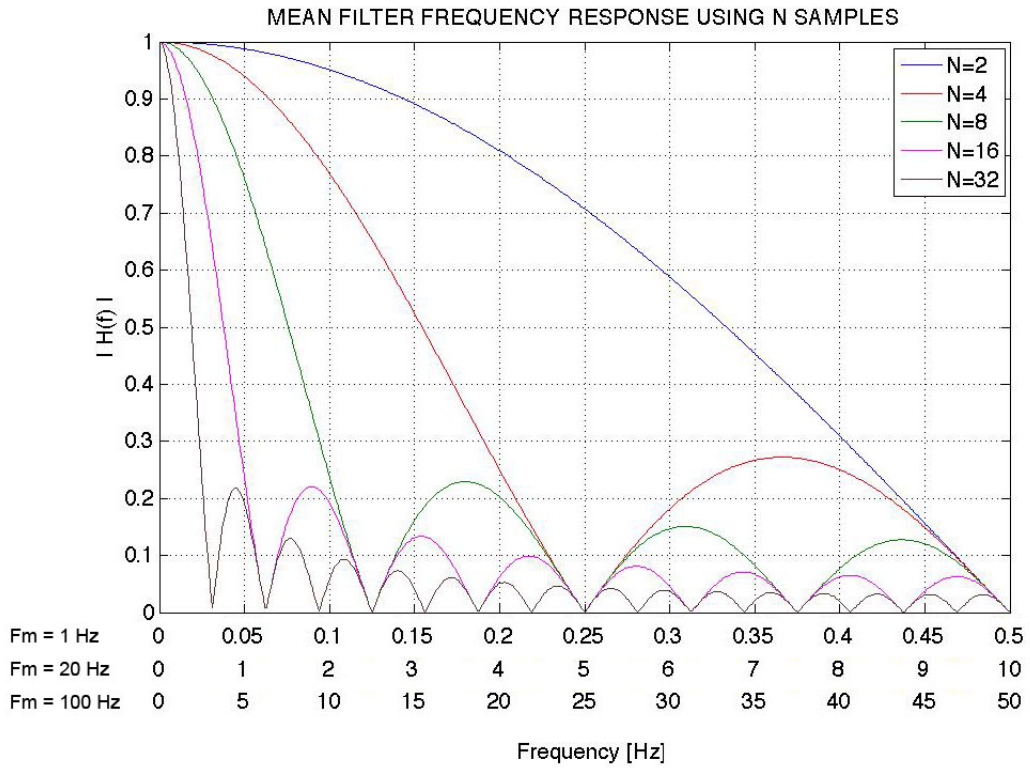


Figure 4.9: Frequency response of a MAF depending on the number of samples used and also in the sampling frequency used for the data acquisition process.

Using the convolution theorem the global response is calculated:

$$h[n] = h_1[n] * h_2[n] * h_3[n] \rightarrow H(f) = H_1(f) \cdot H_2(f) \cdot H_3(f) \tag{4.5}$$

$$|H_1(f)| = |H_2(f)| = \frac{1}{16}|\frac{sin(16\pi f)}{sin(\pi f)}| \tag{4.6}$$

$$|H_3(f)| = \frac{1}{8}|\frac{sin(8\pi f)}{sin(\pi f)}| \tag{4.7}$$

So the global response is:

$$|H(f)| = \frac{1}{2^{11}}|\frac{sin(16\pi f)}{sin(\pi f)}|^2 \cdot |\frac{sin(8\pi f)}{sin(\pi f)}| \tag{4.8}$$

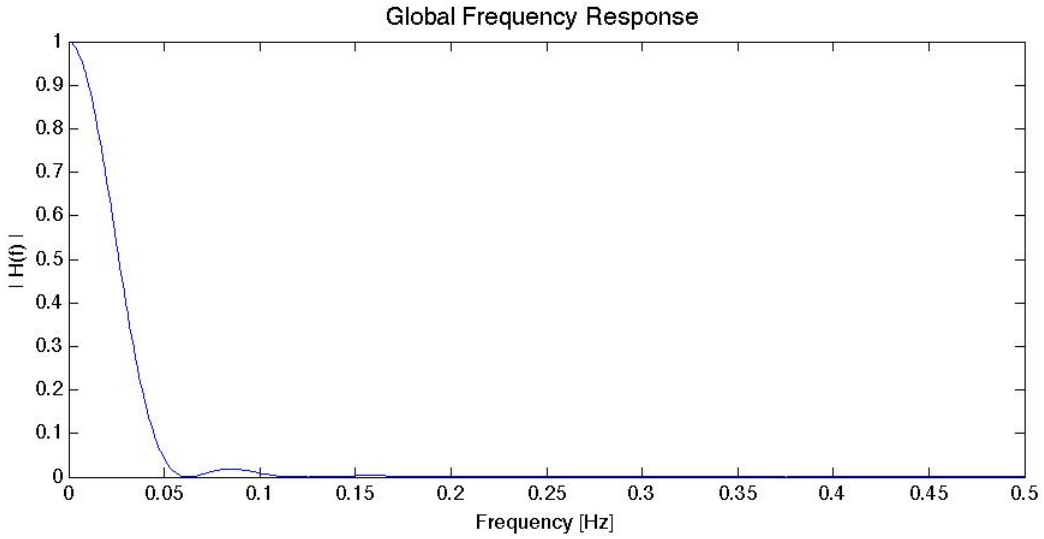


Figure 4.10: Frequency response of the global filtering system.

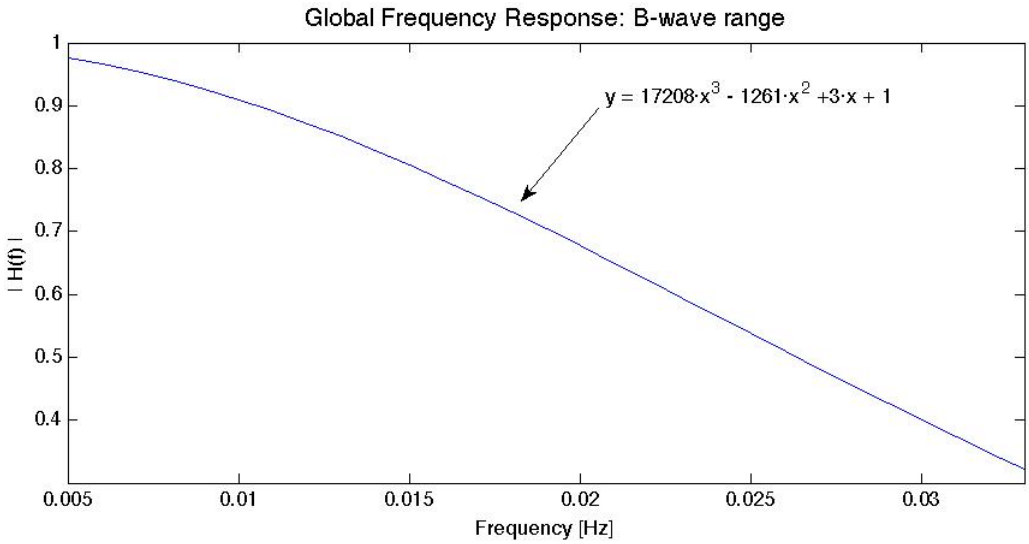


Figure 4.11: : Frequency response of the global filtering system in B waves frequency rang. Polynomial approximation of order 3.

Detecting B waves consists basically in finding out the frequency and the amplitude of the ICP and comparing them to thresholds. To find out this features, the signal that goes out of the moving average filters system will be used. The attenuation follows this curve:

$$\text{Attenuation} = 17208 \cdot f^3 - 1261 \cdot f^2 + 3 \cdot f + 1 \tag{4.9}$$

The threshold used for the amplitude detection has to be attenuated in the same proportion. Note that this equation is valid for sinusoidal signals, but in the case of sawtooth signals the value of the attenuation changes. Since the system is not able to discern between the shapes of the input signals, all of them will be treated as sinusoidal.

4.1.3 Windowing

The next step after erasing the respiration is to divide the filtered signal in sub-signals that contain just one period of the global oscillation. The signals between consecutive notches are the sub-signals to be obtained with this process. The algorithm just needs to detect these notches in order to form the windows.

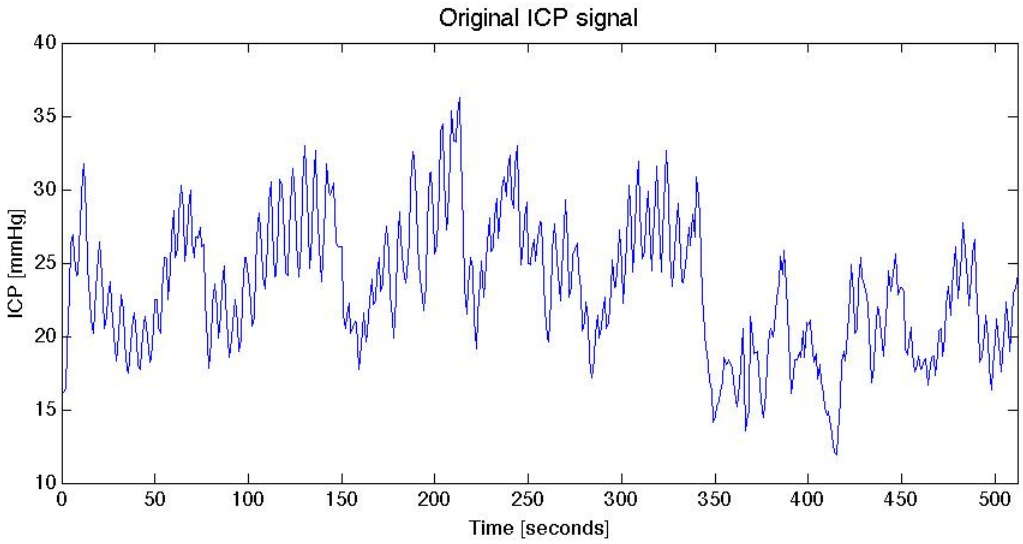


Figure 4.12: B wave before the filtering stage.

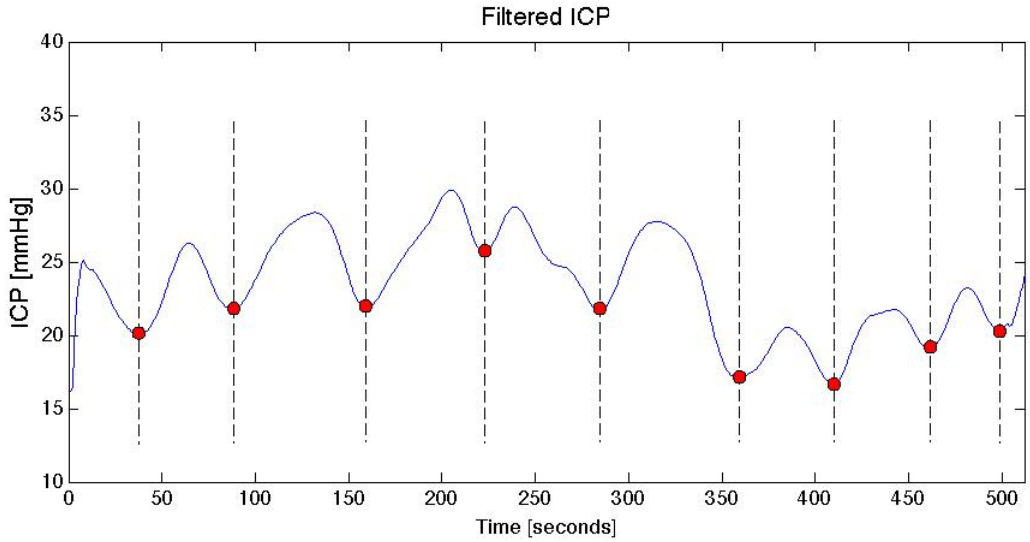


Figure 4.13: B wave after filtering stage and windowing process. Dashed lines divide the signal in windows.

4.1.4 Extracting features and judging

Extracting frequency and amplitude from every sub-signal is a very simple process:

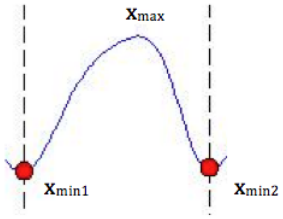


Figure 4.14: Signal from the window 3 in figure 4.13. Features extraction.

- Frequency: To estimate the frequency, the length of the window is inversed.

$$f = \frac{1}{window\_length} \tag{4.10}$$

- Amplitude: There are different ways to estimate the amplitude of the pulse. The algorithm first finds the maximum and then calculates the amplitude on this way:

$$A = \frac{(x_{max} - x_{min1}) + (x_{max} - x_{min2})}{2} \tag{4.11}$$

  Where

  - x_{max} is the maximum value of this sub-signal.
  - x_{min1} is the first notch before.
  - x_{min2} is the second notch after.

As it was described in the theoretical part, B waves are oscillations in the ICP with a period between 30 and 120 seconds (in frequency between 8 mHz and 33 mHz) and amplitude higher than 3 mmHg. Before start judging, a correction of the threshold is necessary since the filtering process attenuates the B waves. To succeed it, the approximation that describes the relation between frequency and attenuation is used.

$$\text{Threshold'} = \text{Threshold} \cdot (17208 \cdot f^3 - 1261 \cdot f^2 + 3 \cdot f + 1) \tag{4.12}$$

The original Threshold is 3 mmHg by definition.

This scheme shows how algorithm judges:

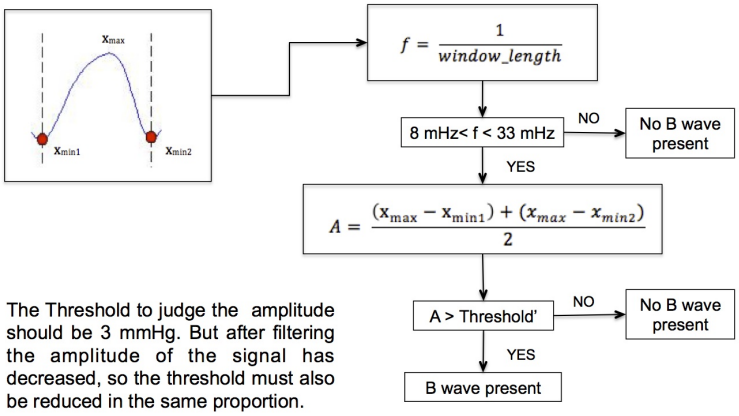


Figure 4.15: Procedure of judging every sub-signal

Following the same methodology as the doctor, number 1 indicates that a B wave is present and number 3 indicates that there is lack of B wave. The doctor has also the option to use the number 2 to express uncertainty and also the number 4 to express no judgement.

After every window has been analysed, a 256 samples length vector is built with the judgements of all the windows. The parts of the original signal that don't belong to any windows, are not judged by the algorithm and appear as a 0 in the vector of judgements.

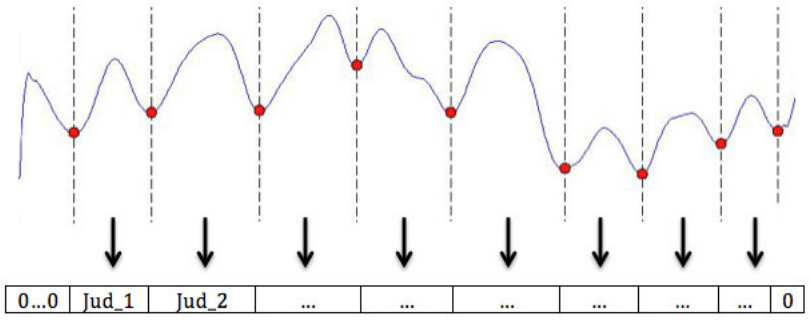


Figure 4.16: Judging process. Every window is judged and afterwards the judgement is stored in a vector. The judgements are 1 (for B wave present) or 3 (for lack of B wave).

In the next page there are some examples showing the results after analysing sets of 256 samples and comparing them with the doctor judgement.

Example 1:

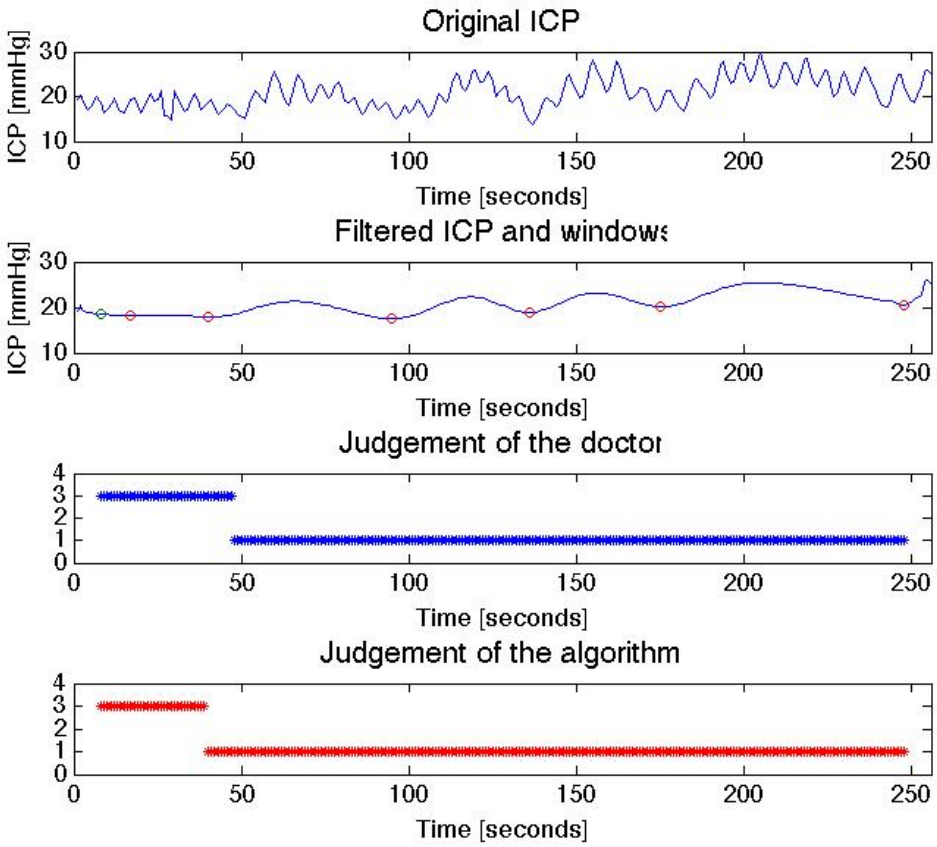


Figure 4.17: Comparison with the judgement of the doctor and the judgement of the algorithm. There's a small delay between the judgement of the doctor and the judgement of the algorithm.

