Universidade de Aveiro
2022

João Miguel Rafael de Carvalho

Reconhecimento de Padrões Baseado em Compressão: Um exemplo de biometria utilizando ECG

Compression-based Pattern Recognition: An example on ECG Biometrics

Universidade de Aveiro
2022

João Miguel Rafael de Carvalho

Reconhecimento de Padrões Baseado em Compressão: Um exemplo de biometria utilizando ECG

Compression-based Pattern Recognition: An example on ECG Biometrics

Tese apresentada às Universidades de Aveiro, Minho e Porto para cumprimento dos requisitos necessários à obtenção do grau de Doutor em Informática (MAP-i), realizada sob a orientação científica da Doutora Susana Manuela Martinho dos Santos Baía Brás, Investigadora do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro e co-orientação do Doutor Armando José Formoso de Pinho, Professor Catedrático do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro.

Trabalho financiado pelas seguintes entidades:

Ao meu tio Ezequiel, que foi a minha bússola ao longo de todo o percurso académico.

o júri / the jury

presidente / president	**Doutor Fernando José Mendes Gonçalves** Professor Catedrático, Universidade de Aveiro
vogais / examiners committee	**Doutor Mário Alexandre Teles de Figueiredo** Professor Catedrático, Universidade de Lisboa
	Doutor Luís Filipe Barbosa de Almeida Alexandre Professor Catedrático, Universidade da Beira Interior
	Doutora Maria Beatriz Alves de Sousa Santos Professora Associada com Agregação, Universidade de Aveiro
	Doutor Ricardo Jorge Gomes Lopes da Rocha Professor Associado, Universidade do Porto
	Doutora Susana Manuela Martinho dos Santos Baía Brás Investigadora Doutorada (nível 1), Universidade de Aveiro

agradecimentos / acknowledgements

First and foremost, I would like to express my deep gratitude towards my supervisors. Both were always available for discussing any topics or questions I had, and promoted an environment where contradicting points of view could be presented without worries and were seen as the way to reach real knowledge. This has allowed an honest exchange of ideas, which made these years much more enriching. On a more personal note, if it were not for their understanding, this document would never have arrived to completion. So, thank you both, truly.

I would also like to thank all my colleagues and friends, but specially: my girlfriend Cláudia Pimenta, for motivating me to write this document and being genuinely happy for my achievements; Pedro Nora and Leandro Gomes, for convincing me to enroll on the PhD program and sharing this journey; Morteza Hosseini, Ana Martins, Diana Rocha, Diogo Pratas and Ana Rocha for making the workplace an enjoyable place to be; and all the people that work in the Institute of Electronics and Informatics Engineering of Aveiro and whom I have had the privilege to interact on a regular basis, namely (but not exclusively): Sónia Gouveia, *Sr. Zé Manel*, Anabela Viegas, Ana Isabel Martins, Sónia Brandão, Raquel Sebastião, José Maria Fernandes, Ilídio Oliveira, Samuel Silva, amongst many others.

I also would like to acknowledge and thank for the financial support from Fundação para a Ciência e Tecnologia – FCT (SFRH/BD/136815/2018).

Finally, I owe the biggest appreciation to my family, who always believed in me and supported my decisions throughout my entire life.

Palavras-chave

Complexidade de Kolmogorov, modelos de contexto-finito, compressão relativa, reconhecimento de padrões, biometria, sinais eletrocardiográficos

Resumo

A quantidade de dados recolhidos por sensores e dispositivos inteligentes que as pessoas utilizam no seu dia a dia tem aumentado a taxas mais elevadas do que nunca. Isso possibilita a utilização de sinais biomédicos em diversas aplicações práticas, com o auxílio de algoritmos de reconhecimento de padrões. Nesta tese, investigamos o uso de métodos baseados em compressão para realizar classificação de sinais unidimensionais. Para testar esses métodos, utilizamos, como aplicação de exemplo, o problema de identificação biométrica através de sinais electrocardiográficos (ECG).

Em primeiro lugar, introduzimos a noção de complexidade de Kolmogorov e a forma como a mesma se relaciona com os métodos de compressão. De seguida, explicamos como esses métodos são úteis para reconhecimento de padrões, explorando diferentes medidas baseadas em compressão, nomeadamente, a compressão relativa normalizada (NRC), uma medida baseada na similaridade relativa entre strings. Para isso, apresentamos os modelos de contexto finito e explicaremos a teoria por detrás de uma versão generalizada desses modelos, chamados de modelos de contexto finito de alfabeto estendido (xaFCM), uma nova contribuição.

Uma vez que a aplicação de exemplo para os métodos apresentados na tese é baseada em sinais de ECG, explicamos também o que constitui tal sinal e os métodos que devem ser utilizados antes que a compressão de dados possa ser aplicada aos mesmos, tais como filtragem e quantização.

Por fim, exploramos com maior profundidade a aplicação da identificação biométrica utilizando o sinal de ECG, realizando alguns testes relativos à aquisição de sinais e comparando diferentes propostas baseadas em métodos de compressão, nomeadamente os não fiduciais. Destacamos também as vantagens de tal abordagem, alternativa aos métodos de aprendizagem computacional, nomeadamente, baixo custo computacional bem como não exigir tipo de extração de atributos, tornando esta abordagem mais facilmente transponível para diferentes aplicações e sinais.

Keywords Kolmogorov complexity, finite-context models, relative compression, pattern recognition, biometrics, electrocardiographic signal

Abstract

The amount of data being collected by sensors and smart devices that people use on their daily lives has been increasing at higher rates than ever before. That enables the possibility of using biomedical signals in several applications, with the aid of pattern recognition algorithms in several applications. In this thesis we investigate the usage of compression based methods to perform classification using one-dimensional signals. In order to test those methods, we use as testbed example, electrocardiographic (ECG) signals and the task biometric identification.

First and foremost, we introduce the notion of Kolmogorov complexity and how it relates with compression methods. Then, we explain how can these methods be useful for pattern recognition, by exploring different compression-based measures, namely, the Normalized Relative Compression, a measure based on the relative similarity between strings. For this purpose, we present finite-context models and explain the theory behind a generalized version of those models, called the extended-alphabet finite-context models, a novel contribution.

Since the testbed application for the methods presented in the thesis is based on ECG signals, we explain what constitutes such a signal and the methods that should be used before data compresison can be applied to them, such as filtering and quantization.

Finally, we explore the application of biometric identification using the ECG signal into more depth, making some tests regarding the acquisition of signals and benchmark different proposals based on compresison methods, namely, non-fiducial ones. We also highlight the advantages of such an alternative approach to machine learning methods, namely, low computational costs and not requiring any kind of feature extraction, making this approach easily transferable into different applications and signals.

Contents

List of Figures

List of Tables

Acronyms

CDM Compression-based Dissimilarity Measure. 39

ECG Electrocardiographic signal. i, v, vii, 1, 2, 3, 4, 5, 24, 25, 27, 28, 29, 30, 31, 32, 33, 34, 38, 39, 40, 42, 43, 44, 45, 47, 48, 50, 51, 52, 53, 56, 57, 58, 59, 60, 61, 64, 65, 67, 68, 69, 70, 71, 75

FCM Finite Context Model. 2, 16, 17, 19, 20, 24, 46, 47, 48, 50, 51, 52, 53, 54, 64, 67, 69, 70, 73, 74, 75

ID Information Distance. 14

NCCD Normalized Conditional Compression Distance. 15, 24

NCD Normalized Compression Distance. 15, 24

NID Normalized Information Distance. 14, 15, 24

NRC Normalized Relative Compression. 2, 16, 24, 39, 41, 43, 47, 67, 69, 70, 74, 75

PAA Piecewise Aggregate Approximation. 35, 36

SAX Symbolic Aggregate AproXimation. v, 2, 36, 40, 42, 44, 45, 48

xaFCM Extended-Alphabet Finite-Context Model. v, 19, 20, 24, 50, 51, 52, 53, 67, 68, 73, 74, 75

Chapter 1

Introduction

1.1 Context

Data compression techniques surmount to the middle of the 19th century, when the inventor (and famous painter) Samuel Morse, along with Alfred Vail, proposed a system for transmitting messages over a telegraph, which was later called the Morse code [1]. The main idea behind this code was simple: letters more frequent on the English language were represented by shorter codes, and vice-versa. This meant that it was possible to transmit English messages using less time (at least, on average). This idea has set the basis to different data compression methods which have emerged since those days, but where the goal remains the same: minimizing some unit of measure in which data can be transmitted–usually, bits of data. In this thesis we aim at using some of those ideas, applied to classification problems, which are usually tackled from a machine learning approach.

Machine learning methods have been used for decades, but are now used on more tasks and domains of society than ever before, given the amount of data available from wearable devices and sensors around the world and the need and desire for companies to automate tasks [2]. They are also commonly used for dealing with biomedical signals, such as the ECG, as identifying patterns in physiological signals is, undoubtedly, an important tool to understand non-trivial phenomena on these signals, as well as to aid in clinical practice. However, usually, machine learning methods require feature extraction to be done, making each problem specific enough to be hard to obtain a more general solution, apt for different tasks. Parameter free data mining methods, on the other hand, are reported in the literature as efficient in classification and extraction of information [3]. Since there is no pre-assumption about the premises, it allows true exploratory data mining. Deep neural networks, of different architectures, like Multi-Layer Perceptrons [4], Convolutional Neural Networks [5] or Recurrent Neural Networks [6, 7], as well as other architectures derived from these ones, can be considered as part of this category methods, as they do not require any kind of feature extraction and provide good results without much parameter *tweaking*. However, they rely on high computational costs and, usually, high amounts of available data. This makes them not possible to use under certain scenarios where the computational power available is low or should be minimized for other reasons, such as minimizing the time of execution of the task at hand (real time scenario) or battery usage, for example.

Data compression models can be used as an alternative to address several data mining and machine learning problems, usually by means of a formalization in terms of the information content of a string or of the information distance between strings [8, 9, 10, 11, 12]. This approach relies on solid foundations of the concept of algorithmic entropy and, because of its non-computability, approximations provided by data compression algorithms [13]. The use of Finite Context Model (FCM) based compressors are examples of such methods. These methods correctly deal with some of the problems reported in the literature for dealing with biomedical signals, namely, for ECG biometric identification, such as: variability, noise, amongst others [14]. However, data compression algorithms are symbolic in nature. Text and DNA sequences are well-known examples of symbolic sequences, with well-defined associated alphabets, where this approach can be directly applied, without any major transformations. ECG data (or many other types of data, such as sound, images, to name a few), need to first be transformed into symbols, usually through sampling and quantization, before lossless data compression can be applied to it.

Early attempts to conciliate compression-based data mining and machine learning with non-symbolic data [3] relied on the SAX representation [15], which we will also use in the current thesis. Notwithstanding, many problems still lack satisfactory solutions, for example, how does time or space and amplitude changes of scale influence the performance of the data mining and machine learning algorithms and, how should they be performed in order to maximize the results. In this work we will experiment with different quantization schemes, evaluate the results and explain the choices made, as the quantization is one important step to perform classification using non-symbolic signals.

1.2 Goals

The main goals of this PhD research project are to explore how different lossless compression ideas can be used for performing pattern recognition on one-dimensional signals. We will be focusing on ECG signals, to perform biometric identification, as an example, but the final goal is to propose methods that are agnostic to the signal being used–*i.e.*, that can be translated into different signals and/or applications, only changing the filtering techniques and parameters of the methods.

In order to do so, we will be focusing on using finite-context model based compressors and using the Normalized Relative Compression (NRC), a relative compression measure, in order to compute similarities amongst segments of data. Given that those compression techniques rely on symbolic data, we also propose some modifications to existing quantization techniques in order to allow the usage of finite-context models.

We also aim at making some improvements regarding the compression models used to compute similarity metrics, so we propose a generalization of finite-context models, that is faster and has memory advantages over the traditional finite-context models, allowing for the proposed methods for pattern recognition to execute in a shorter amount of time, without compromising the final results.

The final and main goal of this project is to discuss the limitations of the proposed approaches and what should be regarded as useful “lessons”, even while doing a non-compression-based approach to solve the same issue, *i.e.*, using traditional methods.

1.3 Datasets

During this project different ECG datasets were used, both for biometric identification, emotion recognition or even more general ECG pattern recognition tasks, like noise/outlier heartbeat detection, for example. In this section we present the main datasets used.

In order to avoid repeating blocks of text along the thesis, whenever a database from the list below is mentioned during the rest of the document, we will only refer to its name and link to its corresponding section on this list to find a more detailed description.

1.3.1 Vital Jacket DB

The database is composed by 67 hours of ECG signal, from which 19 hours (7 signals with 2h40 each, approximately) correspond to a clean ECG signal. The signals were collected using a simulator with artificial/synthetic noise added. Values were gathered from the simulator using a Vital Jacket® [16], a scientifically validated biomedical device, developed by a technological company, called Biodevices SA, in partnership with researchers from IEETA, from the University of Aveiro, which provides an ECG signal with a sampling frequency of 500 Hz. The collecting protocol consisted in gradually increasing and decreasing the heart rate in 20 minute intervals. In total, the signal is composed by 8 heart rate steps in the following order: 60, 80, 100, 120, 140, 120, 100, 80 beats per minute.

The reason for building such a dataset was to know *a priori* which regions of the ECG contain noise, which can be useful for building an ECG noise detection algorithm and/or to measure the impact that the noise has on the overall performance of a certain application.

The noise values were randomly generated using a uniform discrete distribution between the minimum and maximum values (117 and 159, respectively) of the collected signal. This allowed us to have a controlled signal with noise in specific zones.

1.3.2 Biometric Emotion Recognition DB - UA

This database was collected *in house* [17][18], where 25 participants were exposed to different external stimuli – *disgust*, *fear* and *neutral*. Data were collected on three different days (once per week), at the University of Aveiro, using a different stimulus per day.

The data signals were collected during 25 minutes on each day, giving a total of around 75 minutes of ECG signal per participant. Before being exposed to the stimuli, during the first 4 minutes of each data acquisition, the participants watched a movie with a beach sunset and an acoustic guitar soundtrack, and were instructed to try to relax as much as possible.

The ECG was sampled at 1000 Hz, using the MP100 system and the software AcqKnowledge (Biopac Systems, Inc.). During the preparation phase, the adhesive disposable Ag/AgCL-electrodes were fixed in the right hand, as well as in the right and left foot. We are aware that such an intrusive set-up is not desirable for a real biometric identification system. However, for testing purposes, it seems appropriate, as this approach is more reliable – produces less noise.

Even though this database was built for emotion recognition, by using a database where the participants were exposed to different stimuli, it is possible to check if the emotional state of participants affects the biometric identification process as well.

The database is publicly available for download at [1].

1.3.3 PTB Diagnostic ECG Database

This database contains a total of 549 records from 290 subjects (aged from 17 to 87 years old). It was collected for clinical purposes and it is part of Physionet [19].

Each subject is represented by one to five records. Each record includes the conventional 12 leads as well as the 3 Frank lead ECGs. Each signal is digitized at 1000 samples per second, with 16 bit resolution [20].

Each record includes metadata with the clinical information about the pation, including: age, gender, diagnosis, medical history, medication and interventions, coronary artery pathology, ventriculography, echocardiography, and hemodynamics.

The diagnostic classes are missing for 22 subjects. For the other 268 subjects, the distribution of diagnosis can be seen in Table 1.1.

[1] http://sweet.ua.pt/ap/data/signals/Biometric_Emotion_Recognition.zip

Diagnostic class	Number of subjects
Myocardial infarction	148
Cardiomyopathy/Heart failure	18
Bundle branch block	15
Dysrhythmia	14
Myocardial hypertrophy	7
Valvular heart disease	6
Myocarditis	4
Miscellaneous	4
Healthy controls	52

Table 1.1: Diagnostic classes from the 268 subjects available on PTB Diagnostic ECG Database.

1.3.4 UofT ECG Database (UofTDB)

The UofTDB is an ECG database which was collected in the University of Toronto and contains a total of 1012 healthy subjects. We would like to thank Prof. Dimitrios Hatzinakos for allowing us access to the data, after a formal request, for research purposes.

This dataset includes ECG recordings from volunteers in multiple sessions, under exercise condition and different body postures. For specific volunteers, the ECG was recorded over five additional recording sessions which spanned over a six-month interval. These recordings are available only from participants who agreed to participate in follow-up studies.

Each recordings' length varies from two to five minutes. The database includes ECG recordings in four different postures: sit, stand, supine and tripod.

The ECG signal was captured from the volunteers' fingertips, similar to a lead I configuration: the subjects held a pad with dry AgCL electrodes positioned such that the left thumb was placed on the positive electrode, the right thumb on the negative electrode and the right index finger was placed on the reference electrode. The Vernier ECG sensor and the Go!Link interface 1 were used for recording the ECG signals. The device was configured to 200Hz sampling frequency at 12 bits per sample resolution [25].

All the subjects who attended the follow-up sessions were recorded in a seated position and, for that reason, that is the position for which more data is available. Therefore, we have chosen to use only the 46 participants which have 5 ECG samples collected while they were seating [25].

A more detailed description about the database can found in [25] and [26].

1.3.5 ECG Collection Protocol

This dataset was acquired in order to develop a study regarding the data collection protocol for ECG data for biometric identification [27]. The data were collected from twenty healthy participants (nine females and eleven males), all between the ages of 20 and 23 years (20.85 ± 0.91 years). All participants were healthy, without any relevant pathology. Two of the participants were smokers. The collection of this dataset was approved by the ethics council of the University of Aveiro. The participants, recruited at the same university, gave their written consent, after being explained the purpose of the data collection. It was also mentioned to them that the data was confidential and that they could withdraw from the experiment at any time.

Given that the ECG is a vital signal which can be affected by factors such as the skin and the type of electrodes used [28], both dry electrodes (produced by BITalino) and gel ones (manufactured by Ambu®) were used. Both types were of Ag/AgCl type.

The data were collected at 500 Hz, using an Android application called DroidJacket, together with the Vital Jacket®, which allowed to collect data from different devices and redirect them to a common storage location [29].

The gel from the Ag/AgCl electrodes used improves conductivity and reduces the impedance of the skin-electrode interface. However, dry electrodes were placed on the fingers with the aim of making this study close to reality. In daily life, the ideal would be to use dry electrodes in an easily accessible area, without the need to place gel electrodes, since the latter have a more complex placement process and they are also subject to the drying of the gel, leading to a drastic decrease in the quality of the signal, specially when dealing with long acquisition processes [30].

The ECG signals were acquired through a rigorous placement of each electrode in different regions of the body: fingers from both hands, wrists and chest. In each region of the body that ECG signal was collected, three electrodes were placed. Its placement was uniform, since for each of the body locations and for each of the participants, the exact same placement protocol was followed. These electrodes were attached to the end of the wires with different colors, facilitating the process. The red and yellow color electrodes were always placed on a high blood flow region, on the right and left side, respectively. The third electrode, called neutral, was always placed on the right side of the body – in a place with less blood flow. This way, it was possible to collect signals from different collection places simultaneously.

Two data acquisition sessions took place. In both collections, each participant underwent an initial survey with the main objective of evaluating some factors that, according to the literature, can alter the biometric identification “signature” [31]: their levels of anxiety and stress, their last intake of coffee and also if they had any disease. Afterwards, participants watched a documentary, which showed a neutral feeling, thus not introducing any specific emotion. At the end of each collection, a satisfaction survey was completed.

In the first data collection session, each participant was asked to remain at rest for ten minutes. After this period of time, they were asked to perform movements with their hands, feet and torso, for two minutes each, with a pause of one minute between them. In the second session, participants were asked to stay five minutes at rest while signals were being collected.

The two collections carried out had a time spacing of two weeks. This period was rigorously chosen to assess the impact of the temporal distance of data collection for biometric identification.

1.4 Outline

This thesis has a total of five chapters. It starts with an overall introduction about the project. The second chapter presents some compression topics and show how they can be used for pattern recognition. On the third chapter, we introduce the concepts needed for preparing ECG signals before compression can be applied. On the fourth chapter, we merge all the concepts presented on a real classification application using a one-dimensional signal – biometric identification using ECG. Finally, on the last chapter, we present some remarks regarding the project and some suggestions for future work.

1.5 List of Contributions

1.5.1 Publications Indexed by ISI

- Filipa Barros, Cláudia Figueiredo, Susana Brás, João M. Carvalho, and Sandra C. Soares. "Multidimensional assessment of anxiety through the State-Trait Inventory for Cognitive and Somatic Anxiety (STICSA): From dimensionality to response prediction across emotional contexts". In: *PLOS ONE* 17.1 (2022). Ed. by Claudio Imperatori, e0262960. DOI: 10.1371/journal.pone.0262960.

- Mariana S. Ramos, João M. Carvalho, Armando J. Pinho, and Susana Brás. "On the Impact of the Data Acquisition Protocol on ECG Biometric Identification". In: *Sensors* 21.14 (2021), p. 4645. DOI: 10.3390/s21144645.

- Gisela Pinto, João M. Carvalho, Filipa Barros, Sandra C. Soares, Armando J. Pinho, and Susana Brás. "Multimodal Emotion Evaluation: A Physiological Model for Cost-Effective Emotion Classification". In: *Sensors* 20.12 (2020), p. 3510. ISSN: 1424-8220. DOI: 10.3390/s20123510. URL: http://dx.doi.org/10.3390/s20123510.

- Susana Brás, João M. Carvalho, Filipa Barros, Claúdia Figueiredo, Sandra C. Soares, and Armando J. Pinho. "An Information-Theoretical Method for Emotion Classification". In: *IFMBE Proceedings*. Springer International Publishing, 2019, pp. 253–261. DOI: 10.1007/978-3-030-31635-8_30.

- João M. Carvalho, Susana Brás, and Armando J. Pinho. "Compression-Based Classification of ECG Using First-Order Derivatives". In: *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering.* Springer International Publishing, 2019, pp. 27–36. doi: 10.1007/978-3-030-16447-8_3.

- João M. Carvalho, Susana Brás, Diogo Pratas, Jacqueline Ferreira, Sandra C. Soares, and Armando J Pinho. "Extended-alphabet finite-context models". In: *Pattern Recognition Letters* 112 (2018), pp. 49–55. issn: 0167-8655. doi: 10.1016/J.PATREC.2018.05.026. url: https://www.sciencedirect.com/science/article/pii/S0167865518302095.

- João M. Carvalho, Susana Brás, Jacqueline Ferreira, Sandra C. Soares, and Armando J Pinho. "Impact of the Acquisition Time on ECG Compression-Based Biometric Identification Systems". In: *Pattern Recognition and Image Analysis. IbPRIA 2017. Lecture Notes in Computer Science, vol 10255. Springer, Cham.* Springer, Cham, 2017, pp. 169–176. doi: 10.1007/978-3-319-58838-4_19. url: http://link.springer.com/10.1007/978-3-319-58838-4_19.

1.5.2 Other Publications and Communications

- João M. Carvalho, Armando J. Pinho, and Susana Brás. "An Exploratory Study on ECG Biometric Bias Using Compression Algorithms". In: *Proceedings of the 27th RecPad.* 2021.

- João M. Carvalho, Susana Brás, and Armando J. Pinho. "ECG Biometric Identification using Relative Compression". In: *4th Workshop on Maximum Entropy and its Applications.* Department of Mathematics, University of Aveiro, Portugal. July 2021.

- João M. Carvalho, Susana Brás, and Armando J. Pinho. "Compression-based Tools for Non-Symbolic Data". In: *Research Summit 2021.* University of Aveiro, Portugal. 2021.

- João Carvalho, Susana Brás, and Armando J. Pinho. "Entropy-Based ECG Biometric Identification". In: *Proceedings of Entropy 2021: The Scientific Tool of the 21st Century.* MDPI, 2021. doi: 10.3390/entropy2021-09795.

- João M. Carvalho, Susana Brás, and Armando J. Pinho. "Compression-Based Classification of ECG Using First-Order Derivatives". In: *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering.* Springer International Publishing, 2019, pp. 27–36. doi: 10.1007/978-3-030-16447-8_3.

- João M. Carvalho, Susana Brás, and Armando J. Pinho. "Myocardial Infarction Detection Using a Compression-Based Approach". In: *Special Issue - Statistics on Health Decision Making: state of the art.* Vol. 1. 1. May 2019.

- Filipa Barros, C. Costa, C. Figueiredo, Susana Brás, João M. Carvalho, S. Magano, N. Madeira, and S. C Soares. "Cognitive and somatic trait anxiety: subjective and psychophysiological responses." In: *27th European Congress of Psychiatry, Warsaw, Poland,* Apr. 2019.

- Filipa Barros, Cláudia Figueiredo, Susana Brás, João M. Carvalho, and Sandra Soares. "The role of somatic trait anxiety and alexithymia on the physiological and subjective response to emotional stimuli". In: *19th WPA World Congress of Psychiatry.* Aug. 2019.

- João M. Carvalho, Susana Brás, Jacqueline Ferreira, Sandra C. Soares, and Armando J. Pinho. "Impact of the Acquisition Time on ECG Compression-based Biometric Identification Systems". In: *Encontro com a Ciência e Tecnologia em Portugal, Centro de Congressos de Lisboa.* June 2017.

- João M. Carvalho, Susana Brás, Jacqueline Ferreira, Sandra C. Soares, and Armando J Pinho. "Impact of the Acquisition Time on ECG Compression-based Biometric Identification Systems". In: *Research Day 2017.* University of Aveiro, Portugal. July 2017.

- João M. Carvalho, Susana Brás, and Armando J. Pinho. "The Impact of Noise Removal on a Compression-based ECG Biometric Identification System". In: *Proceedings of the 23rd RecPad.* 2017.

- João M. Carvalho, Susana Brás, Jacqueline Ferreira, Sandra C. Soares, and Armando J Pinho. "Impact of the Acquisition Time on ECG Compression-Based Biometric Identification Systems". In: *Pattern Recognition and Image Analysis. IbPRIA 2017. Lecture Notes in Computer Science, vol 10255. Springer, Cham.* Springer, Cham, 2017, pp. 169–176. DOI: 10.1007/978-3-319-58838-4_19. URL: http://link.springer.com/10.1007/978-3-319-58838-4_19.

- João M. Carvalho, Armando J. Pinho, and Susana Brás. "Irregularity Detection in ECG signal using a semi-fiducial method". In: *Proceedings of the 22nd RecPad.* 2016, pp. 75–76.

1.5.3 Awards and Distinctions

- Research Summit 2021 – "Compression-based Tools for Non-Symbolic Data" – Best MAP-i Doctoral Program Pitch Award

- Entropy 2021 – "Entropy-Based ECG Biometric Identification" – Wolphram Award (Best session Poster)

- IbPRIA 20217 – "Impact of the Acquisition Time on ECG Compression-Based Biometric Identification Systems" – Selected Paper for Best Paper

- RecPad 2016 – "Irregularity Detection in ECG signal using a semi-fiducial method" – Selected Paper for Best Paper

Chapter 2

Compression Topics for Pattern Recognition

2.1 Introduction

Compression and coding algorithms have been used for decades on a diverse number of domains and applications. Usually, the motivation for using them is to either reduce the amount of storage space (normally measured in bits) that certain data requires, the bandwidth necessary to transfer data for communications, or even both of them combined.

All compressor schemes follow two steps: the compression and the decompression, done by the encoder and decoder, respectively.

The encoder is responsible for finding patterns on the original data and exploring those patterns to reduce the amount of data required to represent it using a different structure. The data generated by this process is called the compressed data. Given the source and nature of the original data, the reduction of storage needed, denoted by compression ratio, can be of several orders of magnitude or, in case the original data are random or the compressor is not adequate for those specific data, it can be close to one – meaning that the original size and the compressed size are approximately the same.

The decoder is responsible for the inverse process, *i.e.*, it has access to the compressed data and should be able to transform it into something similar to the original data. In the case this transformation is perfect (the decoder is always able to rebuild the original data without any errors), the compressor follows a *lossless compression* scheme. Otherwise, it follows a lossy scheme. Both make sense on different scenarios, but we will not elaborate on that and, for the rest of this document, always assume we are using a *lossless* compression scheme, unless mentioned otherwise.

2.2 Pattern Recognition

Besides aiming at reducing the size of the data, compression algorithms have recently been used on several different applications related with pattern recognition. In order to understand how it is possible to use compression for topics related with pattern recognition and machine learning, we will rely on some examples to explore how a compressor works by analyzing an example:

Example 1: Compressing a text file (encoded in ASCII) containing the text:

$$ABABABABABABABABABABABABABABAB$$

Proposed solution: Given that the text document uses the ASCII representation, each symbol requires 1 byte (8 bits) to be represented. Given that the length of the string is 30, the total file size is 30 bytes (240 bits). This will be our baseline for computing the compression ratio at the end.

Now, there are several aspects we could use to compress this string. The first one, and most obvious, is that the string is just a repetition of the sequence "AB" concatenated fifteen times. So, if a compressor is able to capture that repetition, it can just store something similar to

$$AB \times 15.$$

This means that instead of using 30 symbols and 240 bits it is possible, in theory, to store only the string "AB" (requiring 16 bits) and the number 10 (requiring another 16 bits). Using this representation, we would only require a total of 32 bits to represent the original data. This translates into a compression ratio of $\frac{240}{32} = 7.5$, *i.e.*, the compressed data is 7.5 times smaller than the original one.

Assuming we knew beforehand that all strings that would be compressed by this compressor would only have the symbols "A" and "B", that would allow the compression to be increased even further. Even with only the simple modification that the letters were now represented using a binary form ("A" was transformed into 0 and "B" was transformed into 1, for example), we would save a total of 7 bits per symbol.

In this example, rebuilding the original data using the compression data would be trivial. However, such a simple example is not frequent and there are some steps not being taken into account here, such as the underlying representation scheme used by the compressor, for simplification purposes. Nevertheless, this example should illustrate the basics of what are the main goals of any compressor: to find redundancy patterns on the data to be compressed and exploit them in order to make the data storage requirements smaller.

Example 2: Compressing a text file (encoded in ASCII) containing the first 100 million decimal digits of π.

Proposed solution: Following the same calculations made on the previous example, it is easy to see that this text file would require a total of 800 million bits (approximately 95 megabytes).

Since the distribution of decimal digits from the number π seems to be statistically random [49] (does not follow any kind of known distribution), it is very difficult for any compressor to find statistical patterns on this file. A general purpose compressor should achieve a compression ratio of approximately one, *i.e.*, no compression, as there are no known statistical ways to exploit its distribution of digits.

It is, however, possible to write a very simple computer program under any program language that write those digits into a computer file. We could write a computer program, called P, that computes the arctan(1) with enough precision, multiplies the result by 4 and prints it to a file – lets call this our decoder. There would be other ways to do this, such as using a Spigot algorithm to calculate the sequence [50] and print it, but lets only use the former.

For solving this example, we consider the encoder to be a program that counts the number of digits present in the file and the decoder is the program P. Using both of them, we can achieve the compression/decompression using only the amount of bits required to write that computer program P, plus the amount of bits to store the number "100 000 000" (the amount of digits to compute). It is fair to estimate that this can be done using only a couple of kilobytes, achieving an enormous compression ratio.

One can argue that this is not a "real compressor" (although theoretically it is), as this computer program P serves no other useful purpose than producing that sequence – which makes total sense. This is an unrealistic setting but sets the motivation for introducing the definition of the Kolmogorov Complexity, in the next section.

2.3 Compression-Based Measures

2.3.1 Introduction

Compression-based distances are tightly related to the Kolmogorov notion of complexity (also known as algorithmic entropy).

Let x denote a binary string of finite length. Its Kolmogorov complexity, $K(x)$, is the length of the shortest binary program x^* that computes x in a universal Turing machine and halts. Therefore, $K(x) = |x^*|$, the length of x^*, represents the minimum number of bits from which x can be computationally retrieved [51].

Simplifying the original definition, one can think of a different one, although less formal, where the Kolmogorov complexity of the string x is the size of the smallest computer program possible (in bits), under a specific programming language, that is able to produce the output x. However, this definition raises one issue: it is not possible to prove that a certain program is the smallest to produce x – that is why the Kolmogorov complexity is said to be non-computable and therefore difficult to use in practice.

A way to obtain an estimate for the value of the Kolmogorov complexity of a string is using compression algorithms. A compressor uses the underlying patterns found on the string x to reduce its size as much as possible. It is easy to understand that both concepts are related. Let us define $C(x)$ as the number of bits a certain compressor C requires to represent string x on its compression version plus the size of the corresponding decoder. A question arises: what is the relationship between $K(x)$ and $C(x)$? If we assume that $C(x) < K(x)$, it means the compressor C takes less bits to produce x than $K(x)$ states. But if we take into consideration that the compressor C is a valid computer program that can be used to compute x (and halts), we obtain a contradiction: $C(x) > K(x)$ and $C(x) < K(x)$, proving the initial assumption was invalid. On the other hand, to test whether $C(x) \geq K(x)$ is true, we should divide it into two cases: if C is indeed the smallest computer program that computes x and halts, and if it is not. If it is the former, than the equality takes place ($C(x) = K(x)$). Otherwise, it means there is a smaller computer program than C to compute x, so we have proven that $C(x) \geq K(x)$. Therefore, we can conclude that compression is a way to obtain an upper bound for the value of the Kolmogorov complexity of a certain string. The higher the compression achieved, the most likely the value will be close to its non-computable Kolmogorov complexity's value (the more accurate the approximation is).

A different way to approach the Kolmogorov complexity is using the Information Distance (ID) and its normalized version, the Normalized Information Distance (NID), proposed by Bennett *et al.* in 1998 [52]. These two metrics are defined in terms of the Kolmogorov complexity of the strings involved, as well as the complexity of one when the other is provided. The NID can be defined as

$$\mathrm{NID}(x, y) = \frac{\max\{K(x|y), K(y|x)\}}{\max\{K(x), K(y)\}}, \tag{2.1}$$

where $K(x|y)$ stands for the Kolmogorov complexity of string x when string y is already provided. It is easy to realize that if $x = y$, then $K(x|y) = 0$, given that the smallest Turing machine that computes y would also produce x, as they are equal.

2.3.2 Normalized Compression Distance (NCD)

The Normalized Compression Distance (NCD) is perhaps the most well-known example of how compression can be used for measuring the similarity between strings. This metric was introduced by Li *et al.* [13] and has been used in a number of different applications: music classification [53], to analyze network traffic and to cluster computer worms and viruses [54], authorship attribution [55], gene expression dynamics [56], predicting useful versus useless stem cells [57], critical networks [58], image registration [59], question-answer systems [60]. The NCD is defined as

$$\text{NCD(x,y)} = \frac{C(xy) - \min\{C(x), C(y)\}}{\max\{C(x), C(y)\}}, \tag{2.2}$$

where, like in the previous examples, $C(x)$ and $C(y)$ represent the number of bits of a compressed version of x and y, respectively. Besides, $C(xy)$ represents the number of bits of compression of x and y concatenated.

The NCD is shown to be *quasi-universal* in the sense that it minorizes every computable similarity distance up to an error that depends on the quality of the compressor's approximation of the true Kolmogorov complexities of the files involved [53].

2.3.3 Nornalized Conditional Compression Distance (NCCD)

As it has already been mentioned, the Kolmogorov complexity is non-computable and, therefore, the NID is also non-computable. However, an approximation for it can be used by using compression, providing an upper-bound for its real value. Let $C(x)$ be the number of bits used by a compressor to represent the string x. By replacing the function K by an approximation C we obtain the Normalized Conditional Compression Distance (NCCD), defined by

$$\text{NCCD}(x, y) = \frac{\max\{C(x|y), C(y|x)\}}{\max\{C(x), C(y)\}}. \tag{2.3}$$

Here, $C(y|x)$, also called *conditional compressed information* [53], can be understood as the complexity of a string y with a string x being already known by the compressor, *i.e.*, when the compressor adds the patterns learned on y to the patterns previously extracted while compressing x.

2.3.4 Normalized Relative Compression (NRC)

In 1993, Jacob Ziv, one of the two authors of the popular LZ77 [61] and LZ78 [62] lossless data compression algorithms, proposed a measure of relative entropy between individual sequences [63], which relies on the notions of cross entropy and cross parsing. Using a related approach, Pinho *et al.* introduced a more general compression measure, the *relative compression* [64], denoted by $C(x\|y)$.

This measure represents the compression of x relatively to y. In other words, it represents the amount of bits required by a compressor to represent x using exclusively the models learned from y.

Using this measure, the authors also introduced a measure called the NRC. The NRC of the binary string x, relatively to the binary string y, is defined as

$$\mathrm{NRC}(x\|y) = \frac{C(x\|y)}{|x|}, \tag{2.4}$$

where $|x|$ denotes the length of x.

This measure of relative similarity, $C(x\|y)$, obeys the following rules:

- $C(x\|y) \approx 0$ iff string x can be built efficiently from y;
- $C(x\|y) \approx |x|$ iff $K(x|y) \approx K(x)$.

A more general formula for the NRC of string x, given string y, where the strings x and y are sequences from an alphabet $\mathcal{A} = \{s_1, s_2, \ldots s_{|\mathcal{A}|}\}$, is given by

$$\mathrm{NRC}(x\|y) = \frac{C(x\|y)}{|x| \log_2 |\mathcal{A}|}. \tag{2.5}$$

The reason for normalizing $C(x\|y)$ by the denominator $|x| \log_2 |\mathcal{A}|$ is that the denominator represents the minimum amount of bits required to represent the string x without applying any compression technique, assuming a uniform distribution of the symbols from the alphabet, since $\log_2 |\mathcal{A}|$ is the amount of bits required to represent a symbol from the alphabet $\mathcal{A}$, assuming a uniform distribution.

So, in case $C(x\|y) = |x| \log_2 |\mathcal{A}|$, then $\mathrm{NRC}(x\|y)$ would be equal to one. The more efficiently x can be built from y, *i.e.*, the lower the $C(x\|y)$ and, therefore, the lower the value of $\mathrm{NRC}(x\|y)$. Since $C(x\|y)$ can only be as low as ≈ 0, so does the value of the $\mathrm{NRC}(x\|y)$.

2.4 Finite-Context Models

2.4.1 Definition

A FCM complies to the Markov property, *i.e.*, it estimates the probability of the next symbol of the information source using the $k > 0$ immediate past symbols (order-k context) to select the probability distribution [65]. *Finite-context modeling* has been used for compression in several areas, such as in text and image [66]. Recent work has shown that they also have the ability to measure similarity (or dissimilarity), relying on the data entropy [67, 68, 69]. Therefore, assuming that the k past outcomes are given by $x_{n-k+1}^{n} = x_{n-k+1} \cdots x_n$, the probability estimates, $P(x_{n+1}|x_{n-k+1}^{n})$ are calculated using symbol counts that are accumulated while the information source is processed, with

$$P(s|x_{n-k+1}^{n}) = \frac{v(s|x_{n-k+1}^{n}) + \alpha}{v(x_{n-k+1}^{n}) + \alpha|\mathcal{A}|}, \tag{2.6}$$

where, like on the previous definition, $\mathcal{A} = \{s_1, s_2, \dots s_{|\mathcal{A}|}\}$ is the alphabet that describes the objects of interest, $v(s|x_{n-k+1}^{n})$ represents the number of times that, in the past, symbol $s \in \mathcal{A}$ was found having x_{n-k+1}^{n} as the conditioning context and where

$$v(x_{n-k+1}^{n}) = \sum_{a \in \mathcal{A}} v(a|x_{n-k+1}^{n}) \tag{2.7}$$

denotes the total number of events that has occurred within context x_{n-k+1}^{n}. The parameter α allows balancing between a maximum likelihood estimator and a uniform distribution. Notice that when the total number of events, n, is large, (2.6) behaves as a maximum likelihood estimator [17].

After processing the first n symbols of x, the average number of bits per symbol generated by an order-k FCM is

$$H_{k,n} = -\frac{1}{n} \sum_{i=1}^{n} \log_2 P(x_i|x_{i-k}^{i-1}), \tag{2.8}$$

where we assume the convention that x_{i-k}^{0} is known to both the encoder and the decoder.

2.4.2 Mixtures of Finite-Context Models

It is also possible to use more than one FCM with different orders, in collaboration.