COMMENT:

In this first example the doctor (blue) has judged all the time. According to the numeration, in the first part his judgement says that B-waves are not present. From approximately the 50th sample on, the doctor concludes that B-waves are present until the end. The algorithm (red) is judging almost in the same way with the doctor. There is a small delay in the judgement that will prevent the algorithm to get a 100% of accuracy although it is working nearly perfect. The small delay caused by the filtering stage (the filter is not perfectly symmetric so it introduces a delay in the output signal) shouldn't affect the final results because it is nearly 0 seconds. In this sample set the accuracy obtained by the algorithm was 96,6%.

Example 2:

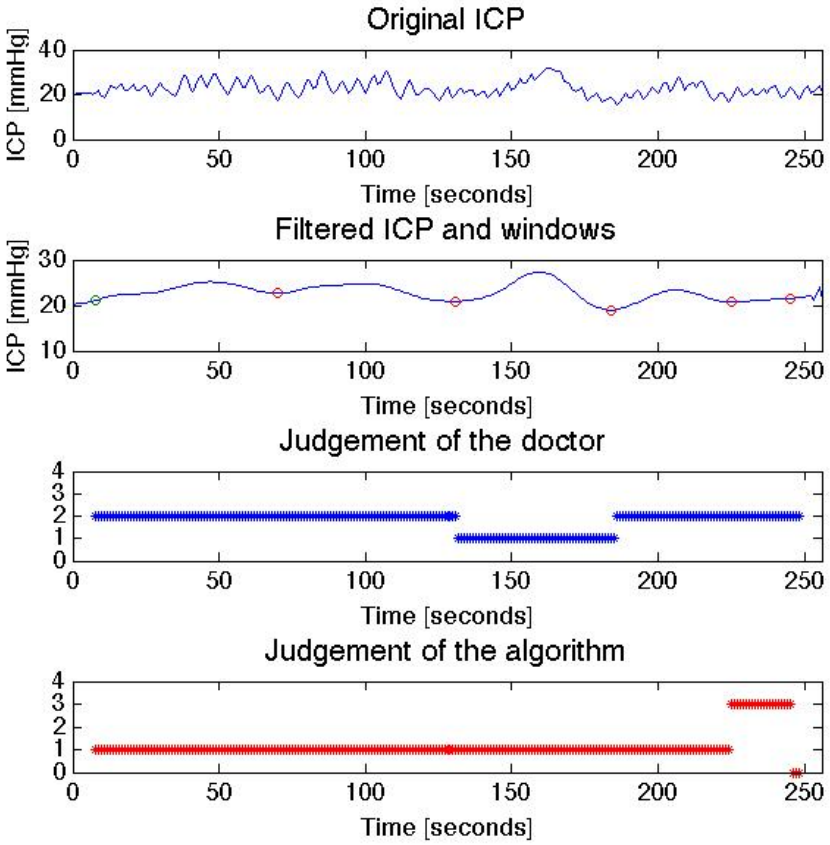


Figure 4.18: Comparison with the judgement of the doctor and the judgement of the algorithm. The algorithm cannot express ambiguity. Nevertheless when the doctor judges with 1, the algorithm does too.

COMMENT:

In this second example the doctor is only judging between samples 130th and 180th, concluding the presence of B-waves. During all the other period he expresses uncertainty.

Since the algorithm doesn't have the possibility to express uncertainty, it is judging all the time. The main point is that the algorithm is matching when doctor judges.

In this example, only the part where the Doctor is judging 1 would be used to calculate the accuracy, the sensitivity and the specificity. In this case we would obtain a 100% of accuracy since when the doctor is judging 1 (B waves present) the algorithm decision matches it all.

Example 3:

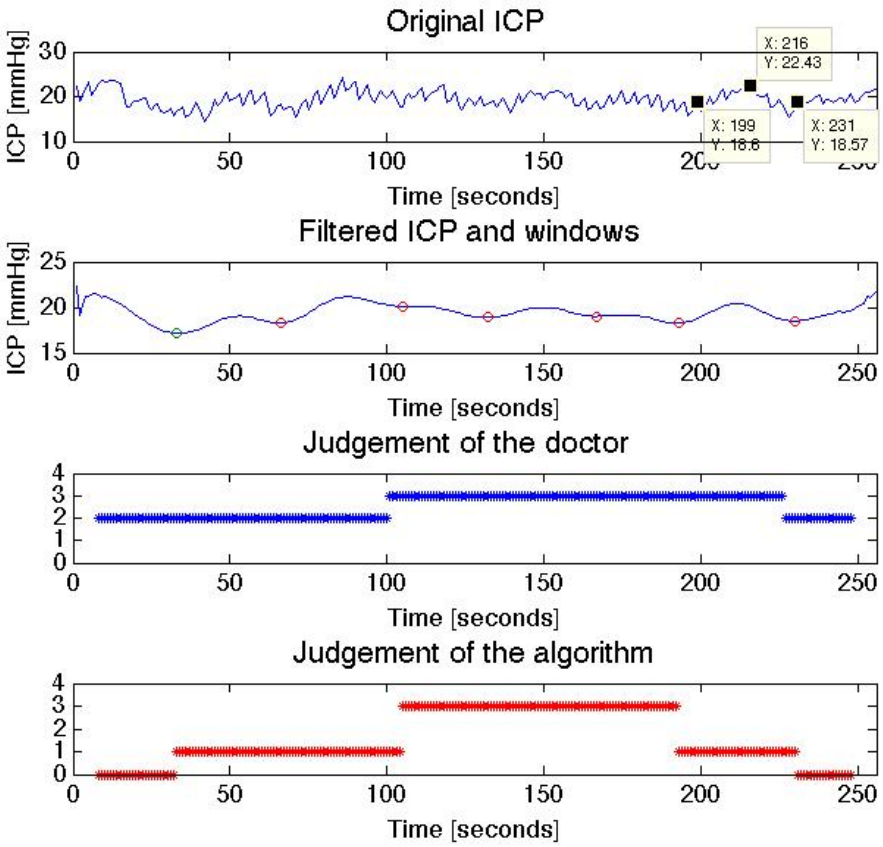


Figure 4.19: Comparison with the judgement of the doctor and the judgement of the algorithm. Oscillation corresponding with a B wave (yellow points in the Original ICP). Wrong judgement of the doctor.

COMMENT:

The relevant part of this example is between samples 190th and 230th approximately. In this range, there is a discrepancy between the judgement of the doctor and the judgement of the algorithm. As it can be observed, the amplitude of the original ICP is higher than 3 mmHg so it should be judged as 1 (B wave present) instead of 3 (lack of B wave).

In this case the algorithm performs a 69,84% of accuracy instead of 100%.

Mistakes in doctor's decision are also going to degrade the results of the algorithm.

4.1.5 Theoretical results

The following table contains the results obtained by the algorithm in every patient.

Patient	Accuracy	Specificity	Sensitivity
1	88,2	87,68	88,44
3	91,2	91,66	75,66
4	96,33	96,34	95,06
6	93,07	91,37	97,82
7	91,76	91,8	91,36
8	97,27	96,86	100
9	92,97	94,95	78,9
10	85,18	81,01	99,93
12	90,01	92,48	78,75
13	94,64	92,77	96,31
14	72,78	63,04	94,88
16	85,94	91,86	72,89
17	89,61	89,13	95,72
18	90,86	93,57	75,68
19	83,67	81,14	98,44
20	91,55	67,92	98,92
21	86,86	81,16	92,95
22	93,8	89,86	95,43
23	87,66	68,97	98,96
24	85,65	79,96	93,79
25	87,36	89,59	84,56
26	83,05	85,42	80,61

Table 4.1: Theoretical results obtained applying the algorithm based on B wave definition

SUMMARY:

	Accuracy	Specificity	Sensitivity
MEAN	89,06	86,30	90,23
DEV	5,56	9,22	8,89

Table 4.2: Summary of the theoretical results obtained by using the algorithm based on B wave definition

4.1.6 Adapting the code to the microcontroller

Due to the limitations of the microcontroller there are some parts of the code that must be adapted and, consequently, the results obtained by the theoretical model will change.

The main limitation that affects the system is that the microcontroller cannot work with floating point data. Fortunately, this problem can be easily solved by expanding the range of integer values. If data is rescaled, the threshold related with the amplitude must be rescaled too.

On the other hand, the amplitude threshold must also be corrected to compensate the attenuation that the filtering process has caused to the signal. In order to do this, a polynomial approximation of the filtering system has been calculated. The problem is that the input of the polynomial is the frequency, which requires floating precision.

$$\text{Threshold'} = \text{Threshold} \cdot (17208 \cdot f^3 - 1261 \cdot f^2 + 3 \cdot f + 1) \tag{4.13}$$

Since the period is an integer value, the response has been recalculated to use this parameter as input instead of the frequency. Then, the approximation will be like:

$$\text{Threshold'} = \text{Threshold} \cdot (A \cdot T^3 - B \cdot T^2 + C \cdot T + D) \tag{4.14}$$

In this case the variable T can be defined as integer. In contrast, the constants A = 0.00000039734, B = 0.0001707, C = 0.0242 and D = 0.1864 are floating but they don't need to be defined as variables in the program.

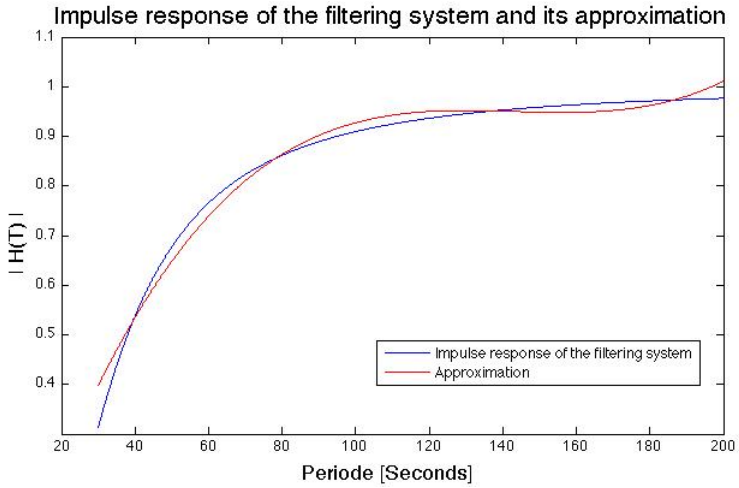


Figure 4.20: Approximation of the "Period Response" using a polynomial of order 3

4.1.7 Results after adapting the code for the microcontroller

Patient	Accuracy	Specificity	Sensitivity
1	87,05	86,72	87,21
2	93,14	21,29	94,34
3	90.56	90,67	86,98
4	96,30	96,31	95,00
5	90,84	91,19	88,16
6	92,96	92,25	94,91
7	89,74	90,11	85,14
8	97,09	96,71	100,00
9	93,04	94,66	81,03
10	83,96	82,31	89,91
11	94,41	95,10	44,18
12	89,61	92,20	77,12
13	94,31	93,62	94,96
14	74,71	66,08	94,51
15	94,07	93,99	94,93
16	87,23	91,35	77,16
17	90,28	89,87	95,92
18	90,46	93,43	73,41
19	85,76	83,46	99,89
20	92,69	71,76	98,43
21	86,38	82,64	90,58
22	91,42	92,76	90,83
23	85,61	73,03	93,18
24	85,39	81,67	90,81
25	86,77	90,02	82,63
26	81,42	84,62	77,99
27	89,53	82,8	90,46

Table 4.3: Results obtained from the algorithm based on B wave definition after being adapted to the microcontroller

SUMMARY:

	Accuracy	Specificity	Sensitivity
MEAN	89,59	89,71	89,16
DEV	6,04	7,71	9,44

Table 4.4: Summary of the results obtained by using the approximation of the algorithm based on B wave definition

Observing the two tables, it can be seen that the results achieved by the theoretical algorithm and the one adapted for the microcontroller are almost the same. It makes sense because the only different between two methods is the function that corrects the threshold value.

4.2 Algorithms based on Pulse wave morphology

4.2.1 Features extraction

Introduction

This method has been designed to extract different features from the intracranial pressure pulses (P-waves) with the less computational cost and, at the same time, with minimum number of errors. The algorithm requires data sampled at 100 Hz and length of 512 samples. For signals with a different length the system incorporates a pre-stage that divides the input signal in blocks of the required length.

- Windowing.
- P-wave analysis.
- Building a representative pulse.

Error handling is present throughout all the steps of the program, so when something unusual happens the program stops analysing the current P-wave and looks for another one.

The next block diagram gives a global idea about how the program proceeds:

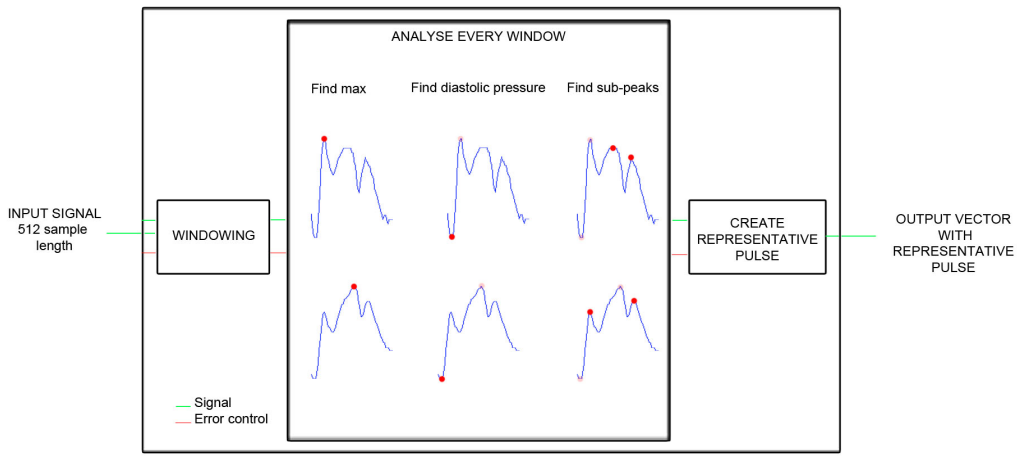


Figure 4.21: Features extraction system for ICP P waves using 512 samples.

Why using signals of 512 samples?

The purpose of the system is to estimate the morphology of the P-wave along the time. In order to get a consistent estimation, more than one pulse is required. In contrast, as long as the number of pulses used for the estimation increases, resolution in time gets lower. In a 512 samples set, 4, 5, or even 6 P-waves can be present depending on the heart rate. This number of waves is enough for a consistent estimation of the P-wave morphology. Moreover, 512 samples are equivalent to 5,12 seconds. One representative pulse every 5,12 seconds is enough resolution for our purpose. Finally, there are technical reasons that restrict the number of samples to use. Since it wants to be an online method, it's necessary to pay attention to the RAM memory.

Moreover than this algorithm, there will be many processes running in the microcontroller that will require using RAM memory. The microcontroller has 8 Kb to use as RAM and 512 samples what is equivalent to a 20% of the total memory.

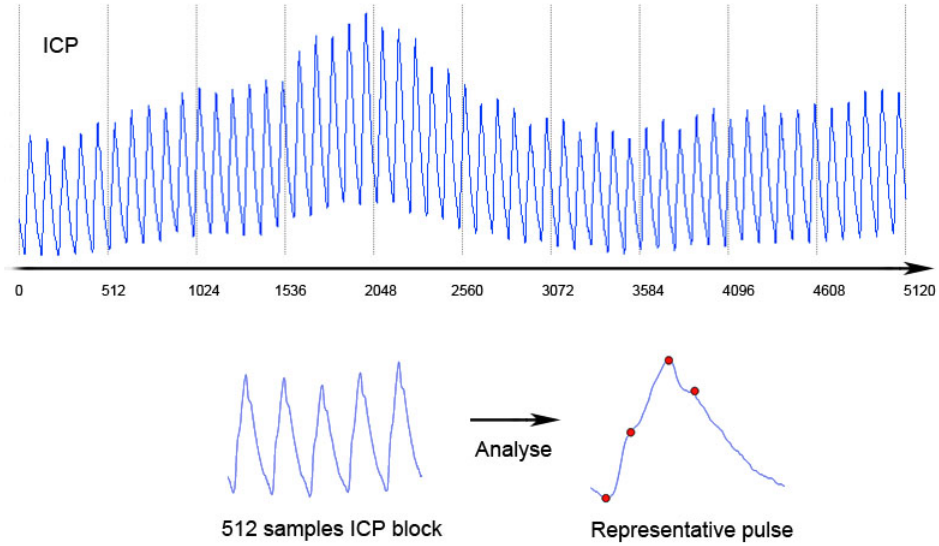


Figure 4.22: Division of ICP long register in blocs of 512 samples and extraction of the representative pulse.

Windowing blocks

Before extracting the representative pulse of one block, an analysis of the features of every pulse is required. So the first step is to separate the different pulses that are present in a block, procedure that is called windowing. The output of this procedure is a vector with the time references to the start and the end of every pulse. All this process can be divided in two main steps:

- Passing the signal through a strong filtering block that erases all the sub-peaks and the noise from the signal. The filtering block consists in two moving averages filters in a row, the first one with N=16 and the other with N=8.

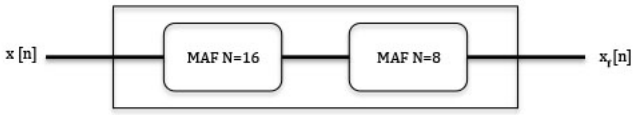


Figure 4.23: Filtering system compound of two MAF.

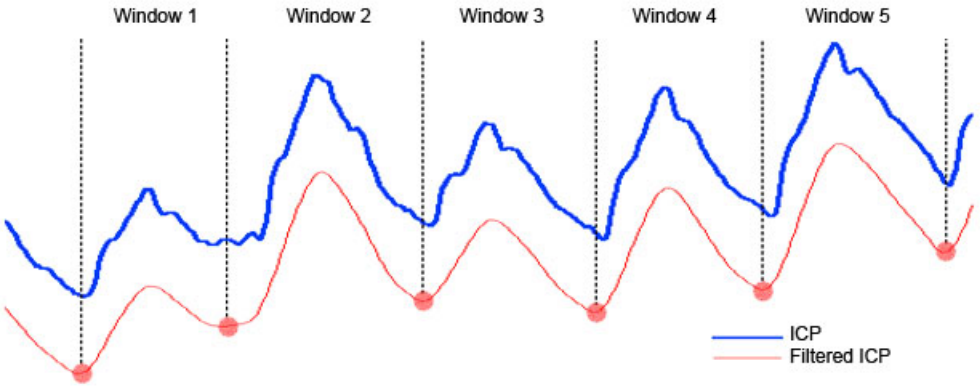


Figure 4.24: Windowing process. Dashed lines divide the signal in 5 windows.

- The picture 4.24 shows the signal before and after the filtering process. As it can be seen, every notch of the filtered signal corresponds closely with the start and the end of a pulse. With this information the vector that contain the windows is build.

In some cases, when the signal is too noisy or the sub-peaks of the pulses are too sharped, the filtering stage is not able to eliminate them. In this instance the algorithm will select windows with a wrong start or end and, consequently, the algorithm will select the pulse wrongly. Owning to this fact, detecting when an issue is present in the windowing process is very important to discard it on time and avoid wrong estimations.

As is shown in the picture 4.25, windows 2, 3, 5 and 6 are shorter than normal. In this occasion the issues are caused by the presence of very sharped sub-peaks in the pulses. If these false windows are transferred to the next stage they will distort the final result.

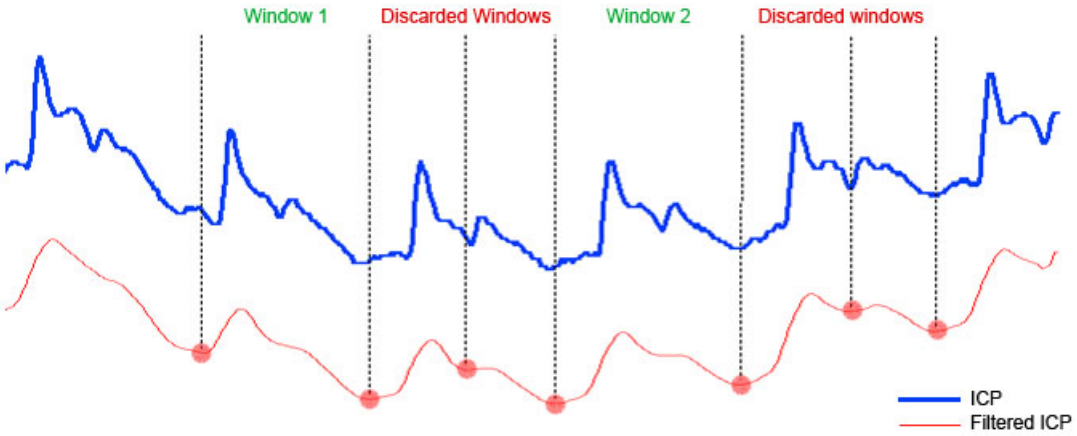


Figure 4.25: Example of wrong windows selection. The system is able to detect and discard them.

The length of the windows is exactly the time between heartbeats. This time is normally between 0,6 and 1 second in adults. So the criterion for select or discard a window is:

- If the window length is shorter than 64 samples (0,64 seconds, in order to use a power of 2), the window will be discarded
- Otherwise the window is selected.

P-wave analysis

Once the windows are found out the next step is to analyse every one separately. The features extraction procedure for a single pulse consists basically in 3 parts:

- Maximum detection.
- Diastolic pressure detection.
- Sub-peaks detection.

After this, features are stored in a vector called pressure vector with the following structure:

Pressure Vector window i

Diastolic pressure (D.P.)	-
Difference of pressure between peak 1 and D.P. (dP1)	Time distance from D.P. position until dP1 position
Difference of pressure between peak 2 and D.P. (dP2)	Time distance from D.P. position until dP2 position
Difference of pressure between peak 3 and D.P. (dP3)	Time distance from D.P. position until dP3 position

Table 4.5: Characteristics extracted by the algorithm from every pulse

For every window a vector like the one below is obtained. Once all pulses are analysed, the representative vector is calculated.

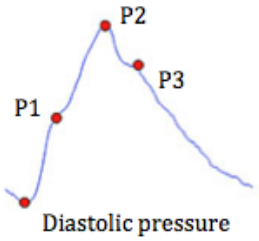


Figure 4.26: ICP values extracted from every ICP P wave.

Representative Vector

Diastolic pressure (D.P.)	-
Difference of pressure between peak 1 and D.P. (dP1)	Time distance from D.P. position until dP1 position
Difference of pressure between peak 2 and D.P. (dP2)	Time distance from D.P. position until dP2 position
Difference of pressure between peak 3 and D.P. (dP3)	Time distance from D.P. position until dP3 position

Table 4.6: Representative vector of the block of 512 samples

- D.P. = mean (D.P window 1, D.P. window 2, ..., D.P. window N)
- dP1 = mean (dP1 window 1, dP1 window 2, ..., dP1 window N)
- dP2 = mean (dP2 window 1, dP2 window 2, ..., dP2 window N)
- dP3 = mean (dP3 window 1, dP3 window 2, ..., dP3 window N)

Maximum detection:

In order to find the maximum of the pulse the program check all the samples of the P-wave. When the maximum value of ICP is detected, the program stores it. The error control is activated when the position of the maximum value is too close to the start or the end of the signal.

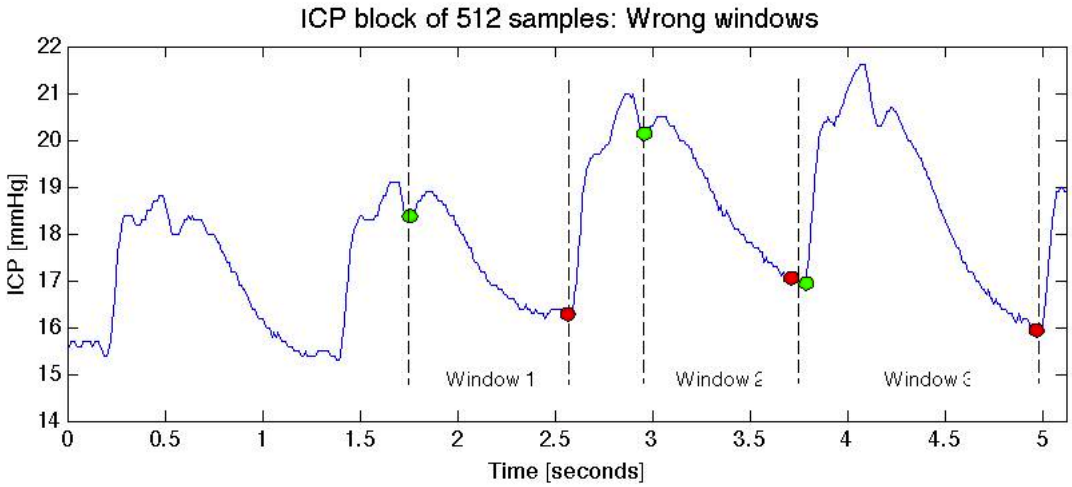


Figure 4.27: Example of two windows wrongly selected by the algorithm.

Windows 1 and 2 pass the first error control because their lengths are more than 64 samples, but they are obviously wrong. When they pass to the next stage, the signals inside the windows present these shapes:

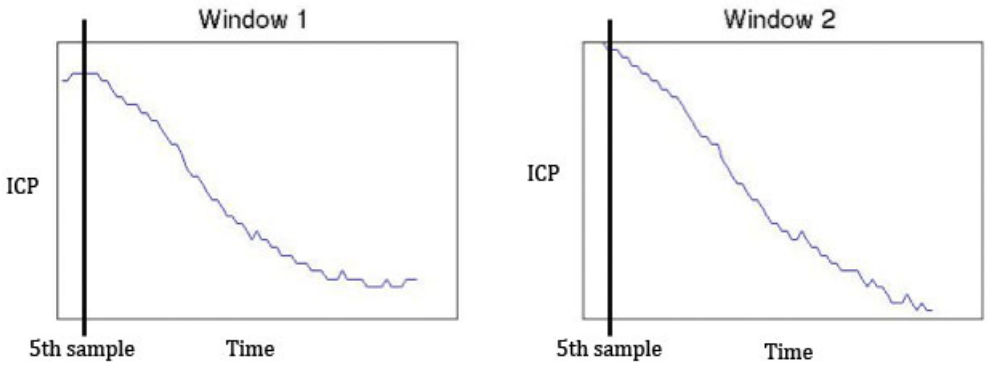


Figure 4.28: Signals inside two widows wrongly selected.

As it can be observed, the maximum is at the beginning of the window. After analysing a big amount of different cases, it has been decided to discard these windows where the maximum of its signal is located in the firsts or the lasts 5 samples.

Diastolic pressure detection:

Diastolic pressure is obtained when the heart is relaxed. A relative minimum is present in the intracranial pressure. After a successful windowing process, diastolic pressure is normally present in one of the firsts 20 samples of the P wave. In the remote case that the algorithm does not find the diastolic pressure in this range of values, the error control is activated.

Sub-peaks detection:

Detecting sub-peaks is a more complex task. The first point is to know if the maximum is P1 or P2 in order to know where to search the other sub-peaks. Depending on the case, the two sub-peaks will be placed just on the right of the maximum (P1=Max) or they will be placed one on the left and one on the right of the maximum (P2 = Max). It is shown in the picture below.

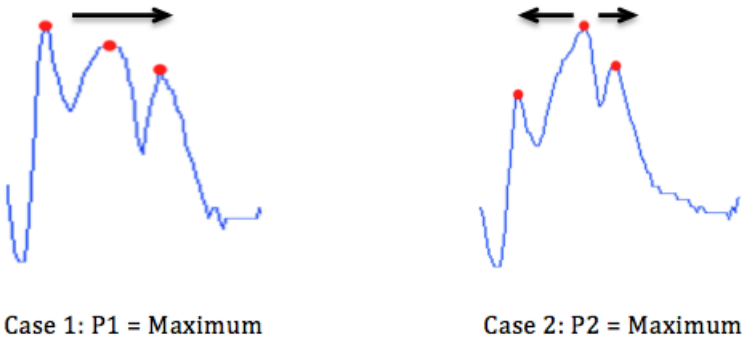


Figure 4.29: The algorithm performs two different analyses depending on two cases. In the first case (left) the maximum corresponds with P1. In the second, the maximum corresponds with P2 (right).

Finding cases:

There are several ways to discern between these 2 cases. The algorithm analyses the derivative of the ICP from diastolic pressure until the maximum pressure. When P1 corresponds with the maximum, the derivative of the ICP presents just one maximum (one turning point). In contrast, when maximum is in P2, the derivate contains two maximums and one minimum (three turning points). This is an easy and effective way to separate both cases.

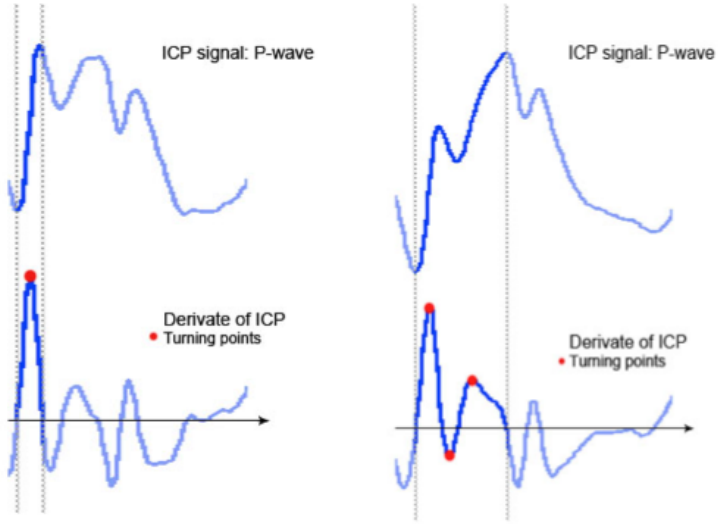


Figure 4.30: Illustrative example of how the algorithm detects the case using the derivative of the ICP. In case 1 (left) the algorithm finds 1 turning point (red) between diastolic pressure and the maximum. In the case two (right), the algorithm finds 3 turning points.

The program just needs to find the number of inflection points between diastolic pressure and the maximum. If just one turning point is found the algorithm decides case one. If three turning points are in the signal the algorithm decides case 2. Otherwise error control will be activated for this pulse. Figure 4.30 exemplifies this procedure.

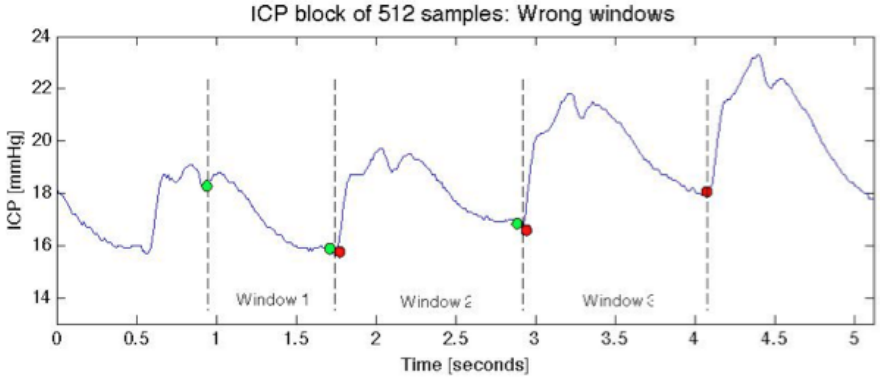


Figure 4.31: Example of wrong windowing without detection from the algorithm (window 1).

The window 1 goes through the first error control (window length ≥ 64) and the second error control (max position ≥ 5).

As we can see in the figure, there's not any turning point between the beginning and the maximum. Since the number of inflection points is different from 1 or 3, the system will discard this window.

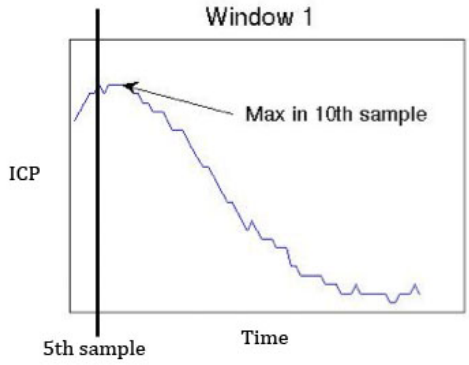


Figure 4.32: Example of how the error control detects that windowing process is wrong. The maximum of the signal is to close in time to the beginning of the window.

Finding sub-peaks:

In order to find the 2 sub-peaks the derivate of the ICP signal will be used again. Sub-peaks can present two different kinds of shape: turning point (case 1) of relative maximum (case 2). It is important that the algorithm can discern between both.

The algorithm follows two different rules depending if it is looking for a sub-peak on the left or on the right of the maximum.

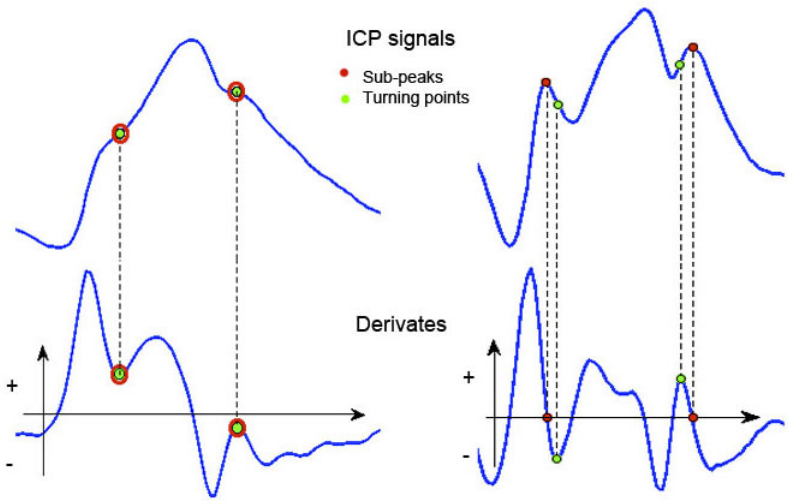


Figure 4.33: Illustration of how the algorithm finds P1 and P3 using the derivation. The algorithm difference two cases, when sub-peaks are turning points (left) or maximums (right).