$H_{k,n}$ (equation 2.8) can be viewed as a measure of the average performance of model k until position n [17]. The overall probability estimate for position $n+1$ can be given by the weighted average of the probabilities provided by each model, according to their individual performance, *i.e.*,

$$P(x_{n+1}) = \sum_{k \in \mathcal{K}} P(x_{n+1}|x_{n-k+1}^{n}) w_{k,n}, \tag{2.9}$$

where $\mathcal{K}$ denotes the set of $|\mathcal{K}|$ models involved in the mixture and,

$$w_{k,n} = P(k|x_1^{n}), \tag{2.10}$$

i.e., the weights correspond to the probabilities that each model has generated x_1^n. Hence, we have

$$w_{k,n} = P(k|x_1^n) \propto P(x_1^n|k)P(k), \tag{2.11}$$

where $P(x_1^n|k)$ denotes the likelihood of sequence x_1^n being generated by model k and $P(k)$ denotes the prior probability of model k [70]. Assuming $P(k) = \frac{1}{\mathcal{K}}$, we also obtain

$$w_{k,n} \propto P(x_1^n|k). \tag{2.12}$$

Calculating the logarithm of this probability we get

$$\log_2 P(x_1^n|k) = \log_2 \prod_{i=1}^{n} P(x_i|k, x_1^{i-1}) = \log_2 \prod_{i=1}^{n} P(x_i|x_{i-k}^{i-1}) = \sum_{i=1}^{n} \log_2 P(x_i|x_{i-k}^{i-1}), \tag{2.13}$$

which is related to the number of bits that would be required by model k to represent the sequence x_1^n. Therefore, it is related to the accumulated measure of the performance of model k until position n [70].

To facilitate faster adaptation to non-stationarities of the data, instead of using the whole accumulated performance of the model, we adopt a progressive forgetting mechanism. The idea is to let each model to progressively forget the distant past and, consequently, to give more importance to recent performance results. To accommodate this, we first rewrite (2.13) as

$$\log_2 P(x_1^n|k) = \sum_{i=1}^{n-1} \log_2 P(x_i|x_{i-k}^{i-1}) + \log_2 P(x_n|x_{n-k}^{n-1}) \tag{2.14}$$

and then,

$$\log_2 p_{k,n} = \gamma \log_2 p_{k,n-1} + \log_2 P(x_n|x_{n-k}^{n-1}), \tag{2.15}$$

where $\gamma \in [0, 1)$ is the forgetting factor and $(-\log_2 p_{k,n})$ represents the estimated number of bits that would be required by model k to represent the sequence x_1^n (we set $p_{k,0} = 1$), taking into account the forgetting mechanism [70]. Removing logarithms, we rewrite (2.15) as

$$p_{k,n} = p_{k,n-1}^{\gamma} P(x_n|x_{n-k}^{n-1}) \tag{2.16}$$

and, finally, we set the weights to

$$w_{k,n} = \frac{p_{k,n}}{\sum_{k \in \mathcal{K}} p_{k,n}} \tag{2.17}$$

to enforce normalization [70].

2.4.3 Extended-Alphabet Finite-Context Models

In this section, we introduce a generalization of the FCMs, called xaFCMs, that calculates the probability of occurrence of the next d symbols, given the previous k symbols.

Compressing using extended-alphabet finite-context models

An extended-alphabet finite-context model (xaFCM) complies to the Markov property, *i.e.*, it estimates the probability of the next sequence of $d > 0$ symbols of the information source (depth-d) using the $k > 0$ immediate past symbols (order-k context). Therefore, assuming that the k past outcomes are given by $x_{n-k+1}^{n} = x_{n-k+1} \cdots x_n$, the probability estimates, $P(x_{n+1}^{n+d}|x_{n-k+1}^{n})$ are calculated using sequence counts that are accumulated, while the information source is processed,

$$P(w|x_{n-k+1}^{n}) = \frac{v(w|x_{n-k+1}^{n}) + \alpha}{v(x_{n-k+1}^{n}) + \alpha|\mathcal{A}|^d}, \tag{2.18}$$

where $\mathcal{A}^d = \{w_1, w_2, \dots w_{|\mathcal{A}|}, \dots w_{|\mathcal{A}|^d}\}$ is an extension of alphabet $\mathcal{A}$ to d dimensions, $v(w|x_{n-k+1}^{n})$ represents the number of times that, in the past, sequence $w \in \mathcal{A}^d$ was found having x_{n-k+1}^{n} as the conditioning context and where

$$v(x_{n-k+1}^{n}) = \sum_{a \in \mathcal{A}^d} v(a|x_{n-k+1}^{n}) \tag{2.19}$$

denotes the total number of events that has occurred within context x_{n-k+1}^{n}.

When learning a model from a string, the sequence counts are performed in steps of only one symbol at each time, instead of d symbols. This is done in order to retain as much information as possible from the data source.

Parameter α allows controlling the transition from an estimator initially assuming a uniform distribution to a one progressively closer to the relative frequency estimator.

The theoretical information content provided by the i-th sequence of d symbols from the original sequence x, is given by

$$-\log_2 P(t_i|x_{id-k}^{id-1}) \text{ bits}, \tag{2.20}$$

where $t_i = x_{id}, x_{id+1} \cdots x_{(i+1)d-1}$.

After processing the first n symbols of x, the total number of bits generated by an order-k with depth-d xaFCM is equal to

$$-\sum_{i=1}^{n/d} \log_2 P(t_i|x_{di-k}^{di-1}), \tag{2.21}$$

where, for simplicity, we assume that $n \pmod{d} = 0$.

If we consider a xaFCM with depth $d = 1$, then it becomes a regular FCM with the same order k. In that sense, we can consider that a FCM is a particular case of a xaFCM.

An intuitive way of understanding how a xaFCM works is to think of it as a FCM which, for each context of length k, instead of counting the number of occurrences of symbols of $\mathcal{A}$, counts the occurrences of sequences $w \in \mathcal{A}^d$. In other words, for each sequence of length k found, it counts the number of times each sequence of d symbols appeared right after it.

Even though, when implemented, this might use more memory to represent the model, an advantage is that it is possible to compress a new sequence of length m, relatively to some previously constructed model, making only m/d accesses to the model. This significantly reduces the time of computation, as shown in the experimental results presented in Appendix 5.3.

Since, for compressing the first k symbols of a sequence, we do not have enough symbols to represent a context of length k, we always assume that the sequence is “circular”. For long sequences, specially using small contexts/depths, this should not make much difference in terms of compression, but as the contexts/depths increase, this might not be always the case.

Since the purpose for which we use these models is to provide an approximation for the number of bits that would be produced by a compressor based on them, whenever we use the word “compression”, in fact we are not performing the compression itself. For that, we would need to use an encoder, which would take more time to compute. It would also be needed to add some side information for the compressor to deal with the circular sequences – but that goes out of scope for our goal.

Example

Let x be the circular sequence *AAABCC*. Using a regular FCM with $k = 2$ and $\alpha = 0.01$, we would build the model from Table 2.1 to represent x.

It is easy to notice that this representation can be implemented using an hash-table of strings

Table 2.1: FCM representation of the sequence *AAABCC*.

Context c	$v(A\|c)$	$v(B\|c)$	$v(C\|c)$	$v(c) = \sum_{a\in\mathcal{A}} v(a\|c)$
BC	0	0	1	1
CA	1	0	0	1
AB	0	0	1	1
CC	1	0	0	1
AA	1	1	0	2

Table 2.2: Proposed xaFCM representation of the sequence *AAABCC* (with $d = 1$). Notice that this model has exactly the same information as the one in Table 2.1.

Context c			
BC	C: 1	Total: 1	
CA	A: 1	Total: 1	
AB	C: 1	Total: 1	
CC	A: 1	Total: 1	
AA	A: 1	B: 1	Total: 2

to arrays of integers with fixed size (alphabet size +1). However, we propose a different alternative, which consists of building an hash-table of hash-tables. The reason for doing so is that often the number of counts of symbols for each context is very sparse, which would be a waste of memory. To represent exactly the same model, we would build the structure presented in Table 2.2.

For compressing the sequence x, relatively to itself, we would need $C(x||x)$ bits, where

$$C(x||x) = C(A|CC) + C(A|CA) + C(A|AA) + C(B|AA) + C(C|AB) + C(C|BC)$$

and

$$C(A|CC) = C(A|CA) = C(C|AB) = C(C|BC) = -\log_2 \frac{1 + 0.01}{1 + 3 \times 0.01} = 0.0283$$

and

$$C(A|AA) = C(B|AA) = -\log_2 \frac{2 + 0.01}{1 + 3 \times 0.01} = 1.007$$

, which means $C(x||x) = 2.1272$ or, in other words, it is possible to compress x relatively to the model learned on itself using just 2.1272 bits.

Using a xaFCM, also with $k = 2$ and $\alpha = 0.01$, but with $d = 2$, we would build the model presented in Table 2.3 to represent x.

Therefore,

Table 2.3: Proposed xaFCM representation of the sequence *AAABCC* (with $d = 2$).

Context c			
BC	CA: 1	Total: 1	
CA	AA: 1	Total: 1	
AB	CC: 1	Total: 1	
CC	AA: 1	Total: 1	
AA	AB: 1	BC: 1	Total: 2

$$C(x||x) = C(AA|CC) + C(AB|AA) + C(CC|AB), \tag{2.22}$$

where,

$$C(AA|CC) = C(CC|AB) = -\log_2 \frac{1 + 0.01}{1 + 3^2 \times 0.01} = 0.110 \tag{2.23}$$

and

$$C(AB|AA) = -\log_2 \frac{1 + 0.01}{2 + 3^2 \times 0.01} = 1.049, \tag{2.24}$$

which means $C(x||x) = 1.269$ or, in other words, using a xaFCM to represent the sequence x it is possible to compress it relatively to itself using just 1.269 bits.

Calculating the NRC for both compressors we obtain:

- **Using FCM** – NRC$(x||x) = \frac{2.1272}{6 \times \log_2 3} = 0.224$;
- **Using xaFCM** – NRC$(x||x) = \frac{1.049}{6 \times \log_2 3} = 0.110$.

Based on this example, we can infer that, at least for some cases, it is possible to obtain better compression ratios, using xaFCMs instead of traditional FCMs to represent a sequence.

Parameter Selection

Selection of α

Since adjusting the α parameter might not be trivial, as it depends on the choice of d as well as on the alphabet size. It is, however, possible to choose α based on a certain desired probability p for a specific outcome.

In our experiments, in order to avoid having one more parameter to "tweak", we are defining α automatically, in a way such that, if sequence $w \in \mathcal{A}^d$ was only found once after a certain context $c = x_{n-k+1}^{n} \in x$, and no other sequence $\in \mathcal{A}^d$ was found after that context c (in other words, the total of that line, in the model, is 1), we want to be 90% sure that the same situation happens when compressing a sequence relatively to the learned model. In other words, when we calculate the number of bits,

$$-\log_2 P(t_i|c) \tag{2.25}$$

needed to compress sequence $t_i = x_{id}, x_{id+1} \cdots x_{id+d-1}$, we want to choose an α such that

$$P(t_i|c) = 0.9^d. \tag{2.26}$$

But, since

$$P(w|c) = \frac{v(w|c) + \alpha}{v(c) + \alpha|\mathcal{A}|^d}, \tag{2.27}$$

where $\mathcal{A}^d$ is an extension of $\mathcal{A}$ to d dimensions and c and w were chosen such that $v(c) = v(w|c) = 1$. Therefore,

$$P(w|c) = 0.9^d, \tag{2.28}$$

which is equivalent to,

$$\frac{1+\alpha}{1+\alpha|\mathcal{A}|^d} = 0.9^d. \tag{2.29}$$

Since both the alphabet size $|\mathcal{A}|$ and the depth d are static parameters, it is easy to solve the equation and choose α in this way. It is also worth mentioning that there is always a possible solution for the equation, since the denominator of the fraction on the left is never equal to zero.

Selection of d

The parameter d is an integer greater or equal to one. As mentioned in subsection 2.4.3, when $d = 1$, we are using a xaFCM which is equivalent to a FCM of the same order k. Therefore, they both produce exactly the same number of bits.

As d increases, so does the RAM needed to store the xaFCM model – but there is not much of an impact (for $d = 11$ the increase in memory usage is about 10%). The reason for the model complexity to only increase this is that the number of different "leaves" in the hash-tables does not change with the choice of d – only the size of each string stored does.

Something to take into account when choosing d is that, the greater the value of d, the harder it would be for an arithmetic encoder to complete its process, given that the produced alphabet increases exponentially with value of d. Since we only want to compute the NRC, we do not use an encoder. However, to avoid unrealistic results, we want to choose a d that produces an alphabet size of, at most, the MaximumValue(integer)-1 (e.g. $2^{31}-1$) symbols. For that reason, using an alphabet of size 6, we can say that $1 \leq d \leq 11$.

Often, we are mostly interested in the time it takes to compress a new target sequence, given an already built model representing the reference sequence. With this application in mind, we can say for sure that the d should be as big as possible, since, as mentioned before, less computations need to be done to compress a new target sequence and, therefore, much less time is needed.

2.4.4 Implementation

We have developed an implementation of the extended-alphabet finite-context models, using Python 3.5, which is publicly available under the GPL v3 license at `https://github.com/joaomrcarvalho/xafcm`. This implementation can be used to build the xaFCM models themselves and to compute the number of bits that would theoretically be required to compress a certain string. It also allows certain useful methods like printing details about the models (both memory requirements and tables with the information stored on each model), which might help understand what is happening underneath the implementation, even for someone not familiar with these methods' details.

2.5 Conclusion

In this chapter, we addressed the question of how data compression is related with the notion of the Kolmogorov complexity of strings. Different compression based measures were introduced as a way to approximate the NID, which is not computable. Examples of such measures include the NCD, the NCCD and the NRC. The NRC was introduced recently and is based on the notion of relative entropy – a measure that makes more sense for a "machine learning approach", as the purpose is to measure how something is in relation to a certain baseline (its class), and not the other way around. This approach achieved very good results on different fields and it is the one we are going to use for most of the experiments presented in this document regarding biometric identification using the ECG.

Given that the NRC depends on using a compression model able to perform relative compression, we chose models based on FCMs and have explored both the original FCMs as well as their different variations: collaborative mixtures, substitution tolerant and extended-alphabet. Since the latter, a generalization of FCMs, were introduced during this work, we covered them into more detail and shown that they are useful in cases where there are memory constrains or testing speed is crucial, specially when dealing with large sequences of data.

Before delving into biometric identification using ECG signal, it is important to understand what is ECG signal and how it should be processed – the topic of the next chapter.

Chapter 3

Electrocardiographic Signal

3.1 Overview

With the goal of understanding the processing and data analysis of ECG signals, it is important to have a general idea of what is an ECG signal, how it is collected and what are some of its key points, usually denoted as fiducial points (or points of interest). We start this chapter with those goals. Given that this is not the main topic of the thesis, we do not go into much detail here and just explain the basics, in order to allow the reader to have a general idea of what an ECG represents. We also explain the idea behind the filtering techniques used on the preprocessing of an ECG signal and explain the choices we have done on this regard.

Since in most applications we will be using compression-based tools like the ones defined in Chapter 2, which operate on symbolic data, we will then define the different typical ways to transform from non-symbolic data, like an ECG, to symbolic data – a process defined in signal processing as quantization.

The rest of the chapter is dedicated to a method we have developed, that automatically detects irregularities on an ECG signal, based on quantization.

3.2 Introduction

The organ responsible of the circulation of blood throughout our bodies is the heart. This organ is mostly comprised of muscle, called the myocardium, which periodically contracts and allows for that circulation of blood to occur. There are two main structures on the heart: the auricles, located on the top half of the heart; and the ventricles, located on the bottom half. On the top of the right auricle we can find the sinus node, commonly called the heart pacemaker. This structure is responsible for our heart rates, as it is the one emitting the electrical signals that allow the beginning

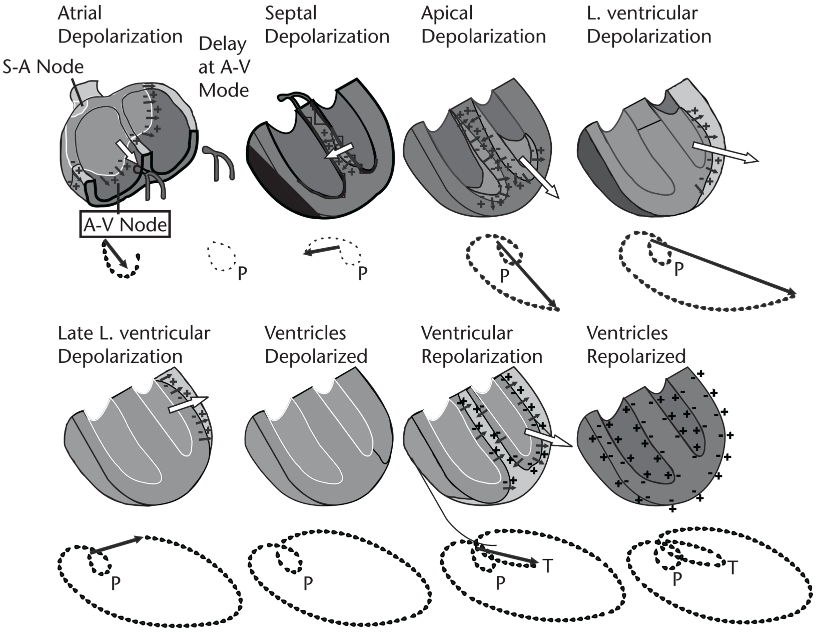


Figure 3.1: Trajectory of a normal cardiac vector (from [71]).

of every normal heartbeat, or systole – this wave of electrical current passes through the entire heart, in a coordinated pattern, which triggers the myocardial contraction. The opposite phenomenon, called diastole, occurs afterwards. This is the period of the relaxation and expansion of the heart chambers between two contractions, when the heart fills with blood [72]. The different phases of this process (see Fig. 3.1) results in measurable changes in potential differences on our body surface. The resultant amplified signal is known as the electrocardiogram (ECG or EKG). This is a typical exam by physicians, part of a complete medical evaluation of a patient, given the importance of the heart on a person's health and also because it is a non-invasive, low cost and virtually risk free exam. For clinical purposes, it is important to notice that ECG signals might be affected by a different number of factors, such as: abnormalities of cardiac conducting fibers, metabolic abnormalities of the myocardium and macroscopic abnormalities of the normal geometry of the heart, making this interpretation sometimes challenging for the clinician [72].

Usually, an ECG signal is collected using 10 electrodes placed on the subject's body: the V_1, V_2, V_3, V_4, V_5 and V_6, placed on very specific regions of the chest (Fig. 3.2 shows the exact placement) and the other four electrodes: the RA, placed on the right forearm or wrist; the LA, placed on the left forearm or wrist; the LL, placed on the left lower leg, proximal to the ankle; and finally, the RL, placed on the right lower leg, proximal to the ankle. It is worth mentioning that even though this process uses only 10 electrodes, it is able to capture a total of 12 leads: lead I, lead II, lead III, lead IV, V_1, V_2, V_3, V_4, V_5, V_6, aVR, aVL and aVF. These leads are collected and recorded simultaneously and

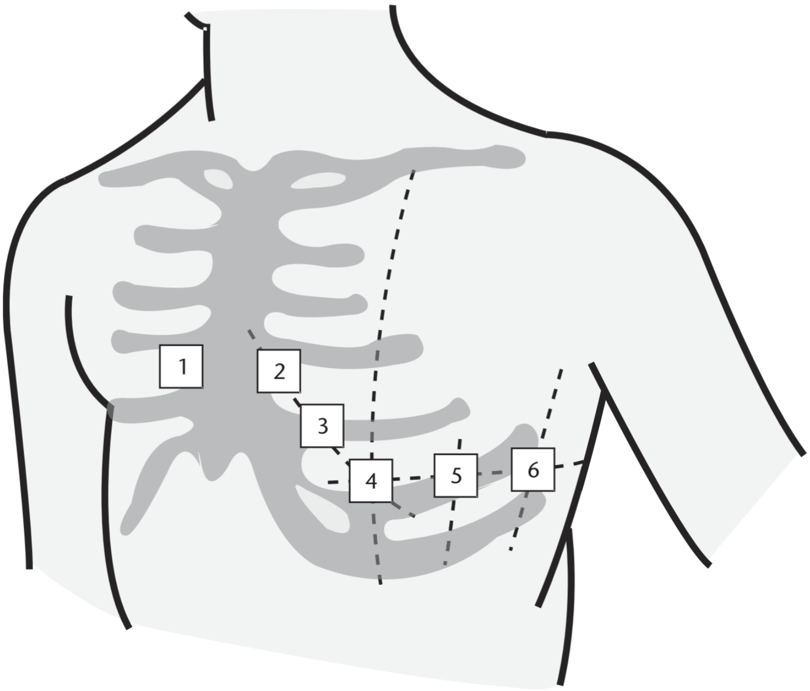


Figure 3.2: The typical lead placement of the six chest leads (v_1, v_2, v_3, v_4, v_5 and v_6) for collecting an ECG (from [71]).

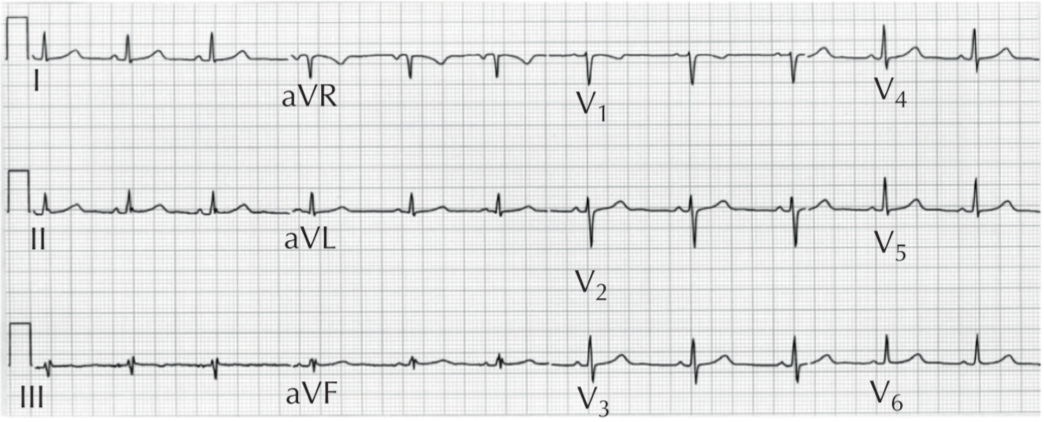


Figure 3.3: A normal ECG recording from an healthy subject, using 12 leads (from [73]).

for clinical interpretations they should be taken into consideration as a whole, as changes in one of them should reflect on others, on most cases. Fig. 3.3, shows a recording of an electrocardiogram from an healthy subject collected on a clinical setting using the approach described.

However, given that our primary goals in this work are not to analyse clinical ECGs, but to perform biometric identification or emotion classification using those signals, the use of on the person sensors and a 12-lead configuration may be too intrusive for daily routine data collection.

Shen and colleagues [74] shown it is possible to identify an individual using only one lead. That setup makes more sense for a biometric identification perspective, as it does not require so many electrodes to be placed on the subject. Therefore, in this work we will be using exclusively one lead. Even in cases where the signal was collected using more than one lead, we choose one and ignore the others.

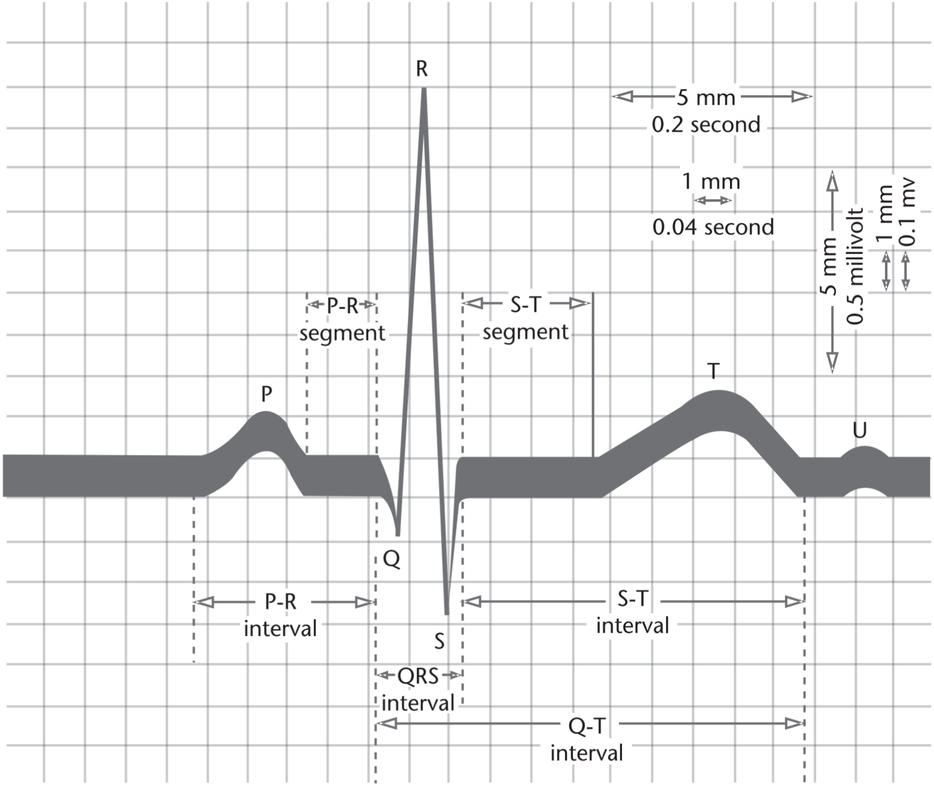


Figure 3.4: Usual features of the electrocardiogram, from an healthy subject (from [71]).

An ECG signal is comprised of different waves and specific points of interest. In Fig. 3.4 it is possible to see some of those points, as well as the intervals between them. The atrial depolarization is reflected on the P-wave, which is usually less than 120ms. The ventricular depolarization of the right and left ventricles is reflected by the QRS complex and lasts around 70 to 110ms, on an healthy person. Inside the QRS complex, we can also mention three waves: the Q-wave, corresponding to the first negative deflection; the R-wave, which is the first positive deflection; and the S-wave, corresponding to the subsequent negative deflection to the R-wave. Beside those waves, there is also the T-wave, which reflects the ventricular repolarization and lasts around 300ms after the QRS complex [75, 76, 77].

There are some intervals between the different fiducial points(P, Q, R, S and T), which may be relevant, depending on the application, but we will only mention the “main one” (as it contains all others) – the RR interval. The RR interval represents the length of a ventricular cardiac cycle, and works as an indicator of ventricular rate. Therefore, it is fundamental in determining a person's heartbeat and, when there is a constant disparity between consecutive RR intervals, allows the diagnosis of arrhythmias [75].

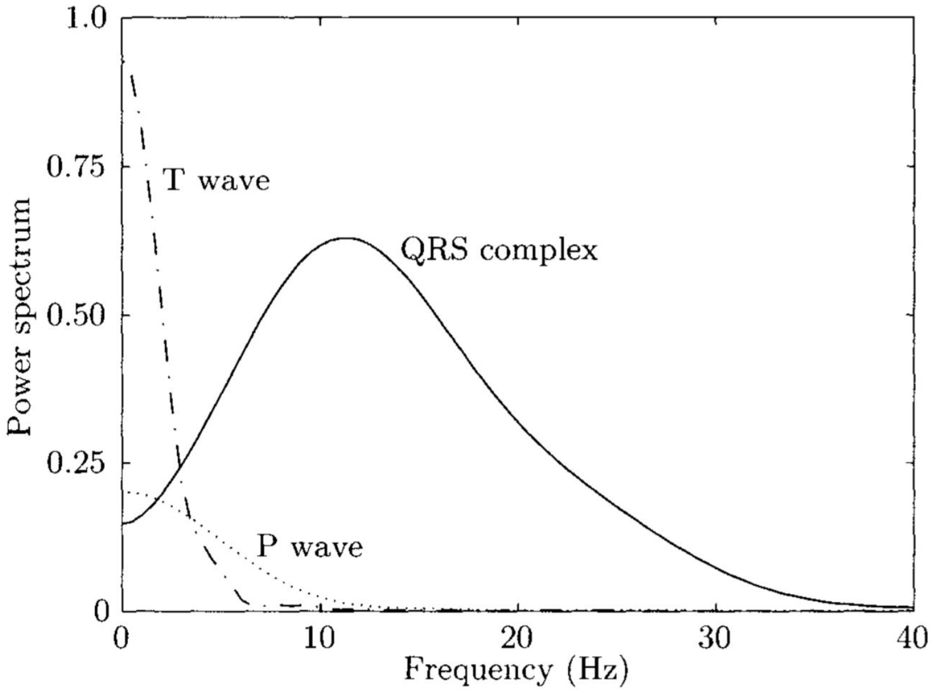


Figure 3.5: Theoretical power spectrum of the P wave, QRS complex and T wave (from [75]).

However, the ECG signal, like any other biomedical signal, is affected by several artifacts and noise. Muscle noise is one of such examples – one for which there is still the need to develop signal processing techniques to deal with [80]. Respiratory noise is also present on the signal, usually on the range between 0.12Hz to 0.5Hz (which corresponds to 8 to 30 times per minute). External electrical interference usually exists in the 50Hz to 60Hz spectrum, while other internal electrical stimuli (like pacemakers with impedance monitoring), are typically above the 10Hz threshold [81].

In order to analyse the signals, it is useful to reduce the amount of noise present in the data.

3.3 Pre-Filtering of the Signal

Most hardware for biomedical data collection include an optional pre-filtering of the signals, while also providing the raw signal. The term "raw signal" refers to a version of the signal in which any digital preprocessing has been applied. This version of the signal will usually include artifacts and noise, making it more difficult to either interpret by a clinician or to use on an automatic pattern recognition system.

On the case of the ECG, depending on the sensors used to acquire the signal, there might be a great difference on the signal quality. Usually, wet gel sensors are more reliable than dry sensors. However, in this work, some databases were collected using dry sensors. The reason for this choice is that the main application of this work is biometric identification. Therefore, we should include a database acquired using a more realistic setup.

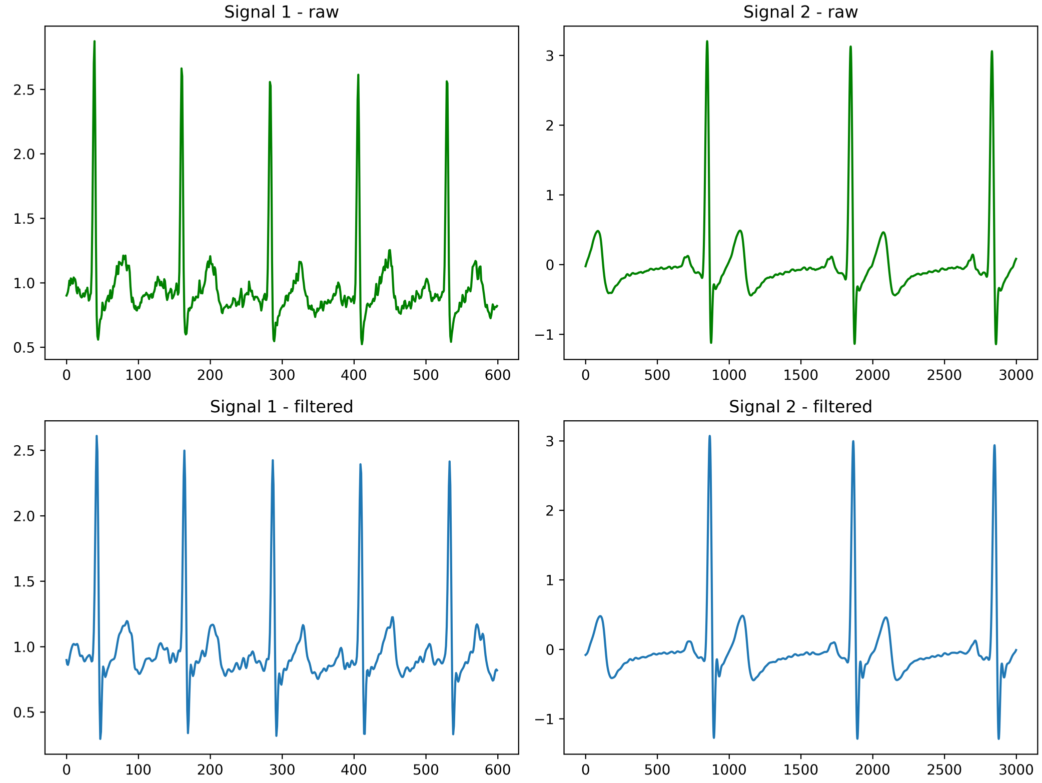


Figure 3.6: Plots of two raw ECG signal segments of four seconds each (in green), as well as their filtered versions (in blue), using a Butterworth low-pass filter of 5th order with a cut-off frequency of 30Hz. Signal 1 (on the left) is an example from the UofT ECG Database (1.3.4) and was collected at 200Hz; signal 2 (on the right) is an example from the Biometric Emotion Recognition Database (1.3.2) and was collected at 1000Hz.

We will not go into details about which different filters can be used for the ECG, as we will follow the choices recommended in the literature for the different applications. But it is important to mention the two main different types of filters that are usually applied to the raw ECG: the *low-pass filters* and the *high-pass filters*.

The low-pass filters allow only low bands to pass while the high-pass filters do the opposite – they allow most information above the selected frequency (called the cut-off frequency) to pass through the filter. The values below the cut-off might be attenuated or removed, depending on the filter used [82]. From these two types of filter we can also derive a third one, which is the usage of both of them together – commonly called a band-pass filter. These systems provide both high and low pass filtering, requiring two cut-off frequencies to be specified: one for the high-pass and another for the low-pass. They are used when it is needed to either remove or reduce both high and low frequencies from the signal – usually to remove noise. In the context the ECG, they are useful for different applications: both for identifying individuals [83]; as well as clinical applications, like heart rhythm monitoring systems [84]. The difference between the upper and lower cut-off frequencies is called the bandwidth of the filter [85].

Figure 3.6 shows an example of a Butterworth low-pass filter of 5th order at a cut-off frequency of 30Hz, applied to two different ECG signal segments: one from the UofT ECG Database (1.3.4), which was collected at 200Hz using dry electrodes; and another from the Biometric Emotion Recognition Database (1.3.2), collected at 1000Hz using wet gel sensors. It is interesting to notice that on the first one the differences between the raw signal and its filtered version are quite obvious, as the original signal had very noticeable fluctuations, caused by noise. On the second signal, which was collected using a more intrusive, but less prone to noise setup, the differences are more subtle, but the filtered signal is a *smoothed* version of the raw signal.

3.4 R-Peak Detection

The development of a robust automatic *R-peak* detector is essential for many tasks related with ECG signals, but it is still a challenging task, due to irregular heart rates, various amplitude levels and shapes of *QRS* morphologies, as well as all kinds of noise and artifacts [86].

We have decided to use a *partially fiducial* method for segmenting the ECG signal and, since this was not the major focus of the work, we used a preexisting implementation to detect *R-peaks*, based on [86]. This method detects the *R-peak* by calculating the average point between the *Q* and *S* points (the *QRS complex*) – this may not give the real local maximum of the *R-peak*, but it produces a very close point.

The process used for detecting the *QRS* complexes is somewhat similar to the one described in [86]. It uses some bandpass filtering and differentiation operations used to enhance *QRS* complexes and to reduce out-of-band noise. A nonlinear transformation based on energy thresholding, Shannon energy computation, and smoothing processes is used to obtain a positive-valued feature signal which includes large candidate peaks corresponding to the *QRS* complex regions.

For more information regarding the process used for detecting the R-peaks check [86]. The process was already validated by its authors using the standard MIT-BIH arrhythmia database, achieving an average sensitivity of 99.94% and a positive predictivity of 99.96%. It uses bandpass filtering and differentiation operations, aiming to enhance the *QRS* complexes and to reduce out-of-band noise. A nonlinear transformation is used to obtain a positive-valued feature signal, which includes large candidate peaks corresponding to the *QRS* complex regions.

3.5 Quantization

3.5.1 Introduction

Quantization is widely used in signal processing. It is a process that takes a signal (or a numerical time-series) and produces only a, usually predefined, discrete set of values. It is a very simple process. However, the design of the *quantizer* has a significant impact on the amount of compression obtained and loss incurred in a lossy compression scheme [1].

A *quantizer* consists of two opposite mappings: an encoding mapping, which splits the range of values that the source generates into a number of intervals – each of them represented by a symbol (or code); and a decoder mapping, which maps from the quantized values into an attempt of reconstructing the original signal. This process is *lossy* (loses some of the original information), as there could be many distinct sample values that can fall in any given output interval. Knowing the interval does not tell us which of the many values in the interval is the actual sample value - the decoder can only output a value that, in some sense, best represents all the values in the interval [1].

We will refer to the number of discrete intervals of the output in a quantization as the *alphabet size* – in other words, the maximum number of different symbols produced by the quantizer. There is a fundamental trade-off to take into account while performing the choice of the *alphabet size*: the quality produced versus the amount of data necessary to represent the sequence [87] – if the alphabet size is higher, both the quality and the amount of data necessary to represent the original sequence will usually increase.

3.5.2 Uniform Quantization

The simpler quantizers are known as *zero memory quantizers*, in which the quantization of a sample is independent of other samples. The signal amplitude is simply represented using a finite number of bits independently of the sample length. One example of a *zero memory quantizer*, that is in fact the simplest one, is the *uniform quantizer*, in which the transition and reconstruction levels (usually referred as *breakpoints*) are all equally spaced [87]. This is a very simple approach that is useful when the random variable being quantized follows a uniform distribution – however that is not usually the case.

Unfortunately, such a quantizer, when applied to ECG signals, for most applications, would not be appropriate. An easy way to prove this point is to remember that different regions of the ECG have changes in different proportions – *e.g.*, the region around the R-peak on the ECG changes a lot, while in other regions such as the the P-wave, the important changes are subtle. Using a zero memory quantizer on the P-wave, those small changes would probably be *lost* during the quantization, since they could be transformed into the same symbol.

3.5.3 Piecewise Aggregate Approximation

The original Piecewise Aggregate Approximation (PAA) method [88], performs a re-sampling in the time series, reducing its length from the original length n to w samples. The idea is to split the data into w intervals of the same length. The mean value of each of the intervals is then calculated and each value inside of that interval is replaced by that result. An example can be seen in Fig. 3.7.

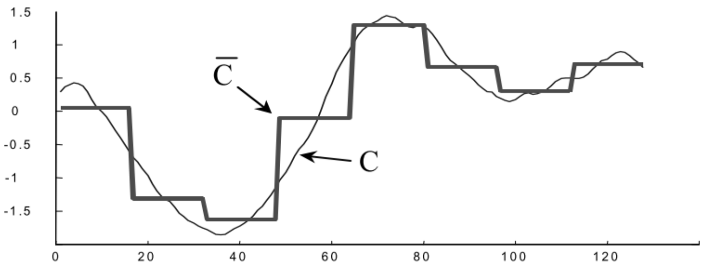


Figure 3.7: PAA method applied to a time series (from [15]).

3.5.4 Nonuniform Quantization

When the random variable to be quantized does not follow a uniform distribution, a *nonuniform quantization* should be performed. In order to decrease the average error of the quantization, we can try to approximate the input better in regions of high probability, perhaps at the cost of worse approximations in regions of lower probability. We can do this by making the quantization intervals smaller in those regions that have more probability of occurrence [1].

If we want to keep the number of intervals constant, this would mean we would have larger intervals in the low probability regions, and small intervals in high probability regions. A quantizer that has nonuniform intervals is called a *nonuniform quantizer* [1].

While a nonuniform quantizer provides lower average distortion/error, the design of nonuniform quantizers is also somewhat more complex. However, the basic idea is to find the decision boundaries that minimize the mean squared quantization error, given by

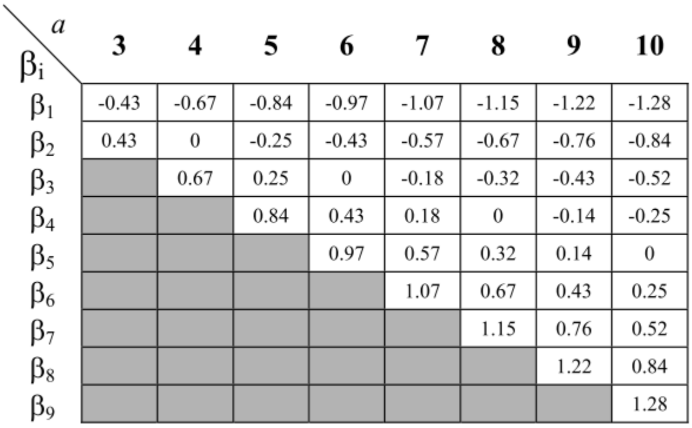
$$\sigma_q^2 = \int_{-\infty}^{\infty} (x - Q(x))^2 f_X(x) dx, \tag{3.1}$$

where X is the variable being quantized, $f_X(x)$ is the probability density function of X and $Q(x)$ is the quantization of value x. Two of such examples are the symbolic aggregate approximation [15] and the Lloyd-Max quantization [89, 90].