- To find a sub-peak on the left, the algorithm search the first relative minimum on the derivate departing from the position of the maximum of the ICP signal. If the found minimum is positive means that it is case 1, so this points is already the sub-peak. In contrast, a negative value means that it is case 2, so it is necessary to go further

in the derivate until this one changes its sign.

- To find a sub-peak on the right, the algorithm search the first relative maximum on the derivate departing from the position of the maximum of the ICP signal. If the found maximum is negative means that it is case 1, so this point is already the sub-peak. In contrast, a positive value means that it is case 2, so it is necessary to go further in the derivate until this one changes its sign, like before.

If an unusual P-wave is present there's the possibility of not detecting any sub-peak, as in the example in the picture below. In this case the error control is activated and the algorithm stops analysing this pulse and starts analysing a new one.

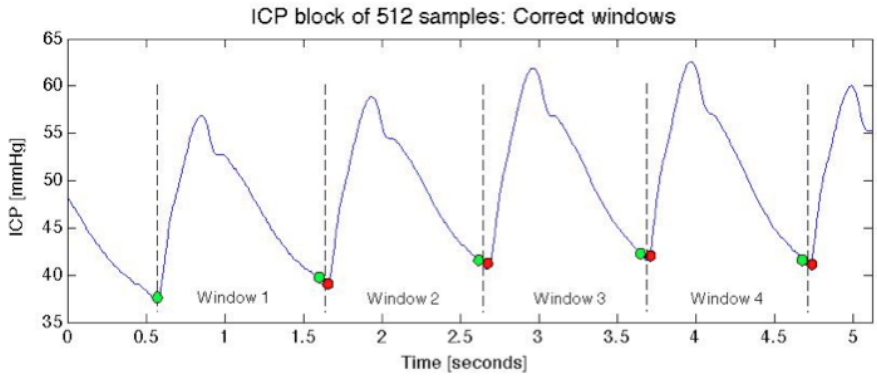


Figure 4.34: Example of non-typical ICP pulse morphology. P1 is too soft and the algorithm isn't able to detect it.

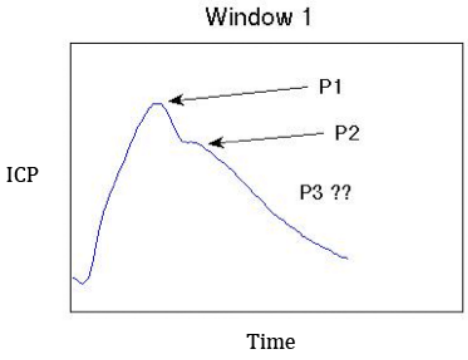


Figure 4.35: The algorithm doesn't detect P1 and associates the maximum to P1 when should be P2. Afterwards the algorithm is not able to detect P3 and discards the signal.

Since there are no more turning points after P2, the algorithm is not able to detect P3. In this case the window is discarded.

4.2.2 Implementing the database

In order to build the database, ICP records from 8 patients with Normal Hydrocephalus have been used. The original information was stored in two different kinds of files:

- ICP files: Theses files contain the ICP acquired at 100 Hz. The data is organized in columns, where every row represents a temporal instant.
- BWArzt files: These files are formed by two columns. In the first on there is the same ICP with the ICP file but converted to 1 Hz, which was displayed to the doctor for judgement. In the second column there is the judgment of a doctor for every instant of time.

Therefore, to build the database, the characteristics will be acquired from the ICP files and the judgements from the BWArzt file. It will be important to have in mind those 100 rows in the ICP file corresponds with just 1 in the BWArzt file.

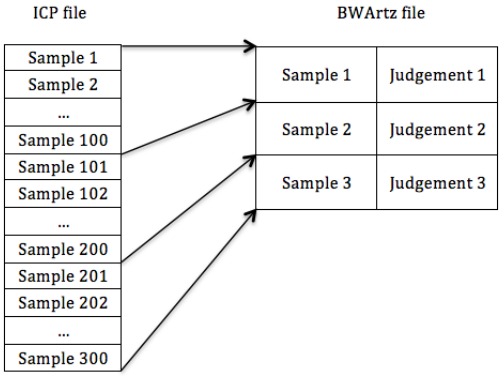


Figure 4.36: Representation of the files used to check the algorithms. Files are compound by one column for the data and another column for the judgement of the doctor.

The different features are extracted from the ICP signal every 41 seconds. As it has been described in the section 4.2.1, the features extractor gives as output a representative pulse every 5,12 seconds. Consequently, if the input signal is 41 seconds long, the output will be 8 representative pulses in a row.

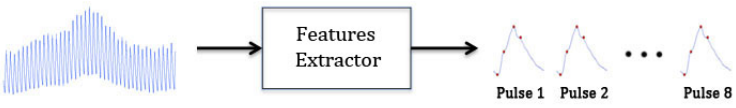


Figure 4.37: System schedule to obtain the 8 pulses that will be used to obtain the features for the classifier.

These 8 pulses are used to calculate the following features:

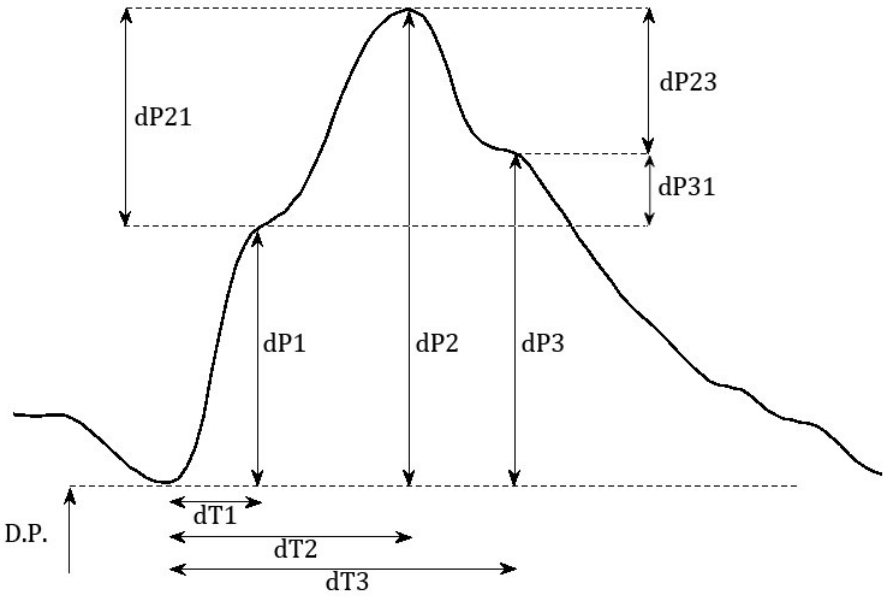


Figure 4.38: ICP P-wave with the features extracted by the system.

1. Mean value of diastolic pressure (D.P.).
2. Variance of D.P.
3. Mean value of the pressure in peak 1(dP1).
4. Variance of dP1.
5. Mean value of the pressure in peak 2 (dP2).
6. Variance of dP2.
7. Mean value of the pressure in peak 3 (dP3).
8. Variance of dP3.
9. Variance of the difference of pressure between P2 and P1 (dP21).
10. Variance of the difference of pressure between P2 and P3 (dP23).
11. Variance of the difference of pressure between P3 and P1 (dP31).
12. Time distance from diastolic pressure until P1 (dT1).
13. Variance of dT1.
14. Time distance from diastolic pressure until P2(dT2).

15. Variance of dT2.
16. Time distance from diastolic pressure until P3(dT3).
17. Variance of dT3.
18. Trend of D.P ($trend\_DP$).
19. Trend of P1 ($trend\_P1$).
20. Trend of P2 ($trend\_P2$).
21. Trend of P3 ($trend\_P3$).

The last three features $trend\_DP$, $trend\_P1$, $trend\_P2$, $trend\_P3$ express wether the ICP is increasing or decreasing in these 41 seconds.

After the features have been extracted the values are stored in a row vector following the same order with the list in the last page. The judgement of the doctor for these 41 seconds is included in the element 22 of the vector.

$vector_1^T$	Judgement 1
$vector_2^T$	Judgement 2
...	...
$vector_N^T$	Judgement N

Table 4.7: Structure of the database

After all the data available from 8 patients have been analysed, 2300 datasets have been obtained. From these 2300 datasets, 1150 contain B waves present and 1150 do not. Therefore, the dimensions of the matrix are $21 \cdot 2300$. This collection of datasets is not ready to train a learning machine system. It will be treated following the steps described in the section 3.3.2.

First of all the outliners, values that are to far from the mean value, must be erased because they can drive the learning process to a distorted classifier. The criterion used is erasing the datasets that are 2σ far from the mean value. After eliminating the outliners, the number of datasets has been reduced to 1430, 566 for B waves and 864 for lack of B waves.

Secondly the next relation between the number of datasets and the dimension of every data set is checked.

$$N \approx 2^P \tag{4.15}$$

Where d is the dimension of every dataset and N is the number of datasets. In this case the values are:

- N = 1430
- $2^P = 2^{21} = 2097152$

Obviously, $2^P \gg 1430$. According to this, there is the possibility that the amount of datasets is insufficient to describe the space and therefore it will lead to lower accuracy of the classifier. In the next sections, the two methods chosen to reduce the number of features are explained.

Finally, the only step left before start testing the first prototypes is to normalize the database in order to give the same weight to every feature. It has been chosen the linear normalization because the calculation cost is lower:

$$s_i = \frac{v_i - min(v_1...v_N)}{max(v_1...v_N) - min(v_1...v_N)} \tag{4.16}$$

4.2.3 Features analysis and classifier selection

Many of the decisions taken in the process of designing a solution for a given classification problem are based on different aspects of the features. In fact, the success or fail of the system will be strongly linked to this analysis.

The analysis of the features has basically two purposes:

- Decide which features are more suitable to be used as inputs of the classifier.
- Decide which classifier seems to approach better the problem and consequently which classifier will reach a higher accuracy.

Regarding to the first question, not all features provide different information to the system. It is possible that two features present a strong correlation and therefore one of them is superfluous. In order to detect which features are redundant, the covariance matrix has been calculated (Appendix 2) using the following expression.

$$C(\mathbf{x}, \mathbf{y}) = \frac{1}{N} \sum_{n=1}^{N} (x_n - \bar{x})(y_n - \bar{y}) \tag{4.17}$$

$\mathbf{x}$ and $\mathbf{y}$ are vectors of 1430 elements that represent two different features. Note that all data has been transformed in order to obtain a normalized matrix. The transformation is described by the next 3 equations:

$$x_i = \frac{v_i - \overline{v}}{\sigma} \tag{4.18}$$

$$\overline{v} = \frac{1}{N}\sum_{i=1}^{N} v_i \tag{4.19}$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(v_i - \overline{v})^2}{N-1}} \tag{4.20}$$

The Covariance matrix must be interpreted as follows:

- C(x,y)>0: There's a direct relation between the variables.
- C(x,y)=0: There's no lineal relation between the variables.
- C(x,y)<0: There's a inverse relation between the variables.

Observing the matrix in detail (Appendix 2), the next conclusions are obtained:

- The mean of dP3, the variance of dP3, the variance of dP21 and the variance of dP31 (features number 7, 8, 9 and 11) have a strong correlation. Therefore, in case of necessity of eliminate features, these three can be discarded.

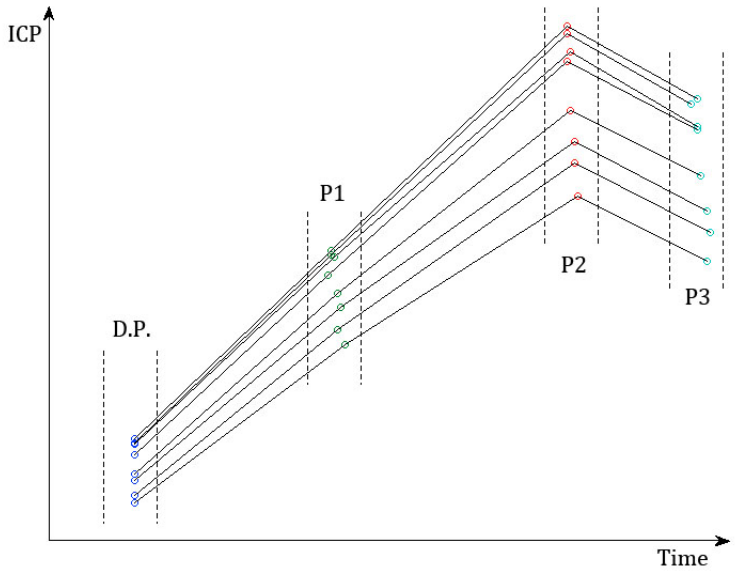


Figure 4.39: Representation of the features of 8 pulses extracted by the system in 41 seconds.

As it can be observed in the figure 4.39, the lines from D.P. to P1 and the lines from P1 to P2 have different slope. In contrast, the slopes in lines from P2 to P3 are very similar. These parallel lines appear in a big amount of different datasets and it justifies why the covariances near to 1 are obtained.

- The six characteristics that regard to time aspects of the P wave are also closely related. Moreover, none recent study has suggested that the presence of B waves alters the duration of the pulses. Consequently, these six characteristics could be discarded too.

- Finally, the three last characteristics that indicate if the amplitude of the peaks trend to increase or decrease present a high correlation. Hence, only one from the three will be used.

The next step is to determine which classifiers are more suitable for the available data. In this project, the selection of a classifier consists basically in finding out wether the probability distribution functions of the features are Gaussians or not. The techniques and parameters used to study the Gaussianity of the data are the Skewness (Appendix 1), the Kurtosis (Appendix 1) and the Histograms (Appendix 3).

By looking at the histograms, it can be guessed that none of the features follow a Gaussian distribution. Supporting this, it can also be observed that most of values from the table that contains the results of skewness and kurtosis of the distributions are far from zero, fact that suggest non-Gaussianity.

Given that the probability distribution functions of the data are unknown and according to the theory explained in section 3.3, parametric classifiers are not suitable to solve this problem. By looking at the histograms it can be observed that classes are significantly overlapped and consequently powerful classifiers will be required. Multilayer Perceptrons and Support Vector Machines are classifiers able to solve classification problems with complex boundaries between classes and for this reason they have been chosen despite they consume an important amount of resources from the system.

4.2.4 Training and testing the classifiers

In this section, the accuracies, sensitivities and specificities of a Multilayer Perceptron and a Support Vector Machine classifier have been computed and compared. Classifiers have been trained, validated and tested using different combinations of input features from the database. Three different cases have been studied.

- All features used.

- A reduced number of features selected by looking at the covariance matrix. The features selected are 2, 3, 4, 5, 6, 10 and 19.
- A reduced number of features by the technic PCA. The number of features has been reduced to 7 in order to compare the results with the second case.

In order to obtain different results from the classifiers and thus to be able to compute its mean and its variance, the function of Matlab crossvalind has been used. The whole database has been divided in ten subsets. Then, 8 subsets have been used for training (80%), one for validation (10%) and one for test (10%), following the scheme below.

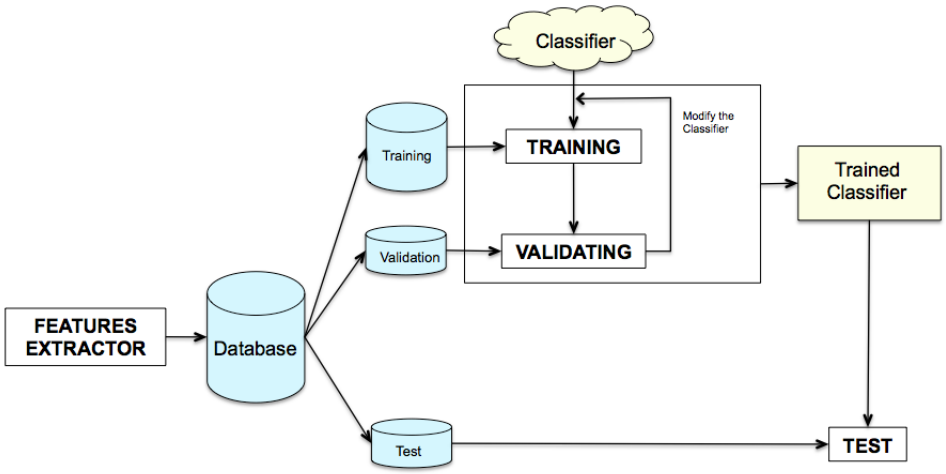


Figure 4.40: Scheme used to train, validate and test the classifiers

- First of all, the classifier is trained using different structures and the error of validation is computed for each structure. In the case of the Multilayer Perceptron, the number of neurons in the hidden layer is scanned from 1 to 20 in steps of 1. For support vector machines, the value of sigma is scanned from 0,2 to 2 in steps of 0,2.
- The structure that obtained best accuracy with the validation subset is selected and the classifier is trained again with this structure.
- Finally, the already trained classifier is tested with the testing subset. The accuracy, sensitivity and specificity from this last test are stored.

This process is repeated ten times using different subsets for the training, the validation and the test and storing the accuracy, sensitivity and specificity reached for every one. Finally, the mean and the variance of these three parameters are shown as a result.