Symbolic Aggregate Approximation (SAX)

The SAX method assumes that the original sample distribution follows a Gaussian distribution [91]. The main idea behind the method is to use normalized values in order to produce symbols with equiprobability. This is done by predefining *breakpoints* based on the amount of symbols that we want to use to represent our series. The symbols used to represent the time series are, usually, the first letters of the alphabet (**a**, **b**, **c**, etc.).

The *breakpoint* values come from statistics about the Gaussian distribution itself, in order to produce equal sized areas under Gaussian curves [92]. Example breakpoint values for some alphabet sizes can be seen in Fig. 3.8.



a β_i	3	4	5	6	7	8	9	10
β_1	-0.43	-0.67	-0.84	-0.97	-1.07	-1.15	-1.22	-1.28
β_2	0.43	0	-0.25	-0.43	-0.57	-0.67	-0.76	-0.84
β_3		0.67	0.25	0	-0.18	-0.32	-0.43	-0.52
β_4			0.84	0.43	0.18	0	-0.14	-0.25
β_5				0.97	0.57	0.32	0.14	0
β_6					1.07	0.67	0.43	0.25
β_7						1.15	0.76	0.52
β_8							1.22	0.84
β_9								1.28

Figure 3.8: Lookup table with the *breakpoints* that divide a normalized Gaussian distribution in an arbitrary number (from 3 to 10) of equiprobable regions (from [91]).

After defining the *breakpoints*, the method obtains the PAA of the time series. The values which are below the smallest breakpoint are mapped to the first letter, **a**. All coefficients greater than or equal to the smallest breakpoint and less than the second smallest breakpoint are mapped to the second letter, **b**, and so on [91]. Fig. 3.9 contains an example of the method applied to a time-series.

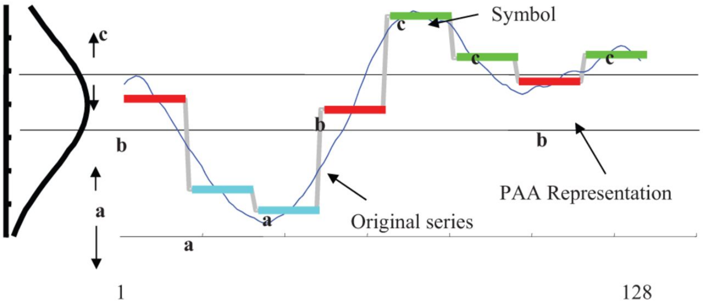


Figure 3.9: SAX (letters) and PAA (lines) methods applied to a time series (from [93]).

A more in-depth explanation about SAX, from its original authors, can be found in [91, 15].

Lloyd-Max Quantization

Stuart P. Lloyd and Joel Max, independently proposed an algorithm to compute optimum quantizers using the mean-square error distortion measure in 1982 [89] and 1960 [90], respectively. The algorithm, later called Lloyd-Max algorithm (or quantizer), is still widely used in practice because it can be easily implemented [94] and has some interesting properties [1]:

- The mean values of the input and the output of a Lloyd-Max quantizer are equal.
- For a given Lloyd-Max quantizer, the variance of the output is always less than or equal to the variance of the input.
- The mean squared quantization error for a Lloyd-Max quantizer is given by

  $$\sigma_q^2 = \sigma_x^2 - \sum_{j=1}^{M} y_j^2 P[b_{j-1} \leq X < b_j], \tag{3.2}$$

  where σ_x^2 is the variance of the quantizer input, and the second term on the right-hand side is the second moment of the output (or variance if the input is zero mean).
- If N is the random variable corresponding to the quantization error, then for a given Lloyd-Max quantizer,

  $$E[XN] = -\sigma_q^2. \tag{3.3}$$
- For a given Lloyd-Max quantizer, the quantizer output and the quantization noise are orthogonal:

  $$E[Q(X)N|b_0, b_1, \cdots, b_M] = 0. \tag{3.4}$$

In sum, the Lloyd-Max is useful when the distribution of the variable to quantize is given by some complex mathematical function. However, in our applications, it is possible to obtain a normalized Gaussian distribution – as we will see.

3.5.5 Quantization Applied to ECG

There are many other techniques for quantization, but we covered some of the more commonly used in the literature. We show in the next sections some real applications where we used different methods. Since the ECG signal suffers from baseline wander, we have chosen to operate on its consecutive differences, instead of on the original signal, and use a Lloyd-Max quantization scheme on top of the signal. As mentioned, the idea is that the quantizer learns breakpoints so that each symbol on the output will have approximately the same number of samples, meaning that regions with higher probability will be lower in interval range and vice-versa. The quantizer then replaces each of the original samples by the symbol corresponding to the interval it belongs to, meaning that a sequence of floating numbers becomes a sequence of symbols (in our implementation[1], capital letters). An example of such a quantization can be seen in Fig. 3.10.

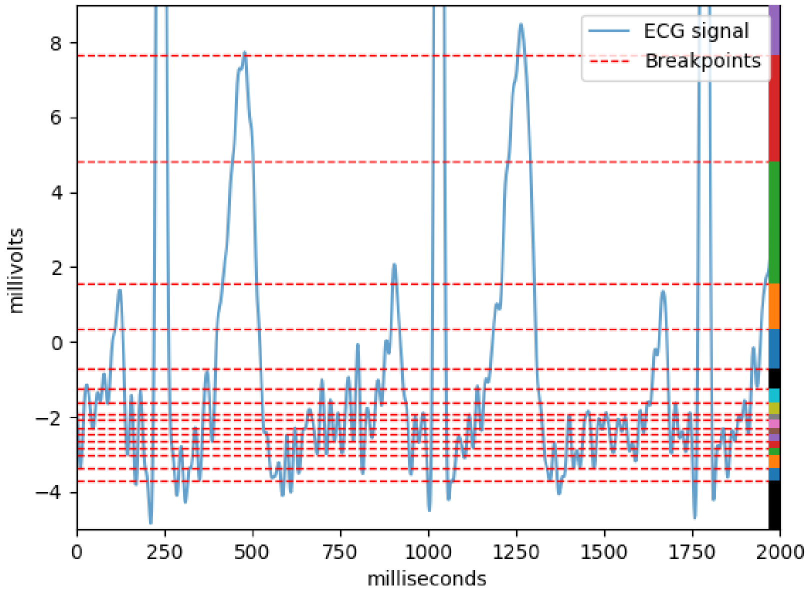


Figure 3.10: Quantization scheme example on two seconds of ECG signal from the dataset described in 1.3.5. The ECG signal is represented with a blue line and the red dashes represent the breakpoints learned by the Lloyd-Max quantizer for this two second interval. The colors on the right are merely illustrative to help visualise the different intervals. Each interval represents a different symbol used after the quantization is done, *i.e.*, when each sample on the signal is replaced by the corresponding symbol from the quantization interval it belongs to.

3.6 Irregularity Detection in ECG Using a Semi-Fiducial Method

3.6.1 Introduction

It is supposed that, at rest, the ECG signal from a complete cardiac cycle is similar to the previous and to the next cycle. However, due to external or internal interferences, this may not be true [95][96][17].

Developing an algorithm to identify where those interferences occur may be of great interest for biometric identification, as well as other applications, as it may allow to incorporate that algorithm into a decision support system, where parts of the signal that contain irregularities may be treated differently than regular signal.

[1] The source code for the quantizer was implemented from scratch using Python 3.7 and we made it openly available on `https://github.com/joaomrcarvalho/diffquantizer`.

3.6.2 Overview

In [3], the authors describe a method for finding noise using compression tools. They do this by computing the dissimilarity distance of a given segment of an ECG against the whole ECG. The Compression-based Dissimilarity Measure (CDM) between two string x and y [3] is defined as:

$$\mathrm{CDM}(x, y) = \frac{C(xy)}{C(x) + C(y)}, \tag{3.5}$$

where xy represents the strings x and y concatenated and $C(x)$ the number of bits required to compress x.

Two important facts about the CDM are: **(a)** it is close to one when x and y are not related; **(b)** if x and y are related, as strongly related they are, the lower the $\mathrm{CDM}(x, y)$ is, but it never reaches zero.

In short, what their method does is to measure how well a small local section can match the global sequence [3], which can be useful to find irregularities in the signal (assuming they are present only in some small portions of the signal).

Since the NRC has been shown to work very well on ECG signals [17] and it also respects both **(a)** and **(b)**, we did an implementation using it as a replacement for the CDM defined by Keogh *et al.* Finally, we compared that approach with our proposed method.

3.6.3 Database

Despite the fact that the proposed algorithm aims at the detection of ECG noise in real signals, we had to manage a control set of ECG signals to test and validate the algorithm. These were already acquired on a previous study.

The dataset used for this work was the Vital Jacket DB, described in detail in 1.3.1. The essential information is that this dataset's collecting protocol consisted in gradually increasing and decreasing the heart rate with 20 minutes interval and noise values were randomly generated using a uniform discrete distribution between the minimum and maximum values (117 and 159, respectively) of the collected signal. This allowed us to have a controlled signal with noise in specific zones, to test the algorithm's behavior in different situations.

3.6.4 Quantization

In order to convert the real-valued ECG signal into a symbolic time series, the first step we had to perform was to reduce its dimensionality. This was achieved by using a modification of the Piecewise Aggregate Approximation, PAA, method [15].

Since, in our case, the *R-peaks* of the ECG signal were already detected, we used this to our advantage and, instead of splitting the complete signal into w dimensions, as the original method suggests, we applied it only inside of each *RR-interval* (intervals between consecutive *R-peaks*) individually, with the values inside each of those intervals previously normalized. This means that, for example, if one heartbeat[1] takes more time than another, then the real value obtained by PAA may correspond to more real values on the original signal. On our case, the number of samples, w, chosen for each heartbeat was 200. This means that each heartbeat on the ECG, on this phase, was represented by 200 real values, independently of how many samples that heartbeat contained originally. As an example, an instantaneous heart rate [97] with 60 beats per minute would have one second per heartbeat. Since the signal was collected at 1000 Hz, this would correspond to 1000 samples, which would be replaced by its' average for every 5 samples interval it contained.

After completing this process for all the heartbeats of a signal, since the purpose is to have a symbolic representation of the series – not a real valued one, the *Symbolic Aggregate approXimation* (SAX) was applied to each heartbeat's *PAA* series individually.

From this explanation, it is already implicit that one parameter used by this method as input is the *alphabet size* (the number of different symbols allowed as output), that we want to use. After performing several experiments, it was decided to use an alphabet size of six, as it showed a good trade-off between performance and complexity of the model – based on empirical tests, using more than six symbols, the performance of the method did not improve (at least significantly), while the method was computationally more costly. Using the process described, each complete heartbeat is outputted as a 200 length string. We refer to a string like that as a *word* or *SAX-word*.

3.6.5 Proposed Method

The proposed *Simple Distance method* consists on the *Hamming Distance* applied to consecutive *SAX-words*.

The idea of our approach is to store all the n *words* (or *SAX-words*) of an ECG on a size n array and compute the $n-1$ distances between those consecutive n words. Since we want to have a sample of size n, an interpolation from size $n-1$ to size n should be performed. After that, some decisions can be made using the values obtained for that metric.

3.6.6 Discussion

From our experiments, we found that a threshold which produced more consistent results for both metrics was $\bar{x} - 2\sigma$ (average value minus two times the standard deviation). We were able to do it by experimentation, because we knew the zones where noise was supposed to be found beforehand.

[1] The term "heartbeat", in this context, refers to a RR-interval.

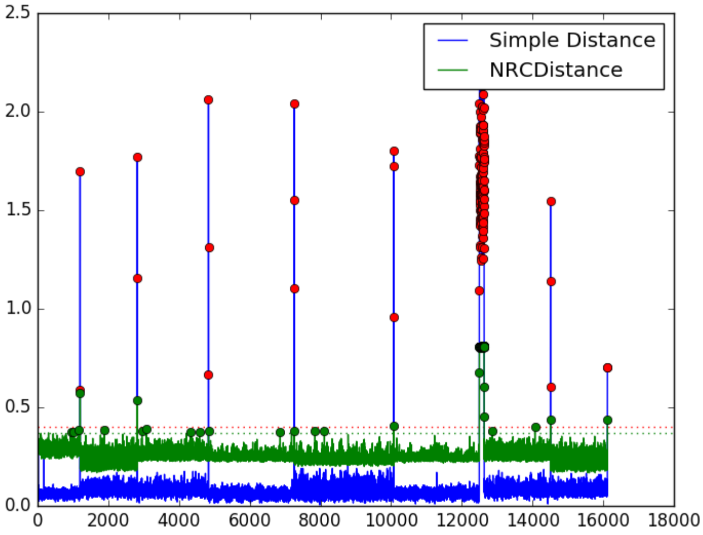


Figure 3.11: The transition between 120 and 100 beats per minute was replaced by 90 seconds of noise (signal number 5).

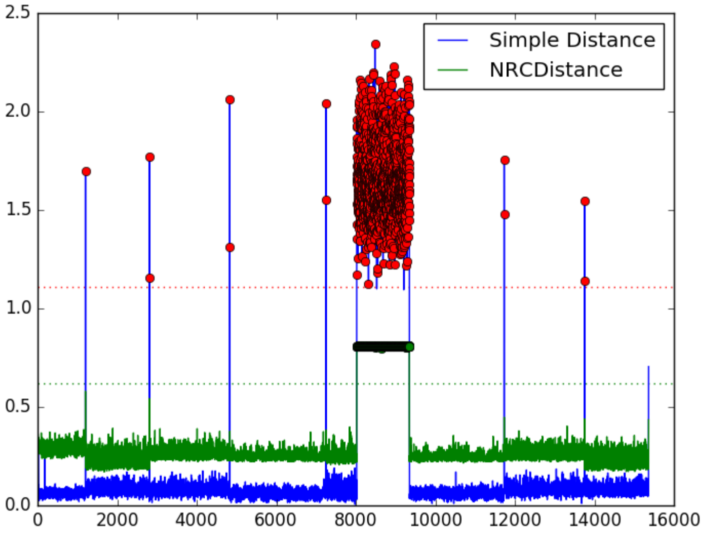


Figure 3.12: In the 140 heart rate step, 15 minutes of noise were introduced (signal number 3).

In Fig. 3.11, it is possible to see that the *Simple Distance* measure detects the areas where random noise was inserted very precisely, while the NRC detects a lot of false positives. In Fig. 3.12, the NRC is not able to detect any noise at all, however, the *Simple Distance* can detect both the zones where the heart beat rate was changed, which, even tough it is not noise, may be considered a point of interest, depending on the application. This method outperforms an alternative proposed by Keogh *et al.* in [3], at least for this specific dataset (dataset 1.3.1).

Even though the threshold choice worked properly for this dataset, it should not be static and, therefore, some future work can be done in order to adjust it in a dynamic way. It can also make sense to perform further tests using different datasets, as well as possible changes to the method itself, namely, by using different quantization methods besides SAX for the initial transformation of the numerical series into symbols.

3.7 Final Remarks

In this chapter, we have mentioned some techniques commonly using for the preprocessing of ECG signals, both for clinical purposes and for other applications, such as emotion classification or biometric identification. ECG signal processing is a field of research on its own, so it would be impossible to cover everything, but we focused on introducing the basics to the understanding of the work as a whole.

Regarding quantization, we also covered the most common quantization techniques used for working with biomedical signals (not exclusively the ECG) and have shown some real applications where we used those methods to obtain useful results. As we have seen, different techniques might be useful depending on the application at hand, so we will use different quantizers from the ones covered during the rest of the thesis.

In the next chapter, we will introduce the field of ECG biometrics and use several techniques from all the previous chapters from this thesis, in order to present some proposals for biometric identification systems, based on compression.

Chapter 4

ECG Biometrics

4.1 Introduction

To understand pathological characteristics of the ECG, in clinical practice, it is usual to try to reduce the subject-dependent inter-variability that characterizes the signal. This inter-variability is precisely the source of richness that renders the ECG an interesting signal for biometric applications. Because of its desirable characteristics (universality, uniqueness, measurability, acceptability and circumvention avoidance [98]), it is worth exploring as an alternative to replace or aid other biometric identification methods already commonly used nowadays, such as face recognition [99], fingerprint recognition [100] or iris recognition [101], speaker identification using voice [102], gait recognition [103], among other less common or more intrusive methods, namely: retina recognition [104], hand geometry recognition [105], typing recognition [106] and signature recognition [107].

In this chapter, we aim at exploring that ECG inter-variability between individuals using methods based on compression algorithms and, more specifically, on finite-context model based compressors, in order to propose a possible solution for biometric identification using the ECG. To reach that outcome, several techniques described in the previous chapters will be used, both to prepare and transform the signals, as well as to analyse and interpret the results obtained, using the NRC relative dissimilarity measure.

In order to understand the nuances of such a system, we have performed studies regarding different parts of both the data collection protocol and the validation step of the proposed approaches. These studies include experiments regarding how much time is required, on average, to correctly identify someone or understand how *noisy* segments inside a sample of ECG influence the models learned by such a system.

4.2 The Impact of Noise Removal on a Compression-based ECG Biometric Identification System

4.2.1 Introduction

ECG signals reflect an individual's cardiac electrical activity over a period of time. It has the advantage of being a unique aliveness indicator as it is difficult to be spoofed and falsified [109], which makes it desirable for biometric authentication purposes. However, this signal is prone to irregularities that are originated from several sources: pathological, psychological, noise, artifacts, among others [95, 96, 110].

For that reason, most ECG applications achieve better results when the signal is cleaned, i.e., with noise removal. In this Section, we have extended a method presented in Chapter 3, based on the *Hamming Distance*, that has proved able to find noise on an ECG signal. We study its effect in the accuracy results while performing ECG biometric identification, using a compression-based approach. This method was only tested in synthetic ECG data. We aim at extending that work, by using real data, and exploring the consequences of noise removal on a compression-based biometric identification system. The method is then tested by using finite-context models (FCM) of different context depths k, as well as a mixture of FCMs for building the models to represent each individual.

4.2.2 Database

The database used in the experiments, was collected *in house* [17], where 25 participants were exposed to different external stimuli – *disgust*, *fear* and *neutral* emotional conditions, inducing external variability to the data. The data signals were collected on three different days (once per week), at the University of Aveiro, using a different stimulus per day. A more detailed description about this dataset can be found in 1.3.2.

4.2.3 Method

Even though the signal is already discrete in the time domain, i.e., it is already sampled, we perform re-sampling using the previously detected R-peaks, using the method described in Section 3.4, based on [86].

For the quantization, we have used the Symbolic Aggregate ApproXimation (described in 3.5.4) [15], SAX, in order to quantize the ECG values into a discrete alphabet. There is a fundamental trade-off to take into account while performing the choice of the *alphabet size*: the quality produced versus the amount of data necessary to represent the sequence. From previous experiments, we found that using an alphabet size of 6 and 200 symbols per each R-R segment (per "heartbeat")

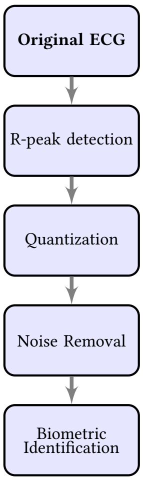


Figure 4.1: Overview of the different steps of the method proposed for biometric identification with noise removal.

produced good results for biometric identification. However, this result does not guarantee that the same will hold true for a different dataset or application, given that the data might have been collected under another protocol and other applications might require different resolutions for the quantization.

4.2.4 Noise Removal

We wanted to propose a simple approach, that could be implemented on an embedded device, for real applications, for detecting noise "on-the-fly". The idea of the proposed approach is to store all the n *words* (*SAX-words*) of an ECG on a size n array, compute the $n-1$ Hamming-distances between those consecutive n *words* and remove the *words* that correspond to an Hamming-distance greater than $\bar{x} + \delta\sigma$, were δ is a parameter of the method.

The distance i is given by the distance of the word i to the word $i+1$ (see Fig. 4.2) and, therefore, if it is greater than the threshold, both words i and $i+1$ are removed.

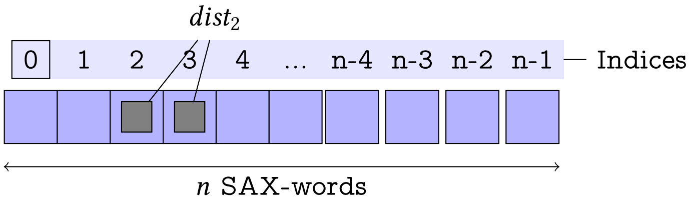


Figure 4.2: Array representation of SAX-words used to calculate the Hamming-distances.

4.2.5 Parameter Tuning

In order to tune the parameter δ, we ran nearly 100 simulations, changing the parameter from 0.5 up to 3. Since the purpose of this experiment was only to tune the parameter, and not to obtain an optimal biometric identification accuracy, we used an extended-alphabet FCM based compressor (xaFCM) [36].

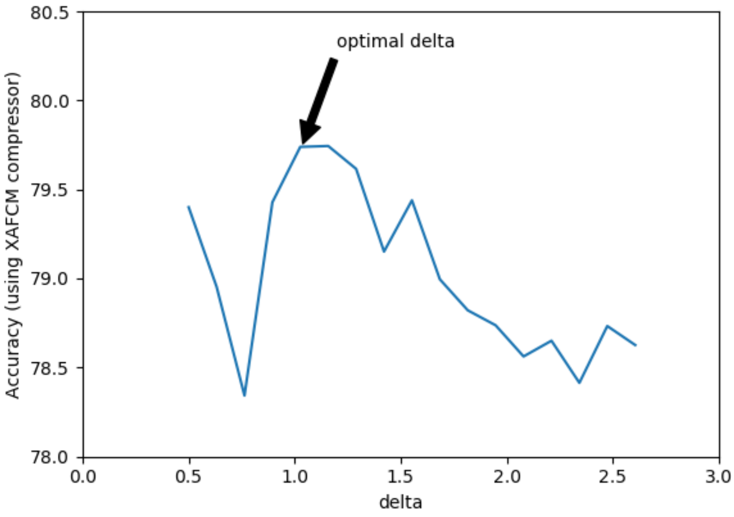


Figure 4.3: Parameter δ tuning.

Because of this result, whenever we use “noise removal” on the results section, a value of $\delta = 1.1$ was used.

4.2.6 Results and Discussion

In order to simulate a real biometric identification system, whenever we test the system, we always reserve one session for testing the data, using only the other two sessions as reference (training). All tests were performed using only 10 heartbeats as target (test), also to simulate a real scenario, where the identification should take place in a reasonable time frame.

From the results shown in Table 4.1, it is possible to notice that the noise removal slightly enhances biometric identification when using a single context FCM, specially when that context k is low ($k = 10$ or $k = 13$). For high order FCMs, the noise removal does not seem to have much impact ($k = 20$) and, when using a mixture of FCMs, it even seems to be counter productive – which makes us think that these type of models are highly robust to noise, being able to capture some useful information, even when great amounts of noise are present. More details regarding these, at first, non-expected results, can be found in the final discussion of the thesis.

Table 4.1: Biometric identification accuracy using different sizes of finite-context models, with noise removal (NR) and without removal (WR). The Mixture used the contexts $k = 2, 4, 8, 12, 16, 20$ with $\alpha = 1, 1, 0.5, 0.1, 0.1, 0.001$, respectively, with a forgetting factor = 0.99.

	Day for Target		
FCM context(s)	Day 1	Day 2	Day 3
k = 10 (WR)	77.94%	78.78%	78.22%
k = 10 (NR)	79.32%	80.31%	77.60%
k = 13 (WR)	79.70%	82.32%	79.10%
k = 13 (NR)	80.68%	82.72%	79.36%
k = 16 (WR)	79.93%	84.33%	80.76%
k = 16 (NR)	80.60%	83.73%	81.87%
k = 20 (WR)	79.36%	84.61%	81.79%
k = 20 (NR)	80.08%	84.36%	82.11%
Mixture (WR)	82.45%	84.98%	83.13%
Mixture (NR)	83.36%	84.93%	81.96%

4.3 Impact of the Acquisition Time on an ECG Biometric Identification System

4.3.1 Introduction

Based on the current literature review, there are no results that evaluate the recommended number of heartbeats to be used for personal identification using ECG signal. This information is undoubtedly useful when building a biometric identification system, as a good acceptability rate when identifying a person for biometry should be fulfilled in as short time as possible. For that, we need to evaluate the minimal number of heartbeats that is needed to be collected.

In this section, we aim at exploring this topic using the NRC relative similarity measure, with the use of FCM based compressors to represent each individual on the database.

4.3.2 Database used

The database used was the same as the previous experiments, where signals were collected from 25 participants on three different sessions. The details for the data collection can be found in 1.3.2, but it is important to mention that during the preparation phase, the adhesive disposable Ag/AgCL-electrodes were fixed in the right hand, as well as in the right and left foot. We are aware that such an intrusive set-up is not desirable for a real biometric identification system. However, for testing purposes, it is appropriate, as this approach is more reliable – produces less noise.

4.3.3 Method

Before any attempt was made, in order to extract useful information for the compressors, the signal was pre-processed using a low-pass filter and quantized using SAX [15], similarly to the previous section. We perform re-sampling using the detected R-peaks, with an alphabet size of 6 and a total of 200 symbols per R-R segment (per heartbeat).

The general workflow can be seen in Fig. 4.4.

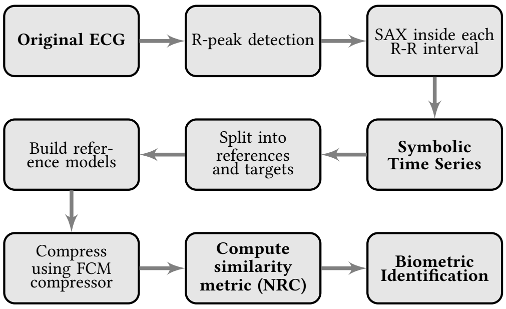

Original ECG
R-peak detection
SAX inside each R-R interval
Symbolic Time Series
Split into references and targets
Build reference models
Compress using FCM compressor
Compute similarity metric (NRC)
Biometric Identification

Figure 4.4: High-level overview of the Biometric Identification process used.

4.3.4 Experimental Results

Since memory usage was not a concern for these preliminary tests, we computed all possible context orders k from 1 up to 40. Theoretically speaking, the number of possible contexts found by a FCM of $k = 40$, with an alphabet size of 6, would be 6^{40} (higher than 10^{31}). However, using hash tables, we do not need to compute all those combinations. In fact, from all the contexts that we computed, no model used more than 10^5 different contexts – different participants have a tendency to produce different contexts, which is in fact what we exploit in order to distinguish amongst them.

In the models' design, the algorithm takes some time (around one second). However, in testing, they are characterized by being fast. After the model is built, we found that a regular computer can run hundreds of similarity evaluations per second, using just one core[1]. In other words, when a small ECG signal is collected, we can obtain the similarity measures to hundreds of models (one model per participant, in the case of biometric identification) that are previously built in our database and loaded into RAM. Another important factor is that this process is easily parallelized, which means that this computation can scale as much as the hardware allows it.

As mentioned in [110], there is an intra-variability for each participant from one day to another, which makes the biometric identification more challenging. The tests (**a**) Day 1, (**b**) Day 2 and (**c**) Day 3 contained no information of the ECG being tested when building the models for each participant.

[1] All the experiments were done using Python 3.5 on an Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz, with 32GB of RAM.

Test (**d**) was performed using the baselines of all days (the first 350 heartbeats) and then by running the biometric identification tests using different segments from all days.

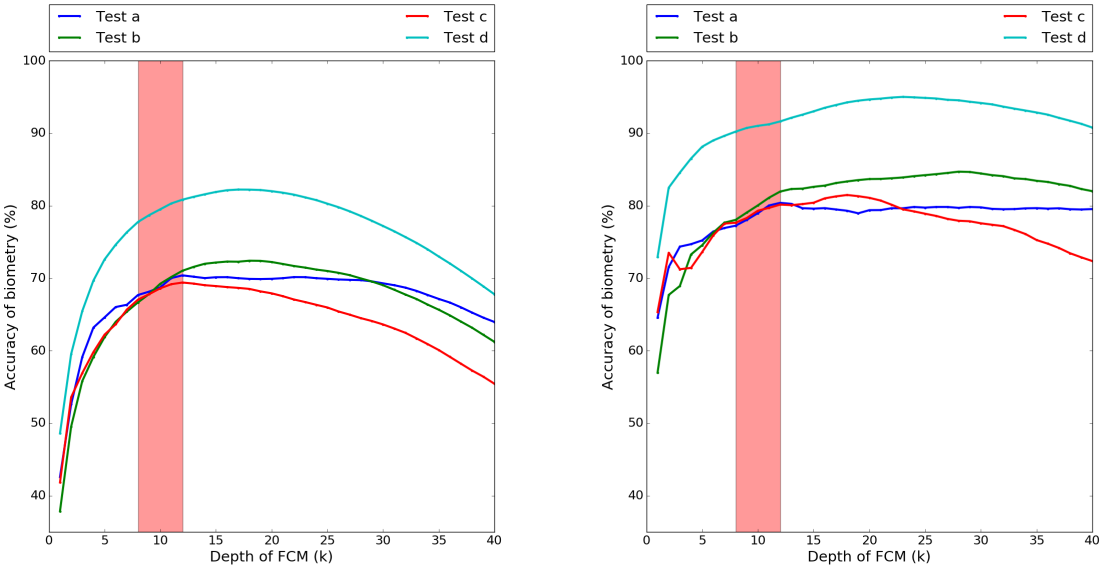


Figure 4.5: Comparison of accuracies obtained using test segments with (**left**) 1 heartbeat and (**right**) 10 heartbeats, for all possible contexts $k \in \{1, \ldots, 40\}$. The area in red represents what we consider the best choices for k, taking into account the complexity of the models produced.

Given that the goal of this work was to measure the minimal number of heartbeats in which it was possible to identify subjects, even in situations where they are under the effect of fear or disgust, we tried different setups: using from just one heartbeat for biometric identification, up to twenty heartbeats, in order to see the differences obtained in accuracy. The reason for choosing accuracy as a performance metric had to do with the fact that our data is almost perfectly balanced, given the protocol used to collect the data (see 1.3.2) *i.e.*, we have the same amount data for each participant (class).

In Fig. 4.5, it is possible to observe clear gains in performance from using one heartbeat to using ten heartbeats, for all possible contexts $k \in \{1, \ldots, 40\}$.

However, in order to have a more accurate way of choosing the "ideal" number of heartbeats to collect for testing, we show a plot with the maximum performance obtained for each of the tests ran in Fig. 4.6. It is possible to see that the performance does not improve clearly when using more than a certain number of heartbeats for testing (area marked in red). In fact, the only test which accuracy does continue to increase is test (**d**), which is the only one that includes the baseline for all the days – even the one we are testing, which does not simulate a real world biometric identification.

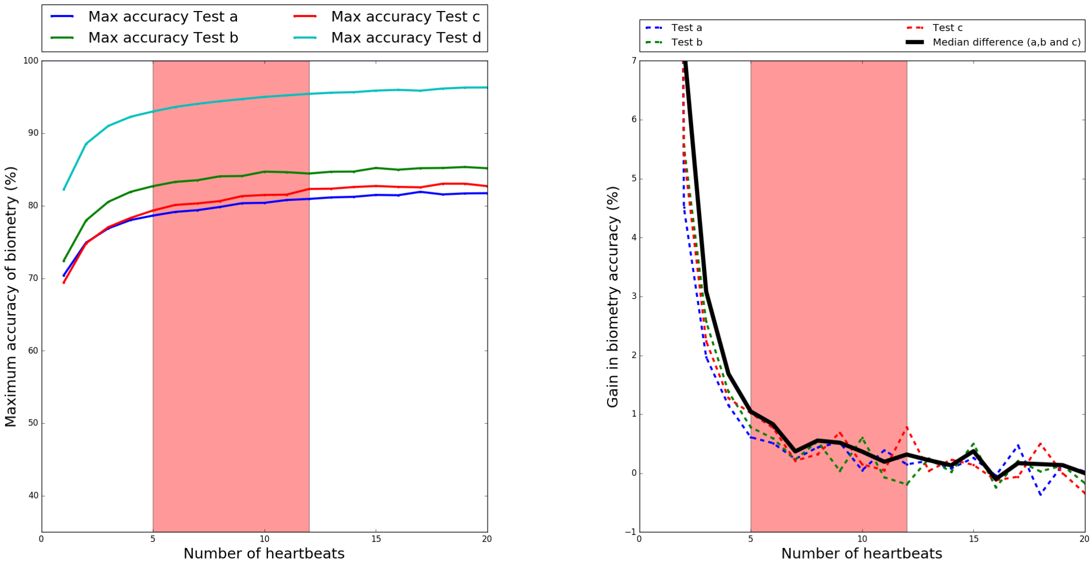


Figure 4.6: (**left**) Best accuracy obtained using all individual contexts tested ($k \in \{1, \dots, 40\}$) for each number of heartbeats as target (from 1 to 20). (**right**) Accuracy increase when adding more heartbeats for the testing segments.

4.3.5 Conclusions

Our results showed that it is possible to identify participants accurately around 75-80% of the time with only 5 to 12 heartbeats, at least for the database used. Even though the result is lower than what would be desirable for a real system, each of the experiments was performed using only one context depth ($k \in \{1, \dots, 40\}$).

4.4 Extended-Alphabet FCMs for Biometric Identification

4.4.1 Overview

In Section 2.4.3, we have introduced the theory behind the xaFCMs, a generalization of the FCMs, that calculates the probability of occurrence of the next d symbols, given the previous k symbols.

Similarly to the previous sections, we will addressed the topic of ECG based biometric identification using the Normalized Relative Compression (NRC). To attain the goal, instead of building finite-context models (FCM) to represent each individual [17, 37], we will use xaFCMs and discuss its possible advantages.

4.4.2 Method

In this section, we have followed the approach described in Section 4.3 for both the preprocessing and quantization. We will compare the results obtained by replacing the compression models from FCMs for xaFCMs, in order to test if they are suitable for this problem, given both their reduced testing times and model complexities.

The theory behind xaFCMs can be found in Section 2.4.3. The main idea behind these models is to predict d symbols at a time by looking at the previous k symbols, instead of making the predictions one symbol at a time, like a regular FCM. Using xaFCMs instead of FCMs has some advantages for biometric identification, which we will explore.

Compression-based approaches found in the literature for ECG biometric identification do not seem to take advantage of the fact that the ECG is a quasi-periodical time-series. Since our method uses a semi-fiducial approach (it only detects the R-peak), it is trivial to know where the repetition should happen and take advantage of that fact. From previous results [48], we concluded that, when consecutive *heartbeats* present low levels of noise, their quantization is almost identical. As a consequence of this, we consider that any sequence we analyze is a circular sequence [111]. From this result, it is possible to infer that, compressing the beginning of an heartbeat using the end of the same heartbeat, may be identical to compress it using the end of the previous heartbeat. This may not sound as an advantage, however, this fact allows us to use heartbeats that are not consecutive, when performing the identification of a participant. That is being taken it into account when building the models (one of the arguments that the algorithm accepts as input is the length of the expected repetition – i.e. for this application, how many symbols one heartbeat contains), because it will be important for building a real system, as we expect more noise to be present and, therefore, some segments need to be discarded when performing the compression [48].

4.4.3 Experimental Results

After all the already explained preprocessing steps are complete, the process in which we perform the biometric identification is the following:

1. Use the complete ECG signals from two days, in order to build a xaFCM model that describes each of the participants;
2. For the remaining day, split the signal, such that each segment has 10 consecutive heartbeats inside it;
3. "Compress" (compute the NRC) each of the segments obtained in the previous step using each of the models obtained in the first step;

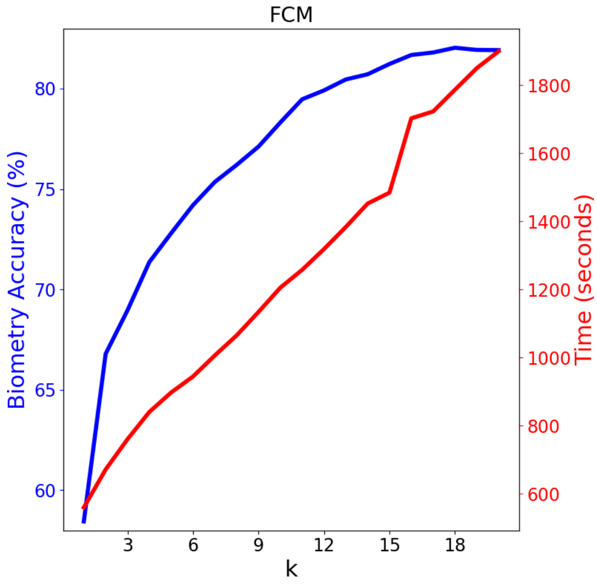


Figure 4.7: FCM biometry process: context k changing; the blue line represents the biometry accuracy (in %) and the red line represents the time of execution (seconds).

4. The model which produces a lowest result is chosen as the candidate for biometric identification.

The justification for the first step is that we do not want to use any information from the ECG of the day where we are trying to perform the ECG biometric identification, since, if we used that information, our results would not match a real situation.

The number of heartbeats needed for ECG biometric identification is undoubtedly useful when building a biometric identification system – any system should ask participants to provide data for identification, using the smallest time interval that is possible, for practical reasons. Based on the results from a previous study [37], we concluded that 10 heartbeats is a good trade-off between collection time (which should be as low as possible) and statistical relevance of the data.

All experiments were implemented and performed using Python 3.5 (Linux 64 bits) on an Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz, with 32GB of RAM. For simplicity of code, we have not parallelized the process – therefore, only one logical core was used for each experiment.

In Fig. 4.7, it is possible to see a plot with the accuracy obtained for the process described, by using FCM models, with all possible values of k from 1 up to 20. In the red line, it is also possible to see how much time does this process take in total. An important fact is that the time taken to perform the biometry is approximately directly proportional to the size of the context, k, used.

Since the purpose of these experiments is to show the appropriateness of xaFCM models, in Fig. 4.8 are shown six examples of the same experiment, but instead of changing the context k, we have chosen a fixed value of k and tested all possible values of d, the depth of the xaFCMs. From these plots, it is possible to see that the time taken to perform the biometry process for the whole database is up to 3-4 times shorter when using high values of d, having, usually, accuracy ratios comparable with the FCMs of the same order k.

On the experiments using "lower" values for the context k (in this case, $k \leq 14$), it is possible to notice a minor improvement in terms of accuracy as the d increases, at least for the first values of d ($d \leq 7$, more or less). This makes us think that increasing the depth d behaves in a similar way to increasing the depth k of the xaFCM, without the additional cost in terms of testing speed (quite the opposite, actually) and the memory needed does not increase so much as it would by increasing k (Fig. 4.9).

In higher contexts k we get the same advantages in terms of computing time and memory requirements. However, after a certain point, there is just no real benefit from increasing neither the context k, nor the depth d, since we are looking for "too specific" patterns, that may not appear again on the segments being tested – which, making an analogy to machine learning, the model would be overfitting to the training data.

Another aspect we wanted to show, regarding the advantages of using xaFCMs, is the model complexity. In order for the biometric identification to be executed fast, in practice, it is needed to have all the participant models previously loaded into memory. This usually does not pose a problem, if there are not many participants, but it may be useful for building a real biometric identification system. In Fig. 4.9, we can see that by increasing the context k of FCM models, the complexity of each model increases exponentially. From our interpretation, a way to avoid this exponential increase is to use a xaFCM with an order slightly lower and increase its depth d. In order to show this, we display the complexity of such models in Fig. 4.10.

4.4.4 Conclusions

In this section, we have shown that extended-alphabet finite-context models, a generalized version of FCMs, are suitable for ECG biometric identification, replacing traditional FCMs with similar results. However, in some conditions xaFCMs might be preferable. For *non-sparse* models, FCMs have shown to increase exponentially in memory usage, while xaFCMs allow to obtain some extra *depth* (d) at a linear memory cost. They have also proven to provide faster speeds (up to d times) for testing, *i.e.*, computing the relative bits necessary to compress a certain string. For applications that run in real time, this might be an important factor to take into account.