In the following tables the results reached by the classifiers are shown. Table 4.8 contains the results obtained when all features have been used. In a single row, the datasets used

for training are the same in both classifiers and the same happens with the datasets used for validation and for test. Thanks to this, the classifiers can be compared also row by row and not only at the end. The results that regard to the MLP have been obtained by using one hidden layer.

MLP				SVM			
Number of Neurons	Accuracy	Sensitivity	Specificity	Sigma	Accuracy	Sensitivity	Specificity
13	81,82	70,18	89,53	0,6	89,51	82,46	94,19
5	80,42	71,93	86,05	0,4	83,22	82,46	83,72
6	85,92	80,36	89,53	0,4	88,73	87,50	89,53
4	82,52	73,21	88,51	0,2	79,72	82,14	78,16
2	85,31	70,18	95,35	0,2	89,51	85,96	91,86
1	83,33	75,44	88,51	0,4	86,81	78,95	91„95
14	87,41	82,46	90,70	0,4	88,81	87,72	89,53
13	86,62	82,14	89,53	0,4	86,62	83,93	88,37
6	84,62	80,36	87,36	0,4	88,11	80,36	93,10
1	82,64	68,42	91,95	0,4	82,64	85,96	80,46
MEAN	84,06	75,47	88,70		86,37	83,74	88,09
STD. DEV.	2.26	5.42	2.57		3.37	2.98	5.49

Table 4.8: Results obtained by the two classifiers when all features have been used.

There are some observations made:

- When all features are used to train, validate and test the classifiers, SVM reaches higher accuracies but with a bit more variance than MLP.
- The results of sensitivities and specificities with SVM are more similar than with MLP, a fact that makes the SVM a better classifier. Anyway, none of the results is critical, since the specificity in both cases is around 90% so there will be a small number of false detections of B wave.

After testing the results it has also been tried a Multilayer Perceptron with two hidden layers. Table 4.9 show the results obtained by the MLP using two hidden layers and all the features. The datasets used for training, validating and testing are the same used in table 4.8.

As it can be observed there is no improvement in the results obtained by the multilayer perceptron with two hidden. Results are almost identical and for this reason, in case of using a multilayer perceptron, it seems reasonable to use only one hidden layer since it reduces the number of operations and the number of variables in memory.

Table 4.10 have the results obtained when features 2, 3, 4, 5, 6, 10 and 19 are used. These features have been selected by looking at the Covariance Matrix. Feature 1 has

MLP with two hidden layers			
Structure	Accuracy	Sensitivity	Specificity
[6 5]	83,93	77,19	88,37
[17 10]	81,82	81,58	64,53
[15 18]	82,39	76,79	86,05
[8 15]	85,31	78,57	89,66
[14 10]	86,01	73,68	94,19
[15 4]	83,33	71,94	90,80
[10 18]	86,01	77,19	91,86
[7 7]	85,92	80,36	89,53
[18 4]	84,62	78,57	88,52
[12 2]	79,86	66,67	88,51
MEAN	83,92	76,25	87,20
STD. DEV.	2.07	4.41	8.26

Table 4.9: Results obtained by a MLP using two hidden layers.

been discarded because it is obtained using real values of ICP, which cannot be obtained by the sensor from iShunt. Feature 18 has been discarded because didn't provide any improvement to the results of the classifiers. After comparing the results of the multilayer perceptron (with one hidden layer) with the results of the SVM it can be seen that SVM still reaches a higher accuracy in mean. Even though, the SVM has suffered a more significant reduction in the accuracies than the MLP. Regarding to the variance, it has increased in both classifiers specially in the sensitivity of the SVM where it has raised from 2,98 to 7,66.

MLP				SVM			
Number of Neurons	Accuracy	Sensitivity	Specificity	Sigma	Accuracy	Sensitivity	Specificity
7	86,71	75,44	94,19	0,4	86,71	73,68	95,35
17	81,82	70,18	89,53	0,2	81,82	80,70	82,56
9	83,10	66,07	94,19	1	80,99	60,71	94,19
15	80,42	73,21	85,06	0,2	79,72	73,21	83,91
8	83,92	70,18	93,02	0,2	86,01	77,19	91,86
12	84,72	73,68	91,95	0,2	84,03	78,95	87,36
1	83,22	71,93	90,70	0,2	86,01	78,95	90,70
18	83,10	75,00	88,37	0,2	86,62	80,36	90,70
9	80,42	67,86	88,51	0,2	84,62	75,00	90,80
17	77,08	59,65	88,51	0,8	78,47	59,65	90,80
MEAN	82,45	70,32	90,40		83,50	73,84	89,82
STD. DEV.	2.67	4,81	2.95		3.03	7,66	4,09

Table 4.10: Results obtained by the two classifiers when features 2, 3, 4, 5, 6, 10 and 19 have been used.

Finally, it has been tried a less drastic method to reduce the number of features called PCA (table 4.11). This technic uses the eigenvectors of the covariance matrix associated to the

MLP				SVM			
Number of Neurons	Accuracy	Sensitivity	Specificity	Sigma	Accuracy	Sensitivity	Specificity
3	75,52	59,65	86,06	0,2	89,51	84,21	93,02
14	82,52	77,19	86,05	0,2	81,12	75,44	84,88
11	82,39	75,00	87,21	1	82,39	75,00	87,21
16	75,52	62,50	83,91	0,2	79,72	78,57	80,46
5	81,12	63,16	93,02	0,6	84,62	68,42	95,35
4	81,94	66,67	91,95	1,2	83,33	68,42	93,10
13	82,52	75,44	87,21	0,4	83,22	82,46	83,72
15	80,99	78,57	82,56	0,4	83,80	78,57	87,21
6	79,72	64,29	89,66	0,4	85,31	75,00	91,95
20	78,47	61,40	89,66	0,2	79,86	70,18	86,21
MEAN	80,07	68,39	87,73		83,29	75,63	88,31
STD. DEV.	2.72	7,31	3,34		2,88	5,49	4,82

Table 4.11: Results obtained by the two classifiers when the number of features have been reduced using PCA.

highest eigenvalues to project the original vectors into a new base with less components. The number of eigenvectors chosen to make the projections are 7 which same number of features used in the case before (2, 3, 4, 5, 6, 10 and 19) so results can be compared. As it can be seen, SVM reaches again higher values than MLP. Using PCA with 7 dimensions hasn't improved the results respect using the seven features 2, 3, 4, 5, 6, 10 and 19.

4.2.5 Adapting the Classifiers to the system

As it has been explained in chapter 3, the system uses an ultra-low-energy microcontroller and there are some requirements that firmware must meet. There are three main limitations that affect the algorithms.

Relative pressure acquired. The pressure acquisition system of iShunt is giving the relative value of the pressure as output. In last section, Diastolic Pressure has been used as a feature to train and test the MLP. This feature is calculated in absolute value and it cannot be obtained in the real system. After eliminating this feature results may worsen.

The microcontroller cannot include mathematic libraries. Inside each classifier there are mathematical functions that are essential for the proper work of the classifier. These functions cannot be used because the microcontroller cannot include mathematic libraries, but they can be approximated trying minimize the consequences.

The microcontroller works with integer data. This is the worst limitation for the algorithm because, by definition, MLP and SVM are structures that work with real data and thus floating-point precision is required. Moreover, the system has been designed to normalize

input data from 0 to 1, so data must be normalized again choosing new boundaries.

Multilayer Perceptron

The second condition commented in the previous lines is that the microcontroller cannot include mathematic libraries. In the case of Multilayer Perceptrons, a specific kind of functions called transfer functions are part of the classifier and they are need for the proper performance of the classifier.

More specifically, the classifier uses a sigmoid function. This function is defined as:

$$f(x) = \frac{1}{1 + e^{-x}} \tag{4.21}$$

Sigmoid has been approximated by a Heaviside step function.

$$f'(x) = \begin{cases} 1 & \text{if } x < 0 \\ 0 & \text{if } x \geq 0 \end{cases} \tag{4.22}$$

In order to solve the third limitation, the range of integers used has been expanded multiplying every weight and bias by the same constant. It doesn't change the result because the output of every neuron depends linearly with its inputs and the sigmoid function has been approximated by a Heaviside step function:

Starting from the basic structure of a Perceptron:

$$y = step(\sum_i x_i w_i + b) \tag{4.23}$$

If weights and bias are multiplied by a constant K:

$$y' = step(\sum_i x_i K w_i + K b) \tag{4.24}$$

$$y' = step(K(\sum_i x_i w_i + b)) \tag{4.25}$$

$$y' = step(\sum_i x_i w_i + b) \tag{4.26}$$

If the output of every neuron remains unchanged, the final result will not be altered either. Therefore this is a simple solution that solves effectively the problem of the data type. The value chosen for K is 1000.

Input data has also been expanded. At the beginning input data was normalized from 0 to 1 but due to the limitations of the system, this range has been expanded from 0 to 1000.

The value of K and the new range of the input data haven't been randomly chosen. After many proofs it has been observed that lower values for the constants degrade the final results because variables haven't resolution enough. In contrast, using higher values is dangerous because it can cause overflow of the variables in internal operations. So it is strongly recommended to use $K = 1000$ and the database normalized from 0 to 1000.

In order to check the agravation of the results when the approximation is used, a theoretical MLP with one hidden layer has been trained, validated and tested and then the approximation has been tested using the parameters of the first classifier and the same datasets of the test. This process has been repeated ten times:

	Theoretical MLP			Approximation of MLP			
Number of Neurons	Accuracy	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity	Δ Accuracies
7	86,71	75,44	94,19	76,92	61,40	87,21	9,79
17	81,82	70,18	89,53	76,92	73,68	79,03	4,89
9	83,10	66,07	94,19	84,51	73,21	92,86	-1,40
15	80,42	73,21	85,06	73,43	64,29	79,31	6,99
8	83,92	70,18	93,02	73,43	78,95	69,77	10,48
12	84,72	73,68	91,95	82,64	68,42	91,95	2,08
1	83,22	71,93	90,70	74,83	78,95	72,09	8,39
18	83,10	75,00	88,37	84,51	76,79	89,53	-1,4
9	80,42	67,86	88,51	79,02	73,21	82,76	1,39
17	77,08	59,65	88,51	74,31	63,16	81,61	2,77
MEAN	82,45	70,32	90,40	78,05	71,21	82,52	4,4
STD. DEV.	2.67	4,81	2.95	4,41	6,51	7,75	4,39

Table 4.12: Comparison of the results obtained by the MLP with one hidden layer using features 2, 3, 4, 5, 6, 10 and 19

As it can be observed, using the approximation have reduced the accuracy around a 4% and the values of the variances have raised.

Support Vector Machines

Similary to the MLP, the SVM must be approximated due to the same facts: mathematical functions and data types. In the case of the SVM the procedure is a bit more complex.

Starting with the mathematical functions the SVM include a function called kernel, which is responsible for changing the initial space to another where classes are easier to separate. The chosen kernel is a Gaussian Kernel, also called Radial Basis Function, and follows this expression:

$$K(\mathbf{x}_n, \mathbf{x}') = e^{-\frac{\|\mathbf{x}_n - \mathbf{x}\|^2}{2\sigma^2}} \tag{4.27}$$

The approximation of the kernel is done by the composition of two operations:

- The first one corresponds with the square of the norm:

$$n_2(\mathbf{x}_n, \mathbf{x}') = \| \mathbf{x}_n - \mathbf{x}' \|^2 = \sum_{i=1}^{N} (\mathbf{x}_n[i] - \mathbf{x}'[i])^2 \tag{4.28}$$

  Depending on how is normalized the input data x, the range of values in the output will change. The range of data chosen for input data is [0, 100].

- The second operation corresponds with the approximation of the exponential. In fact, the operation of the norm is not any approximation so the approximation of the kernel depends basically in how the exponential is approached. Given an approximation of the exponential, it is demonstrated that this structure emulates the kernel:

  Starting from the definition of the Gaussian kernel,

$$K(\mathbf{x}_n, \mathbf{x}') = e^{-\frac{n_2(\mathbf{x}_n, \mathbf{x}')}{2\sigma^2}} \tag{4.29}$$

  If

$$e(n) = e^{-\frac{n}{2\sigma^2}} \tag{4.30}$$

  The Kernel can be expressed as the composition of function n_2 and function e:

$$K(\mathbf{x}_n, \mathbf{x}') = e(n_2(\mathbf{x}_n, \mathbf{x}')) = e^{-\frac{n_2(\mathbf{x}_n, \mathbf{x}')}{2\sigma^2}} = e^{-\frac{\|\mathbf{x}_n - \mathbf{x}\|^2}{2\sigma^2}} \tag{4.31}$$

  So approximating the kernel is reduced to approximate e(n).

  In a first approach, polynomials of grade 1 and 2 have been used to approximate the exponential. The domain of the function has been defined from 0 to 10000 and it has been divided in 8 parts, everyone fitted by a different polynomial. In this first approximation, the chosen sigma has been 25.

$$e(n) = e^{-\frac{n}{2 \cdot 25^2}} \approx \sum_{k=1}^{8} p_k(n) \cdot t(n_{sk}, n_{ek}) \tag{4.32}$$

$$\hat{e}(n) = \sum_{k=1}^{8} p_k(n) \cdot t(n_{sk}, n_{ek}) \tag{4.33}$$

Where

$$t(n_{sk}, n_{ek}) = \begin{cases} 1 & n_{sk} < n < n_{ek} \\ 0 & others \end{cases} \tag{4.34}$$

and

$$p_k(n) \text{ are polynomials of orders 1 or 2} \tag{4.35}$$

With this, the kernel can be approximated by:

$$K(\mathbf{x}_n, \mathbf{x}') \approx \hat{e}(n_2(\mathbf{x}_n, \mathbf{x}')) \tag{4.36}$$

Using the following expression for the exponential.

$$\hat{e}(n) = \begin{cases} (-65,7582n + 9886010)10^{-5} & n < 500 \\ (-44,079n + 88307)10^{-5} & 500 \leq n < 1000 \\ (9,7485 \cdot 10^{-4}n^2 - 53,7287n + 88775)10^{-5} & 1000 \leq n < 2000 \\ (3,0379 \cdot 10^{-4}n^2 - 25,9603n + 59598)10^{-5} & 2000 \leq n < 4000 \\ (6,1335) \cdot 10^{-4}n^2 - 7,6947n + 24969)10^{-5} & 4000 \leq n < 6000 \\ (3,15187 \cdot 10^{-1}n + 2616)10^{-5} & 6000 \leq n < 8000 \\ (-6,3635 \cdot 10^{-2}n + 655)10^{-5} & 8000 \leq n < 10000 \\ 0 & 10000 \leq n \end{cases} \tag{4.37}$$

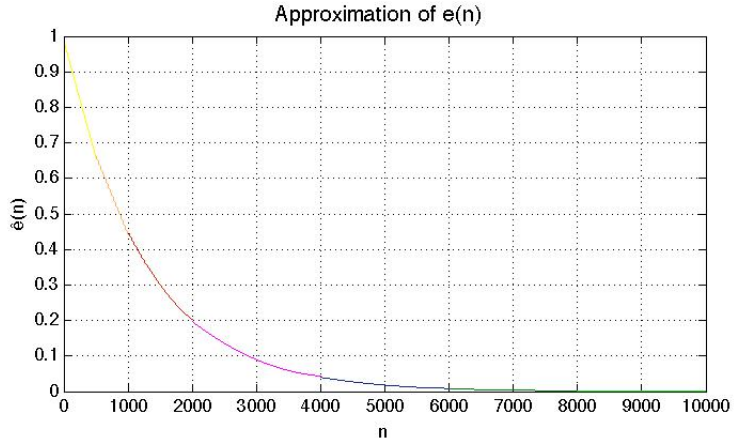


Figure 4.41: Approximation of the exponential using 7 polinomials.

After differents proofs, it was observed that the results obtained were rarely worse than they should be. When the theoretical SVM obtained results around 85% of accuracy, the approximation of the kernel obtained around a 70%. Analyzing the internal working of the algorithm it was noticed that due to higher values of $n_2(\mathbf{x}_n, \mathbf{x}')$ sometimes there was overflow in the variable $\hat{e}(n_2)$ and accordingly the classification was wrong.

Furthermore to this issue, another problem arises from using this kind of approximation. Every time that sigma changes a new approximation of the exponential has to be build, fact that is a big deal when several sigmas have to be checked.

Finally, the problem has been solved by creating a vector that contains the values of the exponential from 0 to 10000. This solution gets over with problem of overflow and it is easier to be build if a scan of sigma is required. Another benefit is that in case exceeding the memory resources of the system, the number of cells can be easily reduced by interpolating the values.

In parallel, the third limitation has been also solved extending the range of integer values, multiplying the parameters of the classifier by constants. According to this, the program will do the following:

1. Acquiere support vectors, alphas and b. This point needs to extend the values of all of them. It has been commented before that the input data range has been extended from 0 to 100 and this includes the support vectors. The range of alphas and b must be also expanded since normally they get values around 1.
2. Calculate $n_2(\mathbf{x}_n, \mathbf{x}')$. The outputs of the norm are big so an extension of range has not been necessary in this step.
3. Calculate the approximation of $K(\mathbf{x}_n, \mathbf{x}')$ by using $\hat{e}(n)$. The kernel needs a range expansion because its range goes from 0 to 1.
4. Product of alphas and $K(\mathbf{x}_n, \mathbf{x}')$ and addition of b. This operation does not need any expansion.

According to this, 3 constants are necessary: C_α for the alphas, C_k for the approximation of the kernel and C_b. All of them are bigger than 0.