These models were developed for the application of biometric identification using the ECG, on a previously quantized dataset, but they should work for any kind of string. In Appendix 5.3 we show some results using relative compressions of DNA data with different chromosomes of the human species and chimpanzee data, in order to show that these models do make sense for different purposes and applications.

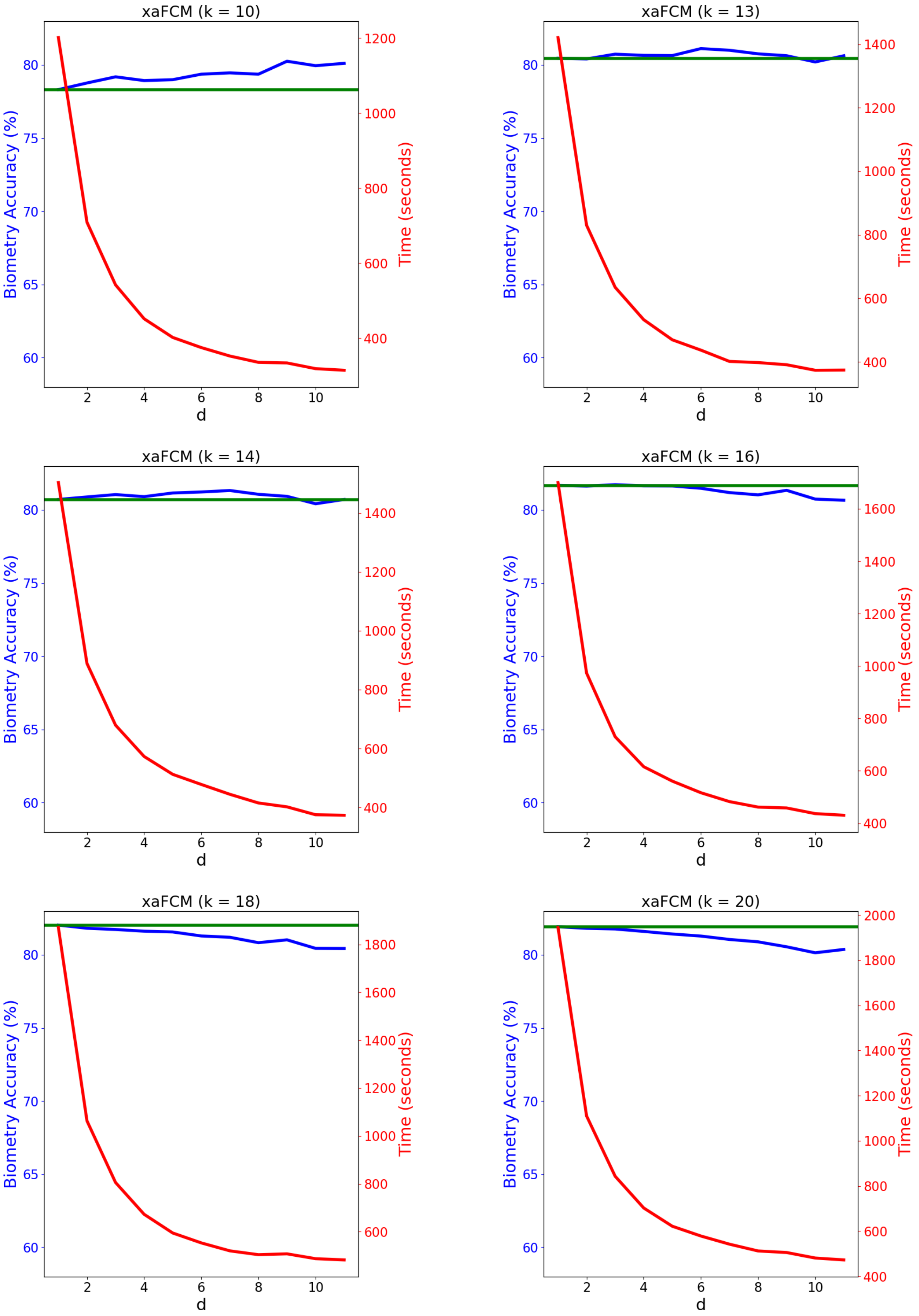


Figure 4.8: xaFCM biometry process using high contexts *k* (fixed); depth *d* is changing from 1 to 11; the blue line represents the biometry accuracy (in %); the red line represents the time (seconds); the green line represents the accuracy that a FCM with the same context would obtain.

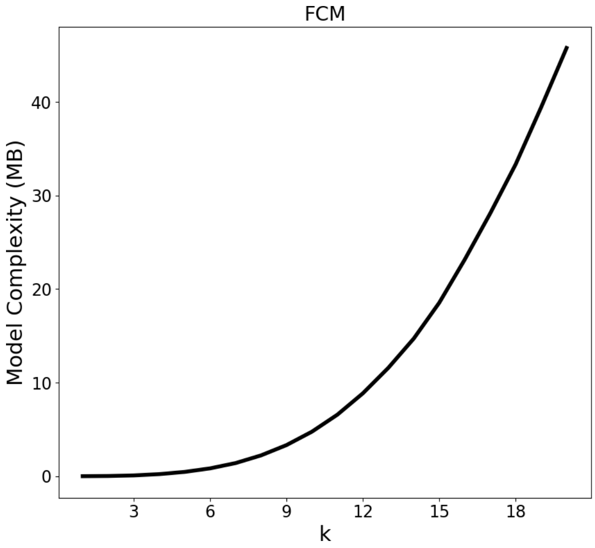


Figure 4.9: FCM average model complexity per participant - context *k* changing.

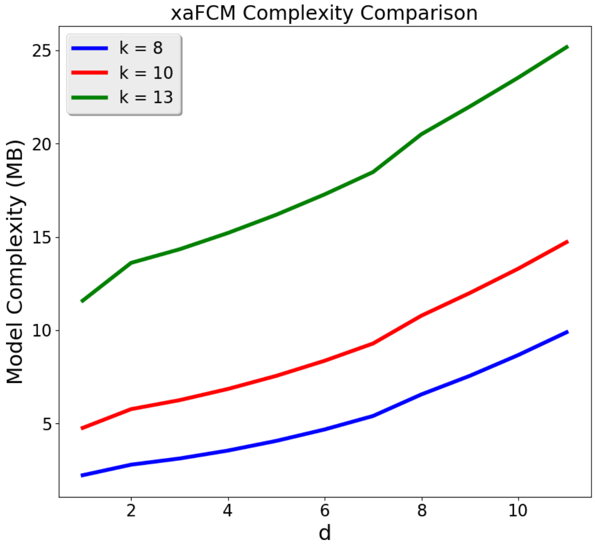


Figure 4.10: xaFCM average model complexity per participant - context *d* is changing; *k* is fixed.

4.5 Non-fiducial ECG Biometric Identification

4.5.1 Introduction

We have already used methods based on compression in order to perform biometric identification using the ECG [17, 37, 36]. However, the approaches used relied on the detection of a fiducial point (a “point of interest”) in each heartbeat found in the ECG signal, called the R-peak. The detection of such points on clean signal is a computationally simple problem, with algorithms attaining accuracies of around 99.9% on clinical signals [86]. But, as it is well known in biometrics, most times we need to deal with highly noisy signals, making that detection prone to error and, by transitivity, partially corrupting the whole process of identification.

In what follows, we present a non-fiducial method for ECG biometric identification, that uses a Lloyd-Max quantizer [89, 90] on first-order differentiation of the signal (the differences between consecutive points in the signal). We show that, using this approach, we improve previous results obtained on the *Biometric Emotion Recognition DB* (see 1.3.2), like the name suggests, originally collected for emotion classification [18].

4.5.2 Method

As mentioned in Chapter 3, when the distribution of the variable to quantize is given by some complex mathematical function, like the ECG signal, for which we cannot find a simple mathematical function that describes the signal, a non-uniform quantizer should be used. In this case, we have opted to perform the quantization using a Lloyd-Max quantizer, applied to the consecutive differences between consecutive samples on our signal. We have used an alphabet size of 17, which the quantizer represents by the symbols corresponding to the first 17 letters of the alphabet: ‘A’, ‘B’, ... ‘P’, ‘Q’.

An overview of the method used in this work can be seen in Fig. 4.11. We start by cleaning the ECG signals by using a Butterworth low-pass filter of order 5 with a cutoff frequency of 30Hz. The obtained signal is then transformed into a series of differences (which corresponds to the first-order derivative of the signal).

Since we want to apply a Lloyd-Max quantizer to this series, we perform a 2-pass process on the training data: first, for each participant in the database, we learn the breakpoints that optimize its Lloyd-Max quantization; on the second phase, we apply the corresponding breakpoints to each participants’ training data, in order to perform the quantization.

From the quantized training data, it is possible to learn a model that describes each participant’s data by using a context-based compressor, such as a xaFCM. It is important to notice that each

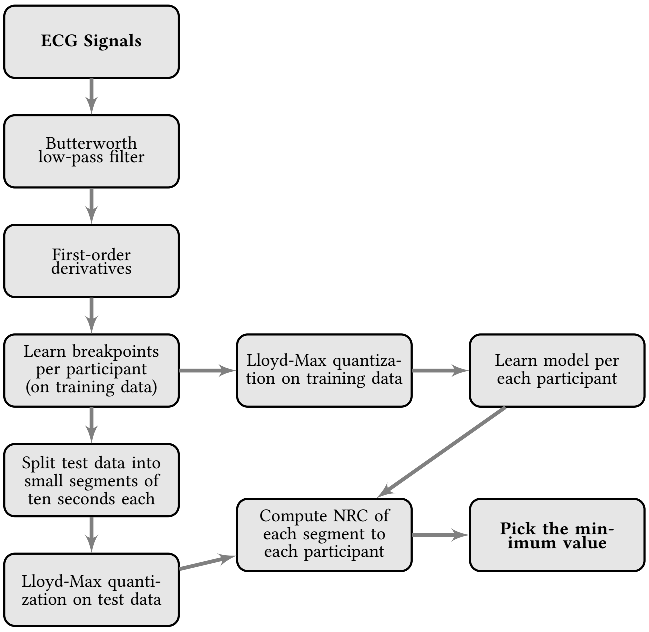

ECG Signals
Butterworth low-pass filter
First-order derivatives
Learn breakpoints per participant (on training data)
Lloyd-Max quantization on training data
Learn model per each participant
Split test data into small segments of ten seconds each
Compute NRC of each segment to each participant
Pick the minimum value
Lloyd-Max quantization on test data

Figure 4.11: High-level workflow for the ECG biometric identification process used.

model, besides learning the xaFCMs, also takes notice on which participant it is representing—this is important, because those breakpoints will also be used during the testing phase.

The splitting of the test data into segments of 10 seconds is then performed. At this point, it is only required to compute the amount of bits it takes to compress each of those segments, by each of the participants' models. This step is done in two phases: the first is to perform the quantization of the segment being tested using the breakpoints corresponding to the model that we are using; afterwards, the estimation of the amount of bits needed to represent that sequence using the xaFCM is computed. The model that produces less bits, *i.e.*, the one which has a lower NRC, is our guess as the correct participant.

4.5.3 Experimental Results

The database used in this study was the biometric emotion recognition database (1.3.2), used in most of the works presented so far, providing a good benchmark to compare this approach with the previous ones.

We tried to replicate as much as possible the experimental setup used in previous works [36, 37], in order to have fair benchmarks against those systems. However, since the previous methods used R-peak detection, the way to measure the size of the ECG samples used for testing was done in complete heartbeat cycles, instead of seconds. In those previous methods, we have used 10 heartbeats for each test. In order for the results to be comparable, in this work we assumed that one heartbeat is approximately 1 second. Therefore, each test is performed using 10 seconds of ECG data. Even if this approximation is not completely accurate, *i.e.*, if we use a little more data (or less, depending on the heartbeat rate of each participant) than the previous experiments, it should not impact the results significantly, as we showed in Section 4.3 [37].

All the experiments were performed on a Amazon AWS EC2 instance (c5.9xlarge), with a 3.0 GHz Intel Xeon Platinum (34 cores) CPU and 72GB of RAM. The operating system used was Ubuntu Server 16.04, and Python 3.6.4. The process could run on a regular laptop computer with 8GB of RAM, but we decided to use a cloud instance computer in order to make use of the parallelized code and have faster results. As mentioned in the previous section, all the base source code is freely available and can be downloaded from the Github repository.

Using higher values of the depth, d, of the xaFCM, has the advantage of providing very fast results, at sometimes the possible cost of some decrease in accuracy (the theoretical explanation for these concepts can be found in 2.4.3 [36]). For that reason, we use high values of d for finding the areas of interest for the parameter k and, afterwards, we start decreasing the value d and reduce the number of simulations that we need to run in order to find the optimal values of the context, k, in order to obtain more accurate results. Of course, the optimal values of k also depend on the depth, d, but using high values of d gives an idea of the region where experiments should be performed with higher resolution.

Figure 4.12 shows all the experiments ran for biometric identification on this database, for different values for the depth d and context k. As mentioned in Section 4.5.2, all the experiments used two days for training the models and all the available ten-second samples of ECG from the remaining day as the tests.

The first phase of tests was ran with $d = 10$, experimenting contexts k from 1 up to 100. It is easy to see from the plot marked in green (Fig. 4.12), that the possible area of interest for k lays somewhere between 15 and 50 – the best value was found for $k = 30$, with an accuracy of 87.5%. The second phase (marked as blue) uses $d = 5$ and narrows down the area of interest for k from around 25 to 50, with the best results for $k = 35$, with an accuracy of 88.6%. Then, since we have a small region of interest, we performed some tests using $d = 2$ and the best value found was with $k = 38$, with an accuracy of 89.3%. Actually, if we look at the differences in accuracy, depending on the requirements in terms of speed, it might not even be worth using small values of d for this application. The results in terms of the choice of d are consistent with our previous results (in ??) [36]. Regarding the context, k, from these results, we can infer that this new approach requires higher values of k in order for the model to have a good internal representation of each participant. This might have an impact in terms of memory usage and time of execution. However, since the size of the data used is usually not significant (a couple of megabytes), the models also do not grow exponentially to values that cannot be represented by a regular laptop, as they would with data like DNA sequences [36].

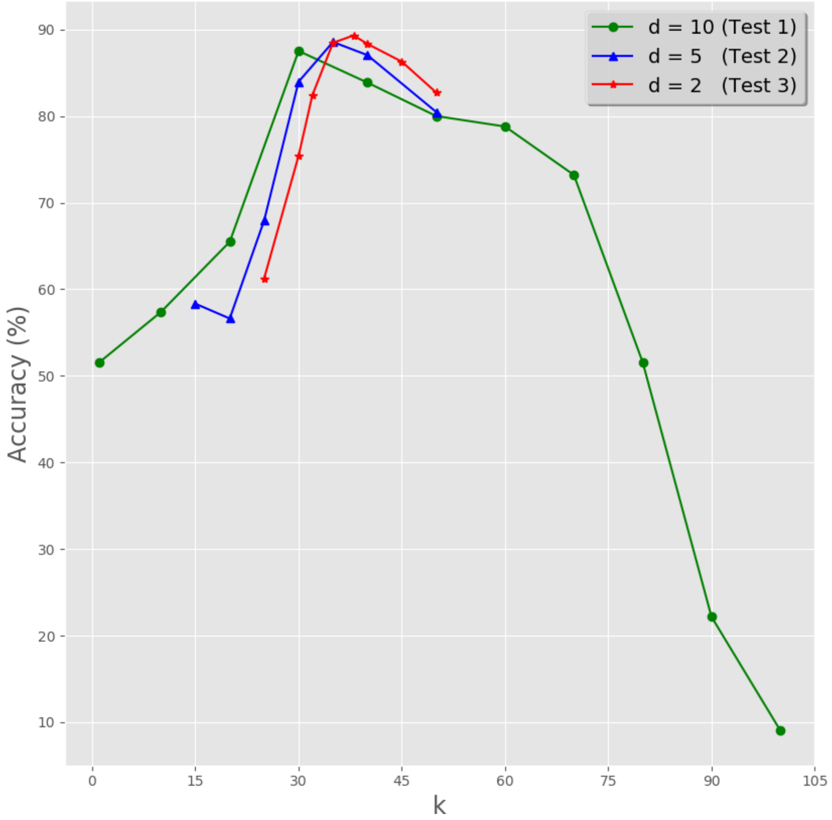


Figure 4.12: Biometric identification accuracy, using $d = 10$, $d = 5$ and $d = 2$, as a function of the parameter k.

Besides the accuracy, it is also useful to check a measure that takes into account the precision obtained. For that reason, and also to have a more clear understanding of the types of error that our system is having for this dataset (to answer questions like "how many false positives/true negatives are we obtaining in each class?"), we show the confusion matrix of the predictions made by the system, against the true labels. In Fig. 4.13, we show the confusion matrix obtained for the experiment using $k = 35$ and $d = 2$, for which we obtained an accuracy of 88.5% and F1-score of 0.88.

It is interesting to notice that when testing ECG signal belonging to participants 0, 3, 6, 8, 13, 14, 16 and 21, the system almost does not make any mistake. For the other participants, the system makes mistakes, but they are "spread" amongst different other participants, *i.e.*, the system never consistently mistakes one participant by a specific different one. This is a very important feature on a biometric system, because it makes it harder for someone to fake a specific identity. Our initial hypothesis was that this usually happens when the amount of training data provided for one participant is not proportional to the other ones–however, as we will explain in the next section, this initial hypothesis might not always be true.

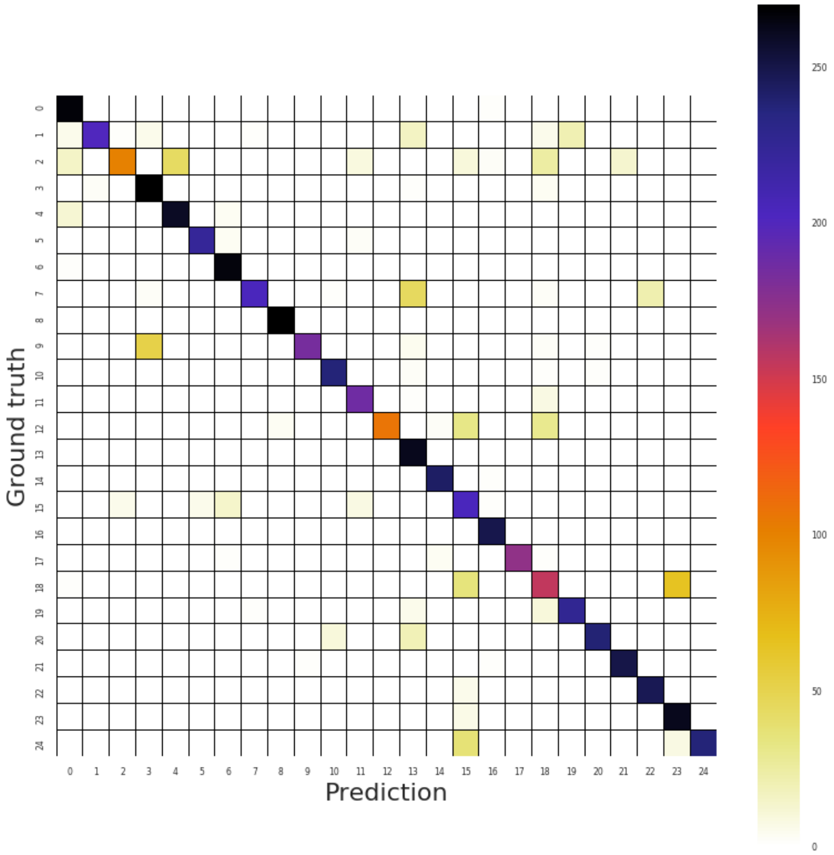


Figure 4.13: Confusion matrix for biometric identification using a xaFCM of context $k = 35$ and depth $d = 2$. This test used two days for training and the other day for testing. Each test was performed using 10 seconds of ECG. This experiment achieved an accuracy of 88.5% and F1-score of 0.88.

4.5.4 Conclusions

We have introduced a compression-based non-fiducial method that works with first order derivatives, instead of the original signal, for performing ECG biometric identification.

This method beats previous state-of-the-art methods using the same database, achieving an accuracy of 89.3%. Moreover, it uses the same amount of training data as the previous methods, that have attained, at most, around 80% of accuracy (4.3) [37].