Now the approximation of the classifier is:

$$\widehat{g}(\mathbf{x}') = sign(\sum_{n=1}^{N} C_\alpha \alpha_n y_n C_k \hat{e} + C_b b) \tag{4.38}$$

In order to influence the minimum in the classifier, the relation between constants must be:

$$C_\alpha \cdot C_k = C_b \tag{4.39}$$

Demonstration

The approximation of the total classifier is:

$$\widehat{g}(\mathbf{x}') = sign(\sum_{n=1}^{N} C_\alpha \alpha_n y_n C_k \hat{\mathrm{e}} + C_b b) \tag{4.40}$$

if $C_\alpha \cdot C_k = C_b$

$$\widehat{g}(\mathbf{x}') = sign(\sum_{n=1}^{N} C_b \alpha_n y_n \hat{\mathrm{e}} + C_b b) \tag{4.41}$$

$$\widehat{g}(\mathbf{x}') = sign(\sum_{n=1}^{N} \alpha_n y_n \hat{\mathrm{e}} + b) \tag{4.42}$$

And $\hat{\mathrm{e}}(n_2) \approx K(\mathbf{x}_n, \mathbf{x}')$

$$\widehat{g}(\mathbf{x}') \approx sign(\sum_{n=1}^{N} \alpha_n y_n K(\mathbf{x}_n, \mathbf{x}') + b) = g(\mathbf{x}') \tag{4.43}$$

Finally,

$$\widehat{g}(\mathbf{x}') \approx g(\mathbf{x}') \tag{4.44}$$

By looking at the following table, the results obtained by the theoretical SVM and its approximation can be compared. The proper values for the constants are: $C_\alpha = 1000$, $C_k = 100$ and $C_b = 100000$.

Theoretical SVM				Approximation of SVM				
Sigma	Accuracy	Sensitivity	Specificity	Sigma	Accuracy	Sensitivity	Specificity	Δ Accuracies
0,4	86,71	73,68	95,35	40	83,22	77,19	87,21	3,50
0,2	81,82	80,70	82,56	20	80,42	82,46	79,07	1,4
1	80,99	60,71	94,19	100	82,39	62,50	95,35	-1,41
0,2	79,72	73,21	83,91	20	80,42	76,79	82,76	-0,70
0,2	86,01	77,19	91,86	20	86,71	80,70	90,70	-0,70
0,2	84,03	78,95	87,36	20	83,33	80,70	90,70	0,69
0,2	86,01	78,95	90,70	20	89,51	85,96	91,86	-3,50
0,2	86,62	80,36	90,70	20	86,62	82,14	89,53	0,00
0,2	84,62	75,00	90,80	20	85,31	78,57	89,66	-0,70
0,8	78,47	59,65	90,80	80	77,08	59,65	88,51	1,39
MEAN	83,50	73,84	89,82		83,50	76,67	87,97	0,00
STD. DEV.	3,03	7,66	4,09		3,66	8,67	4,69	1,89

Table 4.13: Comparing results obtained by the SVM using features 2, 3, 4, 5, 6, 10 and 19

4.3 Summary of results

In the next lines, the results of the three algorithms are summed up and compared. Note that the data used to test the algorithm based on frequency and amplitude detection is different than the data used to test the algorithms based on P wave's morphology.

Every cell contain the results expressed as: mean(%) / variance.

Method 1 Hz

	Accuracy	Sensitivity	Specificity
Theoretical	89,59 / 6,04	89,16 / 9,44	89,71 / 7,71
Approximation	89,06 / 5,56	90,23 / 8,89	86,30 / 9,22

Table 4.14: Summary of the results obtained by using the approximation of the algorithm based in B wave definition

Method 100 Hz: Theoretical

The results have shown that adding more layers to MLP doesn't improve the generalization of the classifier. For this reason it is preferable to work with only one layer. The results in both tables refer to MLPs with one hidden layer.

Classifier	Number of features	Reduction technic	Accuracy	Sensitivity	Specificity
MLP	21 (all)	-	84,06 / 2,26	75,47 / 5,42	88,07 / 2,57
	7	Covariance Matrix	82,45 / 2,67	70,32 / 4,81	90,40 / 2,95
	7	PCA	80,07 / 2,72	68,39 / 7,31	87,73 / 3,34
SVM	21 (all)	-	86,37 / 3,37	83,74 / 2,98	88,09 / 5,49
	7	Covariance Matrix	83,50 / 3,03	73,84 / 7,66	89,82 / 4,09
	7	PCA	83,29 / 2,88	75,63 / 5,49	88,31 / 4,82

Table 4.15: Summary of the results obtained by the MLP with 1 hidden layer and by the SVM

Method 100 Hz: Approximation

To test the approximations it has been used the case of 7 features reduced by the covariance matrix. The approximation of the MLP uses one hidden layer.

Classifier	Accuracy	Sensitivity	Specificity	Difference with theoretical
MLP	78,05 / 4,41	71,21 / 6,51	82,62 / 7,75	4,4 / 4,39
SVM	83,50 / 3,66	76,67 / 8,67	87,97 / 4,69	0 / 1,89

Table 4.16: Summary of the results obtained by using the approximations of MLP and SVM

Comparing the results obtained by the 3 methods it is clear that the best is the one based on Lundberg's definition of B wave. This method not only achieves the best results in accuracy, specificity and sensitivity but also it is extremely simpler than the other two, fact that makes it ideal for running on iShunt. It is also much better than all the 1 Hz methods from the state of the art [27], which reached accuracies around 70% (a 20 % less).

Regarding to the methods based on P wave morphology, the SVM classifier has obtained better results than the MLP classifier in all the situations. Comparing the results with the state of the art, the SVM classifier with the 21 features has achieved an accuracy of 86,37%, a very close value to the accuracy reached by the most successful study in the field until today [30], which is 88,9%. Moreover, the approximation of the SVM seems to be more effective than the one from the MLP. The main problem of the SVM is the amount of memory resources that needs to work. The best configuration for SVM would be using features 2, 3, 4, 5, 6, 10 and 19 and trying to reduce as much as possible the length of the vector that implements the Gaussian kernel.

A first approach for the global system could be a combination SVM and the method based on Lundberg's definition. The method of 1 Hz could work as the main B wave detector and method of SVM could be activated only when 1 Hz detected B waves in order to confirm the decision taken by the first method.

4.4 Global system

The global system must be able to acquire data from the patient, analyse it and proceed according to this analysis. Therefore, the system is divided in the following blocks:

- Data acquisition system + Pre-processing stage.
- Virtual ICP Monitoring system.
- Shunting system.

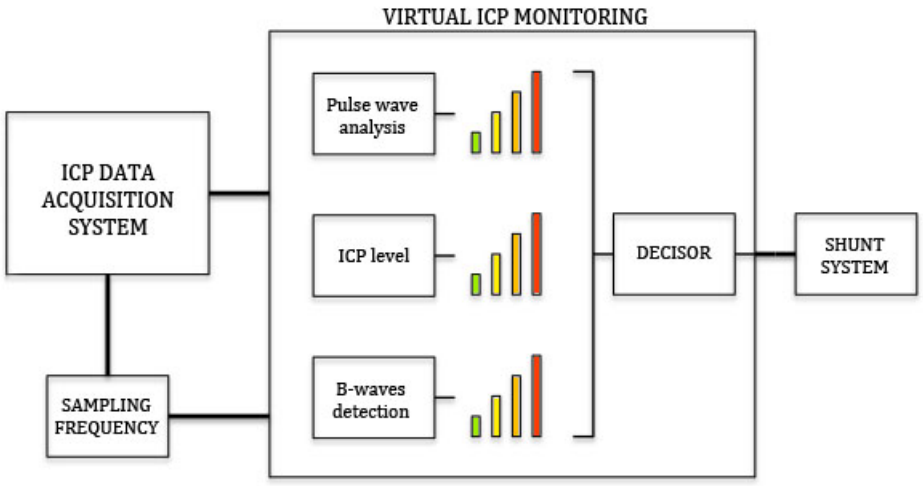


Figure 4.42: Blocks diagram of the global system

Although the Virtual ICP Monitoring System is the most innovative and representative part of iShunt, all the blocks have important differences with the conventional shunting systems.

The Virtual ICP Monitoring System will be divided in different sub-systems which function will be to rate numerically about one or different factors that influence in the shunting decision. The possibility of operation of the sub-systems will depend on some parameters as for example the sampling rate of the acquisition system.

Rates of every sub-system will be send to a common block that will analyse every rate and will decide how the drainage should be carried out. Finally, this decision will be send to the drainage system.

According to the results obtained by the algorithms for B waves detections, the block in charge will be able to work in two modes.

- Sampling frequency of data acquisition system is 1 Hz.
- Sampling frequency of the data acquisition system is 100 Hz.

In the first case, the algorithm of amplitude and frequency that works following the definition of B-wave will be used. This algorithm will output the percentage of B-waves present in long records. According to the guidelines developed in Japan [5], a high incidence of B waves (more than 15%) in long records (12h-48h) indicates a more efficient drainage. Therefore, a first proposal for the operation of the system is:

- When incidence of B wave is less than 10%. The block will not take part in the final decision.
- Incidence of B waves from 10% to 15%. It provides little information about shunting efficiency. It shows instability of the intracranial pressure so suggest that the other blocks in charge to analyse the ICP should start working as well.
- B waves present more than 15% of total records. Moreover than the information provided in the second point, it suggest a high efficiency probability in case of shunting.

In second case, when sampling frequency is 100 Hz, the classifier based in the morphology of P waves can work as well. This algorithm could take advantage of the data extracted by the block in charge to analyse the P waves. Since data at 1Hz will be acquired more often, the decision of the classifier that works at 100 Hz could be used to check the correct operation of the system that works at 1 Hz.

5 Conclusions

Referring to 1 Hz method

The first and the most important conclusion is that not taking into account the shape of the wave improves the detection. It has been said in section 3.2 that detectors proved by now use methods like spectrum estimators or wavelets that are too related with concrete waveforms. The method designed in this thesis assumes that if there is any periodical signal that its frequency and amplitude correspond with B waves definition, the shape will also correspond. This change in the point of view has drastically improved the results, going from a 70% [27] of accuracy in old methods, until almost a 90% of accuracy in this method.

Another important point is that this method minimizes the number of samples per block to be judged. So this method reaches the highest precision achievable in 1 Hz detection.

A further advantage is that the algorithm is universal because it strictly follows the definition of B wave. Thresholds to classify signals haven't been statistically calculated, but they have been extracted from Lundberg's definition. Therefore, this method is valid for every person.

Moreover this method is highly suitable to be implemented in a low energy system because it requires low temporal precision (it works with data sampled at 1 Hz), doesn't need to perform a high numbers of operations and doesn't need to store parameters in the system for its operation. Furthermore the changes made to adapt the algorithm to the microcontroller doesn't affect to the efficiency and therefore, the accuracy in iShunt should be around 90% as well.

As a drawback, the standard deviation of the results depending on the patient is still a bit high (std. deviation = 6). In the case that this algorithm was implemented in iShunt system it would be important to work in reducing this variance.

In this task of B waves detection it is important not to have false trues because of overdrainage. The results show that specificity (related with false trues) and sensitivity (related with false false) are pretty the same. In case it would be necessary to reduce the number of false trues, the threshold referred to the amplitude can be easily changed. For example it could be multiplied by a constant near to 1.1 or 1.2 (always higher than one if the purpose is reducing false trues).

Another interesting thing of this algorithm is that, in case that the distinction between sinusoidal shape and sawtooth shape was necessary it could be easily performed by looking

at the time position of the maximum in every period of the wave.

Referring to 100 Hz methods:

One more time it has been demonstrated that there's a valuable amount of information stored in the morphology of the P waves. In this case this information has been used to detect B waves by two different classifiers, MLP and SVM.

These two classifiers require a vector of characteristics of the P waves. To obtain these vectors, 100 Hz data is passed through a features extractor block that analyses the P waves. This procedure is costly but it will be included in iShunt because P waves analysis is essential to determine the condition of the patient. So if a features extraction block will be included in iShunt this information can be used to detect the presence or absence of B waves.

In this thesis a features extractor block has been implemented and its efficiency has been visually checked. For future projects it would be useful to implement a database that included pulse wave characteristics to check more precisely the operation of this or others features extraction blocks.

The main operation used to perform the features extraction has been the derivation. It is a good option to have in mind for future designs of features extraction blocks

After analysing the features extracted from the ICP, it has been observed that data doesn't follow a Gaussian distribution. This is a fact to have in mind for future projects in order to choose classifiers not based in Bayesian theory.

A big difference with the method of 1 Hz is that the method of 100 Hz judges blocks of 42 seconds. This reduces de precision of the algorithm.

Regarding to the technic for reducing the number of features, PCA hasn't improved the results of MLP and SVM. The performances of the two classifiers were slightly better when features 2, 3, 4, 5, 6, 10 and 19 were directly used.

MLP classifier:

MLP classifier is the one that has achieved the worst results. The best result obtained by this classifier has been an accuracy of 84% using all the features. When the number of features has been reduced from 21 to 7, the accuracy has decreased around a 2%.

An interesting fact is that, in this case, the generalization of the classifier hasn't improved when the number of hidden layers has increased.

Regarding to the approximation for the microcontroller, MLP is the method that has worsened more its results. The difference of accuracies between the theoretical model and

the approximation has been around 4%.

SVM classifier:

SVM has achieved better results than MLP. Referring to theoretical results, the improvement towards MLP is around 2%. Moreover, after approximating the methods, SVM keeps operating in the same values of accuracy, sensitivity and specificity.

The main problem of SVM to be implemented in iShunt is the amount of memory resources that it requires to work. The results obtained in this thesis has been obtained with SVM working with more or less 1000 support vectors and in the case that SVM was included in iShunt, all this data should be included in the memory of the system. The best option would be using features 2, 3, 4, 5, 6, 10 and 19 (without PCA reduction), in which case the total amount of memory required by the algorithm would be around 100 KBytes.

In case that the final system has enough availability of memory and computational capacity SVM would be clearly a better choice than MLP.

A Appendix

A.1 Skewness and Kurtosis

	B wave present		Lack of B wave	
Feature	Skewness	Kurtosis	Skewness	Kurtosis
1	0,388	0,570	2,394	2,228
2	2,452	3,399	11,452	20,836
3	0,540	0,998	2,142	3,447
4	1,260	3,239	4,442	18,123
5	0,579	1,126	2,598	3,552
6	1,313	3,088	4,584	15,743
7	0,897	1,328	2,998	4,5809
8	1,360	3,333	4,557	19,310
9	1,692	4,116	5,392	24,465
10	1,827	3,484	6,946	19,965
11	1,878	4,699	6,464	31,780
12	0,392	0,326	2,150	2,890
13	1,839	2,069	6,088	7,588
14	0,806	1,165	2,897	5,600
15	1,908	1,691	6,153	5,371
16	-0,374	-0,543	3,122	2,942
17	1,892	1,670	5,991	5,090
18	-0,038	-0,030	2,542	5,020
19	-0,035	-0,079	2,631	4,576
20	-0,036	-0,160	2,495	4,527
21	-0,042	-0,095	2,468	4,478

Table A.1: Skewness and Kurtosis computed for the 21 features when B waves are present and when there is lack of B waves

A.2 Covariance Matrix

Covariance Matrix	mean D.P.	var D.P.	mean dP1	var dP1	mean dP2	var dP2	mean dP3	var dP21	var dP23	var dP31	mean dT1	var dT1	mean dT2	var dT2	mean dT3	var dT3	trend DP	trend P1	trend P2	trend P3
mean D.P.	1,00	0,12	-0,07	-0,02	-0,24	-0,04	-0,19	-0,03	-0,03	-0,02	-0,27	-0,19	-0,05	-0,20	0,19	-0,23	0,06	0,05	0,02	0,02
var D.P.	-	1,00	0,08	0,46	0,07	0,43	0,07	0,39	0,27	0,39	-0,09	0,36	-0,06	0,44	-0,23	0,43	0,14	0,12	0,08	0,07
mean dP1	-	-	1,00	0,15	0,76	0,31	0,72	0,29	0,31	0,09	0,27	-0,03	-0,35	0,24	-0,29	-0,14	-0,31	0,11	0,10	0,09	0,09
var dP1	-	-	-	1,00	0,13	0,62	0,11	0,55	0,47	0,45	0,41	0,02	0,39	0,00	0,43	-0,20	0,37	0,05	0,08	0,06	0,04
mean dP2	-	-	-	-	1,00	0,49	0,99	0,48	0,50	-0,01	0,46	0,49	-0,20	0,71	-0,16	0,24	-0,18	0,08	0,08	0,10	0,09
var dP2	-	-	-	-	-	1,00	0,49	0,98	0,97	0,34	0,91	0,27	0,21	0,37	0,27	0,08	0,18	0,10	0,10	0,09	0,07
mean dP3	-	-	-	-	-	-	1,00	0,48	0,50	-0,03	0,46	0,51	-0,21	0,74	-0,17	0,28	-0,19	0,09	0,09	0,10	0,10
var dP3	-	-	-	-	-	-	-	1,00	0,97	0,29	0,96	0,28	0,19	0,38	0,26	0,11	0,18	0,10	0,10	0,08	0,07
var dP21	-	-	-	-	-	-	-	-	1,00	0,28	0,97	0,28	0,14	0,40	0,19	0,13	0,11	0,11	0,11	0,09	0,08
var dP23	-	-	-	-	-	-	-	-	-	1,00	0,27	-0,11	0,19	-0,08	0,22	-0,12	0,26	0,03	0,04	0,02	0,01
var dP31	-	-	-	-	-	-	-	-	-	-	1,00	0,28	0,12	0,39	0,19	0,15	0,13	0,11	0,11	0,08	0,07
mean dT1	-	-	-	-	-	-	-	-	-	-	-	1,00	-0,05	0,85	-0,07	0,75	-0,10	0,01	0,02	0,05	0,05
var dT1	-	-	-	-	-	-	-	-	-	-	-	-	1,00	-0,21	0,86	-0,34	0,84	-0,07	-0,05	-0,05	-0,06
mean dT2	-	-	-	-	-	-	-	-	-	-	-	-	-	1,00	-0,19	0,80	-0,23	0,04	0,05	0,08	0,08
var dT2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1,00	-0,34	0,93	-0,06	-0,05	-0,07	-0,08
mean dT3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1,00	-0,37	0,03	0,04	0,07	0,07
var dT3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1,00	-0,08	-0,06	-0,07	-0,08
trend DP	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1,00	0,98	0,93	0,93
trend P1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1,00	0,95	0,95
trend P2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1,00	0,99
trend P3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1,00

COV(x,y) ≤ 0,1
0,1 ≤ COV(x,y) < 0,3
0,3 ≤ COV(x,y) < 0,7
0,7 ≤ COV(x,y) < 1

A.3 Histograms

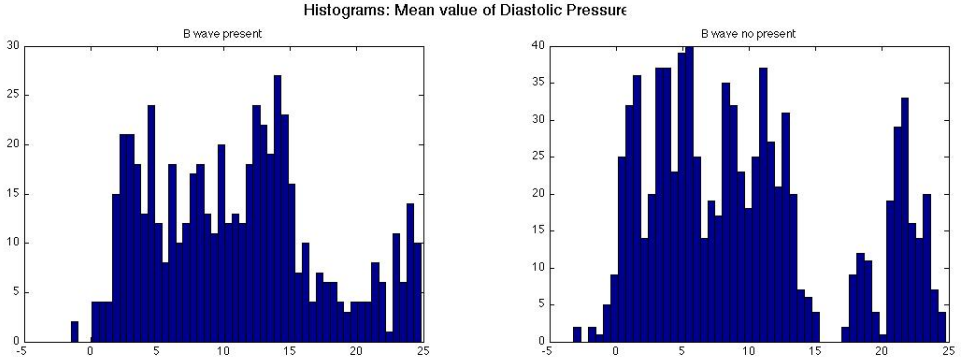


Figure A.1: Histograms of feature 1.

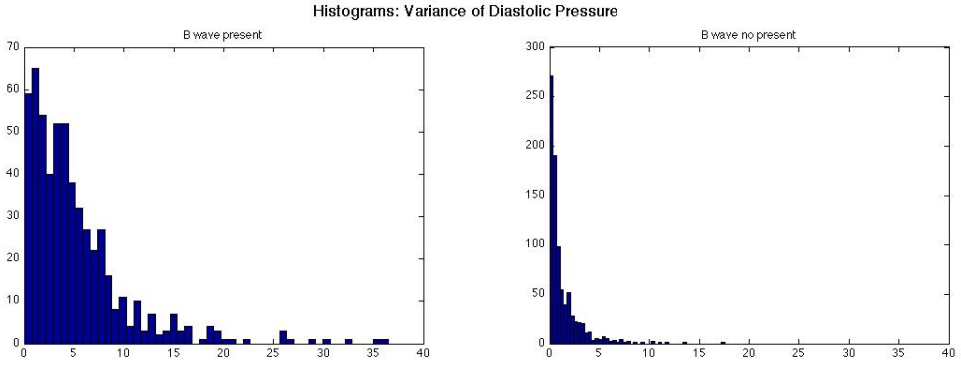


Figure A.2: Histograms of feature 2.

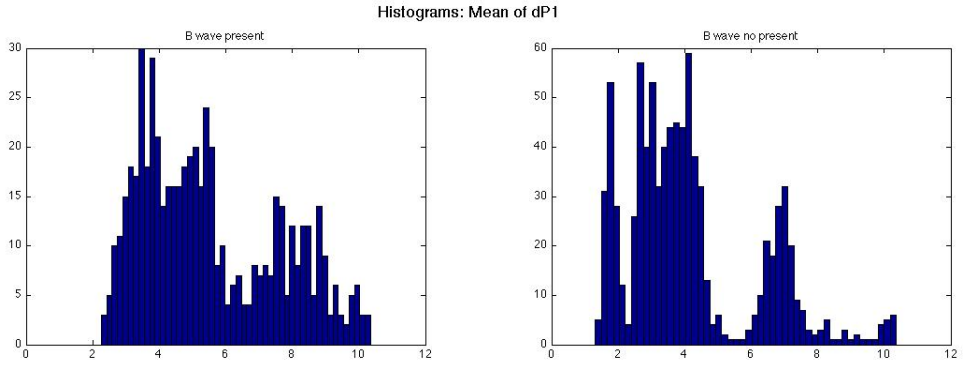


Figure A.3: Histograms of feature 3.

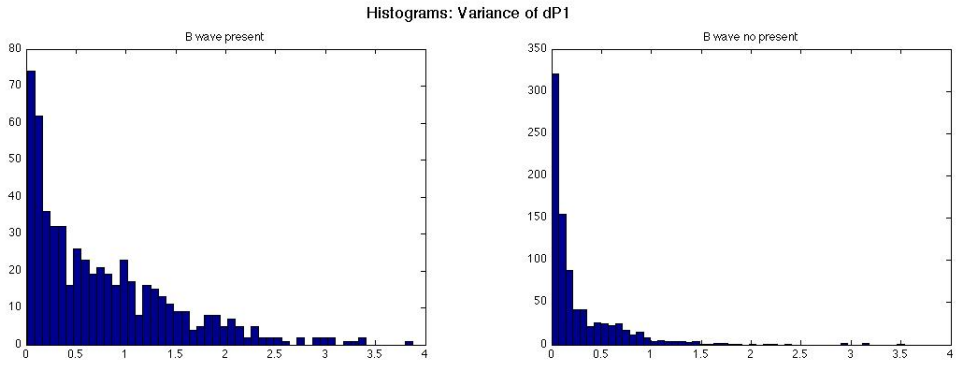


Figure A.4: Histograms of feature 4.

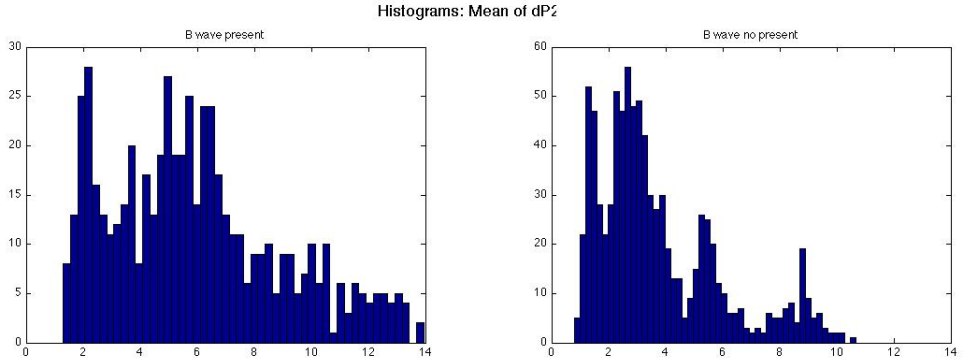


Figure A.5: Histograms of feature 5.

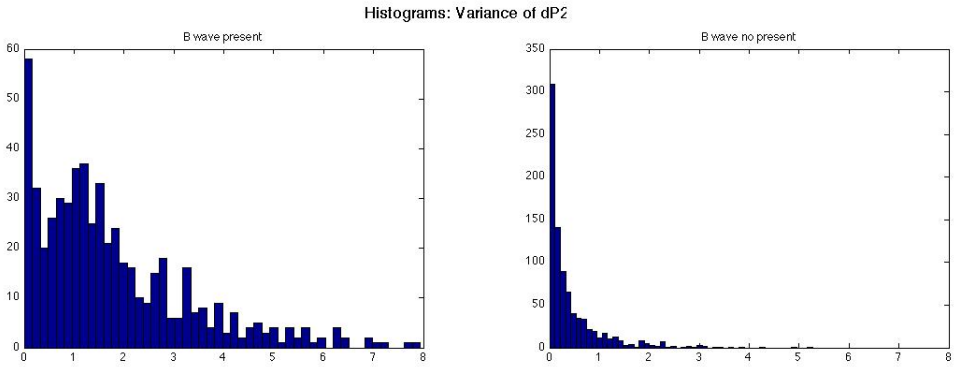


Figure A.6: Histograms of feature 6.

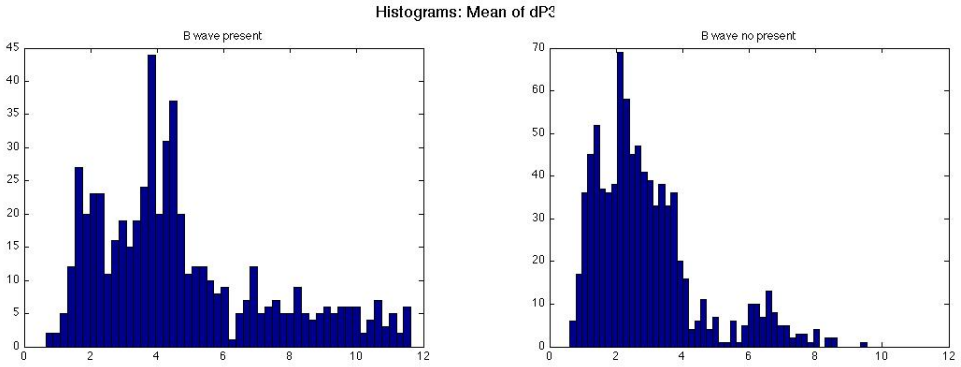


Figure A.7: Histograms of feature 7.

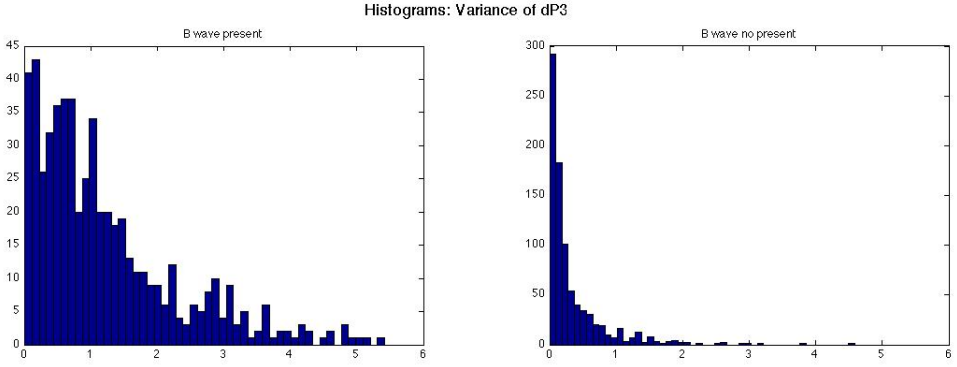


Figure A.8: Histograms of feature 8.

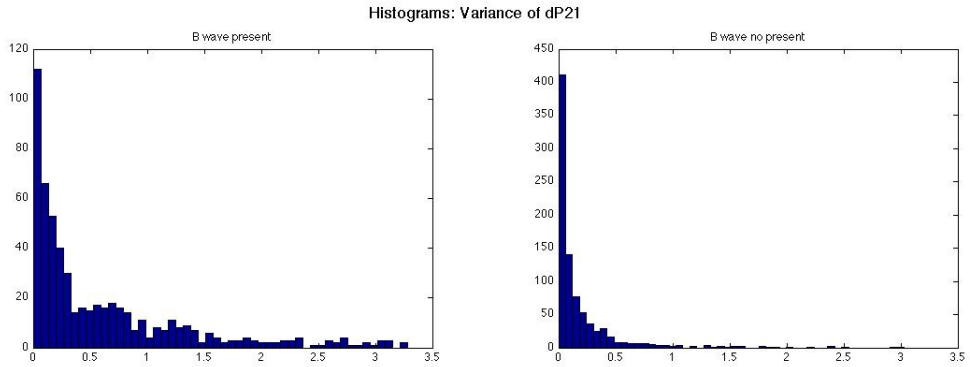


Figure A.9: Histograms of feature 9.

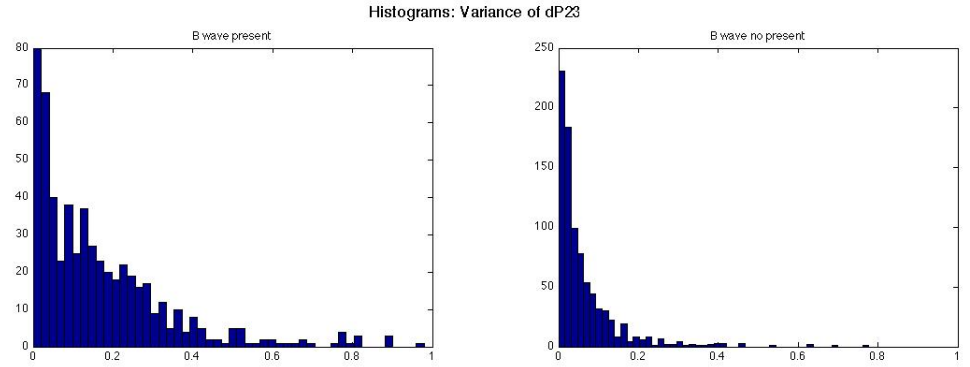


Figure A.10: Histograms of feature 10.

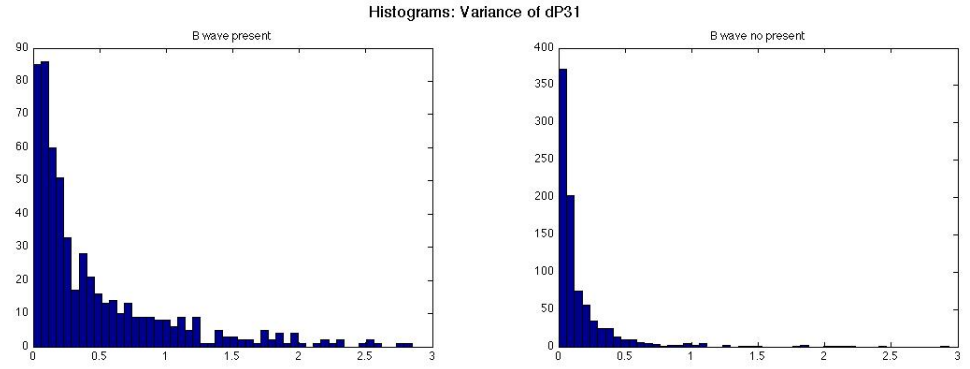


Figure A.11: Histograms of feature 11.

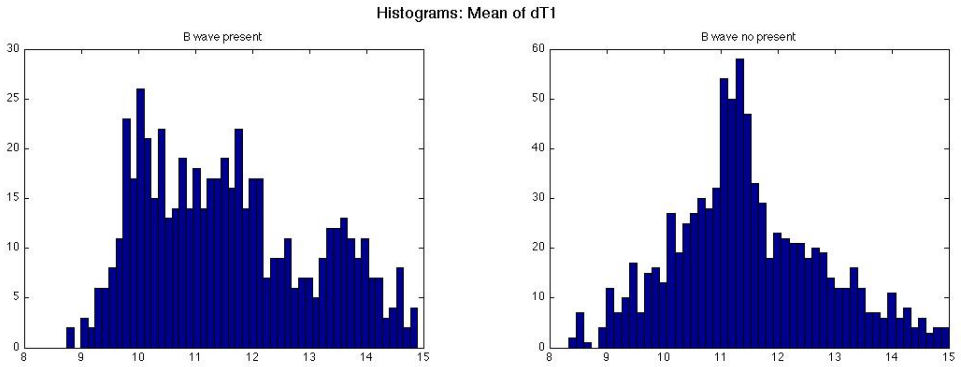


Figure A.12: Histograms of feature 12.

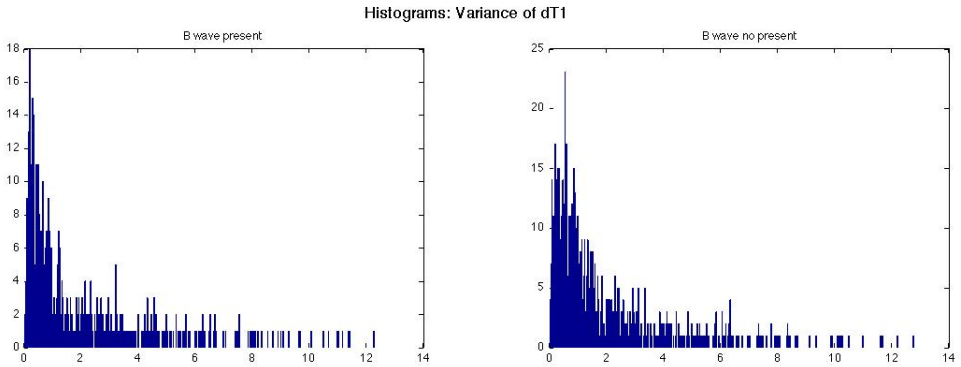


Figure A.13: Histograms of feature 13.

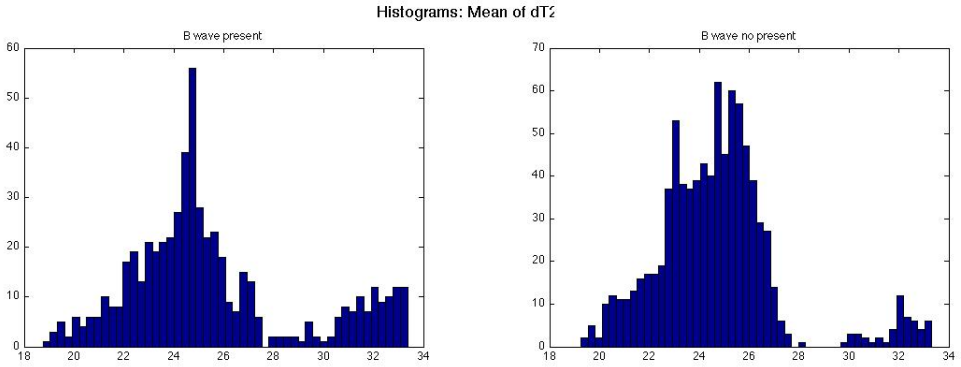


Figure A.14: Histograms of feature 14.

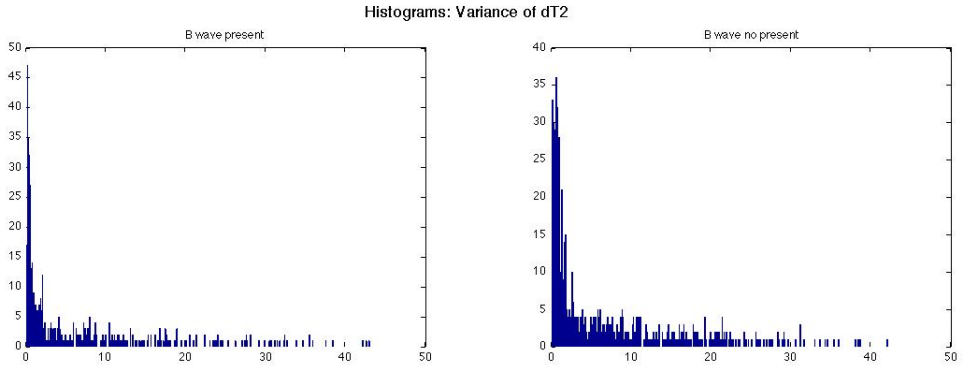


Figure A.15: Histograms of feature 15.

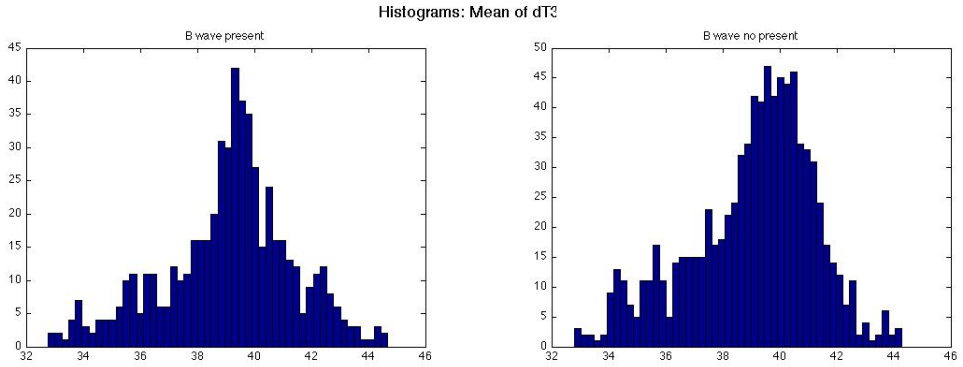


Figure A.16: Histograms of feature 16.

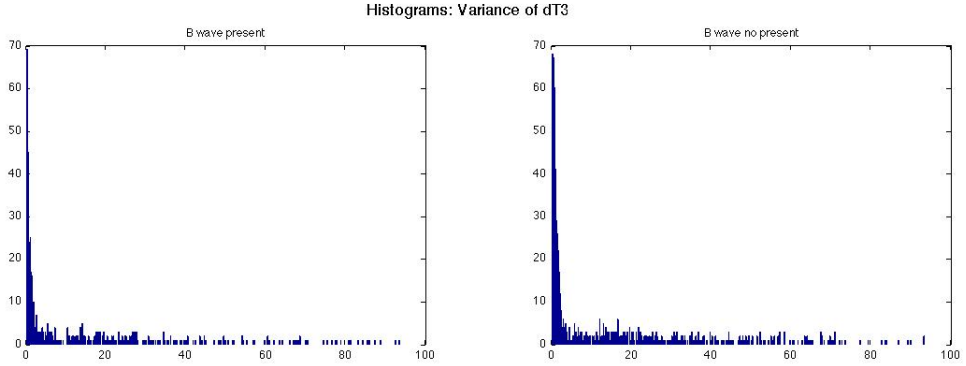


Figure A.17: Histograms of feature 17.

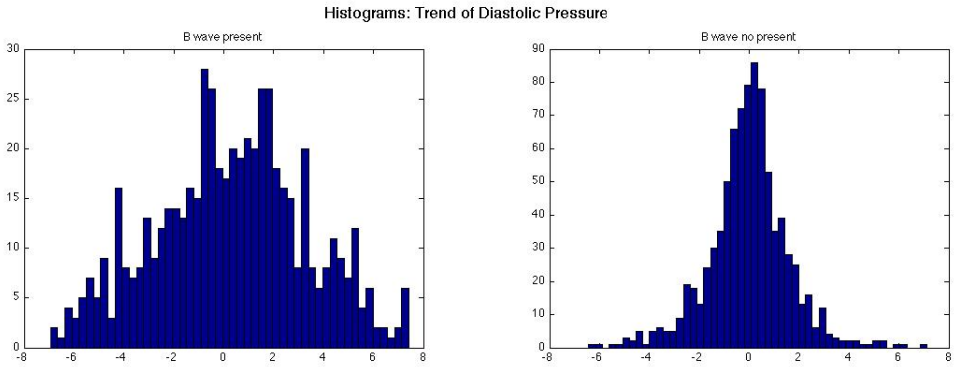


Figure A.18: Histograms of feature 18.

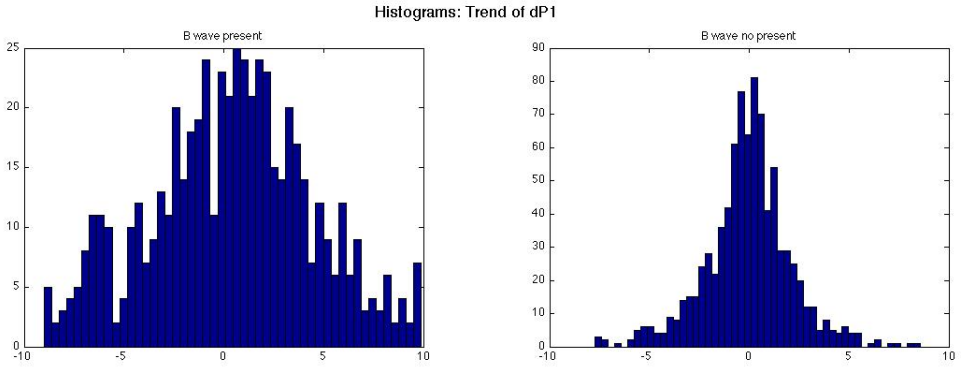


Figure A.19: Histograms of feature 19.

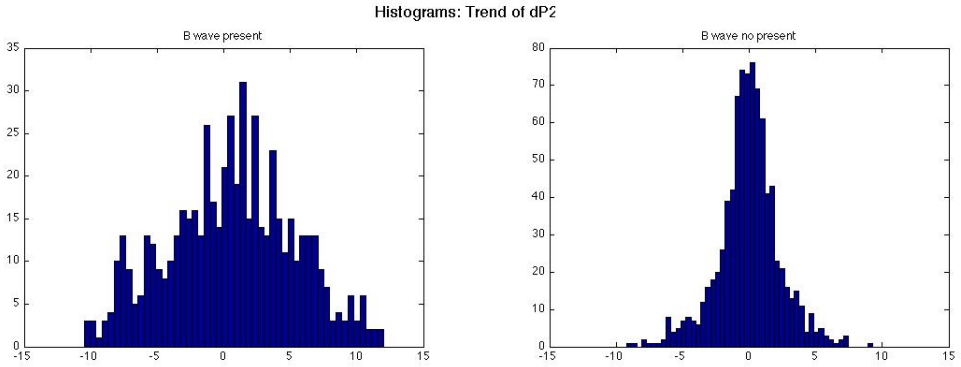


Figure A.20: Histograms of feature 20.

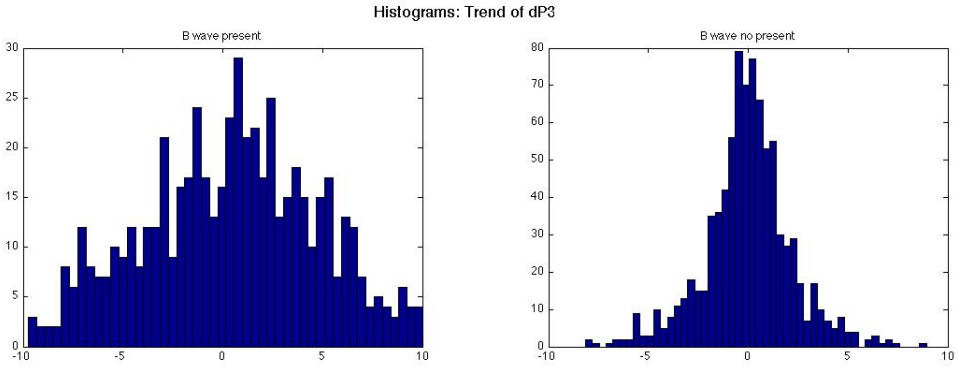


Figure A.21: Histograms of feature 21.

Bibliography

[1] Neurological Disorders, National I. ; Stroke: *Hydrocephalus.* `http://espanol.ninds.nih.gov/trastornos/la_hidrocefalia.htm`. Version: 8 2012

[2] Ontario (SBH), Spina Bifida Hydrocephalus A.: *About Normal Pressure Hydrocephalus (NPH).* `http://www.sbhao.on.ca/hydrocephalus/normal-pressure-hydrocephalus`. Version: April 2013, Abruf: 2013.04.2

[3] Bech RA, Waldemar G Klinken L Gjerris F. Juhler M M. Juhler M: Frontal brain and leptomeningeal biopsy specimens correlated with cerebrospinal fluid outflow resistance and B-wave activity in patients suspected of normal-pressure hydrocephalus. In: *Neurosurgery* 40 (1997), S. 497–502

[4] Droste DW, Krauss J.: Intracranial pressure B-waves precede corresponding arterial blood pressure oscillations in patients with suspected normal pressure hydrocephalus. In: *Neurological Research* 21 (1999), S. 627–630

[5] Society, Japan N. (Hrsg.): *Guidelines for Management of Idiopathic Normal Pressure Hydrocephalus.* Medical Review Co., 2004

[6] A, Monro: *Observations on structure and functions of the nervous system.* Ceech and Johnson, 1783

[7] Sergio Mascarenhas, C. Carlotti L.E.G Damiano W. Seluque B. Colli K. Tanaka C.C. W. G.H.F. Vilela V. G.H.F. Vilela ; Nonaka, K.O.: The new ICP minimally invasive method shows that the Monro-Kellie doctrine is not valid. In: *Intracranial Pressure and Brain Monitoring XIV* 114 (2010), S. 117–220

[8] K., Becker: *Management of increased intracranial pressure.* ANN Syllabi, 2000

[9] Irma Fiordalidi, Glenn D. H. ; Gilliland., M.G.F: Prehospital cardiac arrest in diabetic ketoacidemia. Why brain swelling may lead to death before treatment. In: *Journal of Diabetes and its Complications* 16 (2002), S. 214–219

[10] Irma Fiordalidi, Glenn D. H. ; Gilliland., M.G.F: Fifth International Hydrocaphalus Workshop, Crete, Greece, May 20-23-2010: Themes and Highlights. In: *Hydrocephalus. Selected Papers from the International Workshop in Crete* 113 (2010), S. 1–7

[11] *What is Hydrocephalus?* `http://www.seattlechildrens.org/medical-conditions/brain-nervous-system-mental-conditions/hydrocephalus/`. Version: 8 2012, Abruf: 2012.08.31

[12] *Children's doctor improves hydrocephalus treatment in Africa.* http://childrenshospitalblog.org/childrens-doctor-imporves-hydrocephalus-treatment-in-africa/. Version: 8 2012, Abruf: 2012.08.31

[13] Wikipedia: *Hydrocephalus.* http://en.wikipedia.org/wiki/Hydrocephalus. Version: 9 2012

[14] Codman Shurtleff, Inc: *Diagnosis of NPH.* http://www.lifenph.com/. Version: 9 2012

[15] Inc., MedicineNet: *Hydrocephalus.* http://www.medicinenet.com/hydrocephalus/page3.htm#what_is_the_current_treatment_for_hydrocephalus. Version: 2012 9

[16] Nouzhan Sehati, MD: *Ventriculoperitoneal (VP) Shunt Placement.* http://sehati.org/index/patientresources/neurosurgicalprocedures/vpshunt.html. Version: 9 2012

[17] Inga Margrit Elixmann, M. K. M. Walter W. M. Walter ; Leonhardt., S.: Simulation of Existing and Future Electromechanical Shunt Valves in Combination with a Model for Brain Fluids Dynamics. In: *Hydrocephalus. Selected Papers from the International Workshop in Crete* 113 (2010), S. 77–82

[18] Jan-Uwe Müller, Joachim O. Jürgen Piek ; Gaab, Michael R.: *Intracranial pressure (ICP) and cerebrospinal fluid (CFS) dynamics.* http://www.panarabneurosurgery.org.sa/journal/vol4_2/ICP_and_CSF/JournalVol4_2_ICP_and_CSF.htm. Version: 9 2012

[19] J.J Lemaire, F. Cervenansky G. Gindre J. Y. Boire J.E. Bazin B. I. T. Khalil K. T. Khalil ; Chazal., J.: Slow Pressure waves in the Cranial Enclosure. In: *Acta Neurochir (Wien)* (2002)

[20] Auer LM, Sayama I.: Intracranial pressure oscillations (B-waves) caused by oscillations in cerebrovascular volume. In: *Acta Neurochir (Wien)* 68 (1983), S. 93–100

[21] Krauss JK, Bohus M Regel JP Scheremet R Riemann D Seeger W. Droste DW D. Droste DW: The relation of intracranial pressure B-waves to different sleep stages in patients with suspected normal pressure hydrocephalus. In: *Acta Neurochir (Wien)* 136 (1995), S. 195–203

[22] Lemaire JJ, Chazal J Irthurm B. Boire JY J. Boire JY: A computer software for frequential analysis of slow intracranial pressure waves. In: *Computer Methods and Programs in Biomedicine* 42 (1994), February, S. 1–14

[23] Dunn, Laurence T.: Raised Intracranial Pressure. In: *Neurology, Neurosurgery and Psychiatry* 73 (2002), S. 23–27

[24] Müller JU, Tschiltschke W. Junge HM H. Junge HM: B-waves in the intracranial pressure - a physiological phenomenon and monitoring of severely head injured patients. In: *Zentralbl Neurochir* 15 (1998)

[25] Siddiqui, Javed: *Neurosurgical Intensive Care.* Thieme Medical Publishers, Inc, 2007. – 225–226 S.

[26] Keita Hara, Kohji Ozki Takuya I. Susumu Nakatani N. Susumu Nakatani ; Mogami, Heitarou: Detection of the B waves in the oscillation of intracranial pressure by fast Fourier transform. In: *Medical Informatics* 15 (1990), Nr. 2, S. 125–131

[27] Stefanie Jetzki, Regina Eymann Dilpreet Buxi Marian W. Michael Kiefer K. Michael Kiefer ; Leonhardt., Steffen: Automatische Erkennung Intrakranieller B-wallen. In: *Jahrestagung der Deutschen Gesellschaft für Biomedizinische Technic, RWTH Aachen.* (2007), S. 26–29

[28] Hans E. Heissler, Joachim K. K. Kathrin König K. Kathrin König ; Rickels, Eckhard: Analysis of Intracranial Pressure Time Series Using Wavelets. In: *Intracranial Pressure and Brain Monitoring XIV* 114 (2012), S. 87–92

[29] Inga Margrit Elixmann, C. Goddin S. Antes K. Radermacher S. Leonhardt Senior Member I. J. Hansinger H. J. Hansinger: Single Pulse Analysis of Intracranial Pressure for Hydrocephalus implant.

[30] Magdalena Kasprowicz, Marvin Bergsneider Marek Czosnyka Robert Hamilton Xiao H. Shadnaz Asgari A. Shadnaz Asgari: Pattern Recognition of overnight intracranial pressure slow waves using morphological features of intracranial pressure pulse. In: *Journal of Neuroscience Methods.* 190 (2010), S. 310–318

[31] Hu Xu, Scalzo F Vespa P Bergsneider M. Xu P P. Xu P: Morphological clustering and analysis of continuous intracranial pressure. In: *IEE Trans Biomed Eng.* 56 (2009), S. 696–705

[32] Magdalena Kasprowicz, Marek C. Marvin Bergsneider B. Marvin Bergsneider ; Hu., Xiao: Association between ICP Pulse Wafeform Morphology and ICP B Waves. In: *Intracranial Pressure and Brain Monitoring XIV* 114 (2012), S. 29–34

[33] M. Czosnyka, J. D. P.: Monitoring and interpretation of incracranial presure. In: *Neural Neurosurg Psychiatry.* 75 (2004), S. 813–821

[34] Richard O. Duda, David G. S. Peter E. Hart H. Peter E. Hart: *Pattern Classification.* 2. JOHN WILEY SONS, INC., 2001

[35] Bishop, Christopher M.: *Pattern Recognition and Machine Learning.* Springer, 2006