We are confident that these results can be further improved. However, since the purpose was to introduce the main ideas associated to the method, we did not perform an exhaustive search for optimal parameters, neither experimented with mixtures of finite-context models (collaborative models), which, on machine learning terms, behave like a dynamic voting system.

While these results seem very promising, future work needs to be done in order to check how well this approach works when dealing with intruders in the system, for example. For attaining that goal, the first step is to switch from a classification problem to a real biometric system, where there should be a threshold value for the NRC, instead of always accepting the minimum value as the correct participant.

4.6 A Preliminary Exploratory Study on ECG Biometric Bias Using Compression Algorithms

4.6.1 Introduction

On most machine learning applications an open issue is how to properly deal with unbalanced datasets and avoid bias towards some classes in favor of others. This is also a problem on biometric identification systems, where such a behavior from the system would allow some individuals to deceive the system, being identified as someone else. In order to solve this problem, it is important to understand why some individuals are more prone to force this behavior to the system.

We have shown that compression-based methods can be used for the task of ECG biometric identification. However, biometric identification systems, like any other artificial intelligent systems, make mistakes. Those mistakes, when examined closely, do not seem to be random by nature, as they tend to be made when trying to identify a specific subset of participants. In this work, we start an exploratory study on what errors are made by such a system so that, in the future, they may be prevented.

In this section, we will use a mechanism to force data imbalance between the classes, in order to evaluate how it affects the performance of the biometric identification system. We will use the previous results as a baseline to explore what mistakes are made by the system, as it seems that the biometric signature of individuals differs in nature. We will then discuss how this might affect real systems and what can be done to make such systems more robust.

4.6.2 Experimental Results

To perform this study, we used the openly available database *Biometric Emotion Recognition DB - UA* (described in detail in 1.3.2). On previous studies, where we used the same database, the test performed was to use two out of the three sessions of data per participant as the training data (reference data) and the other session as the test data (target data). Each test segment contained exactly 10 seconds (see 4.3 [37] for more information). On the previous study, we could achieve an accuracy of approximately 89.3% (4.5.3) [35]. The confusion matrix for that problem can be seen in Fig. 4.11. This was the ground floor for the tests that we have done.

However, since the purpose of this study was to explore the impact of the data imbalance available for training each class (subject) in the biometry results obtained and how it might provide bias towards some participants, we opted for a simpler setup: instead of using all the subjects on each

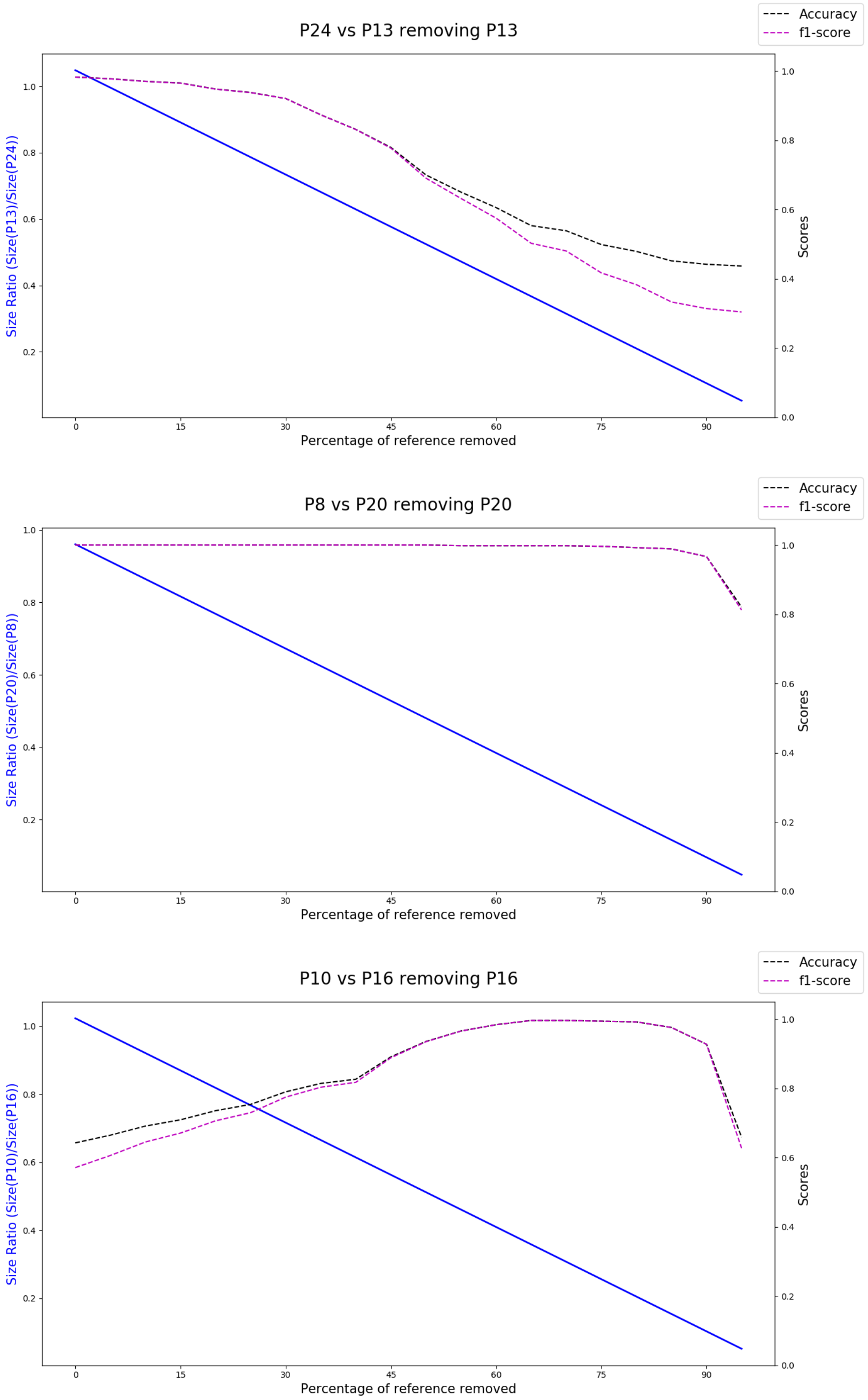


Figure 4.14: (Dashes-lines) Measuring the accuracy and f1-score as the percentage of the reference of participant one of the participants is trimmed from the end. On the top plot, both performance measures decrease slowly as expected; on the middle plot it is possible to see an example where even with only 10% of the reference data the subjects can be easily distinguished; on the bottom plot is an example where there is another kind of weird behavior by the system – instead of decreasing the performance measures as on of the reference gets trimmed, the opposite occurs (until a certain point). **(Blue line)** Measuring the ratio of reference size available for one participant and the other – the further away from 1, the higher the data imbalance.

test, we used two subjects per test and performed a one-to-one classification – *i.e.* we performed the biometric identification between pairs of individuals at each time. This way it was easier to notice how decreasing the amount of data available for the reference of participants, forcing the problem to be unbalanced, impacts the biometric identification results. Given the time required for each experiment and also the exploratory nature of the experiment, we have decided to do only 50 of these tests.

It is important to notice that we never used data from the same recording/session for training and testing. Since the ECG signal changes substantially from one session to another, using data from the same session for the training and test would lead to over-optimistic results [37]. The reason for this is that there is an intra-variability for each participant from one day to another, which makes the biometric identification more challenging [110, 95].

The basis for all tests is to pick two participants randomly, use all the available reference data for both participants and keep removing parts of the reference for the other participant to see how the performance measures for the biometric identification are affected as the data imbalance increases between the classes. Instead of measuring the amount of data we have for each class, we only care about the ratio between the data available for one class and the other – the further away from one, the higher the data imbalance. It is also worth mentioning that we measure the amount of data by the number of samples available and not the file size itself. We performed three different tests:

1. The beginning of the reference (training) data was removed, 5% at each time;
2. The end of the reference data was removed, 5% at each time;
3. The training data was divided into 20 equally sized intervals and the first 5% of the data of each interval were removed.

The idea behind having three different tests was to evaluate if the results obtained were similar when removing data from the reference on different regions. Surprisingly, the results were similar on the three tests for almost all pair of participants evaluated. For that reason, the experiments suggest that the biometric results depend on the amount of data available for training each class and it does not seem to be related to a specific part of the signal.

Between all test combinations, three different categories of results were obtained, *i.e.*, three different types of behaviour were found on the experiments. In Fig. 4.14 we show one result for each of those categories, using test number one (removing the beginning of the reference data) and reporting both the accuracy and f1-scores obtained as the reference data of one of the participants is removed. From those results we can see clearly the three different categories/behaviors:

(a) A decrease on the performance measures as the information available for the reference (training) decreases – what was theoretically expected. As the data available for the reference decreases, the problem becomes more unbalanced and tends towards the class for which more data is available. Example: top test from Fig. 4.14;

(b) An almost perfect biometric identification between the two subjects, even when the problem starts becoming highly unbalanced – it is possible to notice that even without 80%-90% of the original reference data for one of the participants, the biometric identification is still successful most of the time. Example: middle test from Fig. 4.14;

(c) Instead of the results getting worse as the problem becomes more unbalanced, they improve. This does not happen on most of the tests, but it is worth inspecting these specific cases to get some insights on what is the mechanism that causes this unexpected behavior – that might be useful to understand the bias on biometric identification. It is important to mention that this behavior is the opposite of what was expected. Example: bottom test from Fig. 4.14.

4.6.3 Discussion and Conclusions

An important insight to take from this exploratory study is that not all people need the same amount of data to be correctly identified by an ECG based biometric identification system. These results suggest that the biometric signature of some individuals might be "simpler" than others, which reinforce the need for improved encryption methods and security protocols to deal with this kind of application. An example of such an individual is P20 (from the middle test of Fig. 4.14), where even without 90% of its original data was still successfully identified against P8. In this kind of situation, where decreasing the amount of reference data does not decrease the performance measures, we may question if there is the need to collect and use so much data. Probably there is no harm in collecting it (except for privacy concerns), but future work needs to be done in order to understand if the data is really necessary for biometric identification, *i.e.*, if it helps discriminate against any other participant.

We showed that not all participants need the same amount of data for ECG biometric identification. However, that does not mean that the information content of their signatures is not the same. Our interpretation, that we plan to test as an extension of the current work, is that this question might be related with the Kolmogorov complexity of the ECG signals used for references: if the complexity of the signals of all participants is similar, they should have approximately the same information. Otherwise, it is natural that the participants information is unbalanced and we obtain unexpected behaviors. That might explain why in some cases turning the problem unbalanced actually helps the results – we are turning it unbalanced in terms of data, not necessarily in terms of information. We want to further explore this problem from that perspective, as it might be useful to solve or at least to quantify not only this problem, but also other unbalanced classification problems. Given that the Kolmogorov complexity of a signal is non computable, we need to obtain an approximation using appropriate compression algorithms. As a first approach, we plan to use FCM based compressors to find an upper bound to that complexity and further test this theory before doing any kind of classification/biometric identification.

4.7 Conclusions

In this chapter, we presented different schemes to explore and solve the problem of ECG biometric identification, using compression-based approaches. In order to do so, we have used methods from all the previous chapters: from pre-processing, to quantization into symbolic data and using finite-context models to compute the normalized relative compression.

The presented results did not outperform current state of the art results, but they provide some advantages over most of the commonly used alternatives nowadays, such as deep learning, which rely on higher computation costs. The approaches described in this work can successfully be reproduced on low cost devices, such as a Raspberry Pi [112], which are more commonly found in a real world scenario in the industry. Another advantage is that these types of models allow for an incremental learning approach [113], allowing the models to be updated when new signals are acquired, both for subjects that are already present in the database or for new ones, without the need to re-build the models from scratch each time.

Furthermore, since we have followed a non-fiducial approach, these methods are agnostic to what type of signal is being used, *i.e.*, there is no theoretical reason for these methods not to work on different classification problems, both with ECG signals, or any other 1-dimensional signal, as long as the pre-processing filters are adjusted and the parameters of the different methods are adjusted accordingly.

Chapter 5

Conclusions

5.1 Discussion

During this thesis we have covered parts of different areas of expertise. We started by introducing the Kolmogorov notion of complexity, as well as some distances based on it – which are not computable, since the metric itself is not computable. We justified why using compression based metrics is a reasonable alternative and introduced some of them, namely, the NRC, a measure based on the notion of relative compression. In order to use that metric in practice, we introduced FCMs and how they can be applied to compute the NRC. But since the datasets are usually "large", we ended up proposing a generalization to FCMs, called xaFCMs (eXtended Alphabet Finite-Context Models), which can compress d symbols at a time, achieving similar results at a lower execution time.

Even though the ECG biometric identification was only an example for the methods proposed, and not the goal of the work itself, we have introduced what is an ECG signal and some basic concepts regarding signal processing applied to ECG data, as without applying some pre-processing techniques and understanding the essentials about those signals it would be difficult to perform any application. We also mentioned some common quantization techniques and proposed an approach based on the Lloyd-Max quantization which operates on the differences between consecutive samples on a signal – an approach that is more suitable to ECG data than operating on the signal directly, because of baseline wander, which is common while collecting long ECGs.

Using the previous methods, we built some different workflows for performing biometric identification, which were presented on the previous chapter, as well as some studies regarding the impact of variables such as noise removal or the acquisition time necessary to successfully identify an individual. We finished the chapter with a preliminary study regarding the errors made by our proposed algorithm – which we believe are key to advance towards a more robust approach.

In this section, we discuss some topics that we believe might be of interest and worth exploring further, but might not have been examined with enough depth. Also, some topics might have been mentioned but we would like to highlight the reason behind some choices that were made. We will also mention some limitations of this study and what we would have done differently if this project would start now as well as the reasons for that.

Biometric Identification and Classification

Along the document, we always refer to ECG biometric identification. However, in the literature, when that term is used, it might refer to any of the following three situations:

1. A binary problem, where the person being identified is declared as an impostor or as himself;
2. A n-classes (n is the number of subjects in the database) classification problem where the person being identified is assumed to be already in the database (training data) and the classifier should output which;
3. A n-classes problem, where there is no assumption whether the individual being identified is on the database or not. The algorithm should output if it corresponds to any individual(s) from the database.

In this work, whenever the term "biometric identification" is used, it always refers to the second situation in the previous list. The reason for doing so was that we wanted to make sure that this type of compression-based approach was able to obtain reasonable results before attempting to use it as on the third situation, which is clearly more difficult, as it requires the definition of thresholds for the algorithm to decide if someone is present on the database or not.

Compression versus number of bits

Regarding the "compression phase" of the step for classification presented, although it was mentioned that we only compute the number of bits that the compressor would produce, instead of performing the compression itself, we did not explain how that would have been done. There are several ways this could be achieved, with one being using an Huffman-encoder together with the probability table for the next symbol (or d symbols, in the case of an xaFCM), for each context as the compressor runs trough the sequence being compressed. Because this is a relative compression, the model was previously learned and "frozen" on a different sequence, so the compression process is straight-forward, as described.

It is also important mentioning that the size of the models learned from the references were not taken into account while computing the NRC values during this work. This might seem like a huge bias from a compression perspective, however, for the classification/biometric identification task, it would make no difference on the end result. The reason for this is that, on this approach, the minimum value for the NRC is always picked as the correct guess – this means that for two different values for the NRC n_1 and n_2, if $n_1 \leq n_2 \implies n_1 + \epsilon \leq n_2 + \epsilon$, where ϵ represents the overhead of the reference model.

Information loss during quantization

Although the different compression schemes used in this work are lossless, we are aware of the loss of information during the quantization step. There is no complete guarantee that on this step there is no loss of information that could end up being important for correctly identifying someone.

In [35] we introduced a way to mitigate that information loss or, more specifically, to incorporate some of the lost information on the models learned. Since FCM based compressors are symbolic, we performed a modification where each model was learned both from the original data and from its quantized output – from the original data, the model learned what were the breakpoints used for optimize a Lloyd-Max quantizer for the training (reference) data; the finite-context models were learned from the output of the quantization of the same data.

Data Leakage

We had some trouble comparing our methods to some of the results by other methods proposed in the literature, which sometimes did not report some details regarding how the performance was evaluated, making it difficult to perform a fair comparison. We plan on writing a more detailed paper regarding this subject, but would like to report the two main issues we encountered and a way to avoid them.

First, it is worth mentioning the importance of making sure the train and test dataset do not belong to the same session, when dealing with physiological signals and, specifically, the ECG. We have done some tests on biometric identification which show that including a small sample of the same session as the test data on the training data, provides great results ($\approx$ 4% error instead of $\approx$ 19%) [37]. Of course such a scenario is unrealistic and therefore should not be taken into account while evaluating any biometric identification system or any other system where the real problem will deal with data from different sessions. Nonetheless, in the literature it is possible to find different studies where the biometric identification is evaluated using databases where only one sample per subject was collected – a scenario in which, for the reason mentioned, it is not possible to simulate a "realistic" biometric identification scenario and cannot be directly compared with the results presented on this thesis.

Another way in which *data leakage* from the test set into the train set, or even between different samples on the test set (quantizing a 30 minute ECG as a whole should output different results from quantizing the consecutive 30 segments of one minute of ECG individually, depending on the quantizer used) occurs is during the quantization/data transformation step. If the quantization of the signals is done on each complete signal and part of that signal is split and used for the test dataset, there is already information being leaked from the train to the test dataset (or, in compression terms, from the reference to the target data). In order to avoid this issue, we suggest to quantize each test signal individually, simulating a more realistic scenario. In our case, since the ECG signal suffers from baseline wander, we have chosen to operate on its consecutive differences, instead of on the original signal, and use a Lloyd-Max quantization scheme on top of the signal – both on the training and testing data separately), avoiding data leakage from training to test [114].

Collaborative models

During this work we used different types of compressors based on FCMs in order to compute the NRC between a test sample and its possible class (or participant, in the case of biometric identification). We have also mentioned in Chapter 2 that the theory behind the idea of using compressors to obtain a (relative) similarity metric is the Kolmogorov complexity, as they can provide an upper bound for such value, which is not computable in nature. Based on these, the expectation was that the higher the compression ratio achieved, the more useful and accurate would be the classification via the NRC. Surprisingly, that does not seem to be the case. Sometimes, simpler models (for example, a FCM using a single high order context instead of a mixture of finite-context models), provide better classification results.

These results which, at first glance, would contradict the theory behind using these methods to approach the relative Kolmogorov complexity, can actually be possible and not contradict each other. Our theory regarding how this might occur is that the single context (high order) models capture some of the variability that is in fact key for distinguishing between different classes. On the other hand, the mixtures of models require a lower number of bits to compress a certain test string, but the gain in compression is not being gained by something that discriminates the class it belongs to. This phenomenon does not only occur when dealing with single context *versus* mixtures, but also while using mixtures of exclusively *low/middle-order contexts* versus mixtures of exclusively *high-order contexts* – the latter seem to provide better classification results, even when they do compress worse than the former.

5.2 Future Work

Auto-Encoders

Regarding the proposed workflow for biometric identification, some other alternatives could be explored in order to lose less information during the quantization step or explore that loss further, but had to be left out as future work–namely, using auto-encoding neural networks. The idea would be to use the reference data to "learn" an auto-encoder [115, 116] where the input and the output data layers were the original ECG data, using a bottleneck with the same dimension as the alphabet size of our current scheme. Afterwards, using the weights from the learned network in order to predict the error on the test data. Two options could be done for using these results: either ignore the "pure" compression based approach, and use this measure for performing classification or, use this metric on an hybrid (weighted) approach with the methods we have used.

Myocardial Infarction

The automatic classification of myocardial infarction (commonly known as heart attack) is, undoubtedly, a useful tool for hospitals, as a decision support tool. In [42] we did a preliminary study, using the same approach used for biometric identification, in order to validate that the method would be useful for clinical applications. In this case, the task was infarction detection and the data used were the PTB Diagnostic ECG Database (Section 1.3.3), obtaining an accuracy of 85.6% for this task. It would be interesting to test this method for other databases and clinical applications and compare the results to those obtained in the literature.

5.3 Final Remarks

This document was meant at proposing a possible solution to classify 1-dimensional signals using a compression-based approach–goal which in our opinion was achieved. Nonetheless, there are limitations to the approach presented, which were covered in this discussion. We would like, in the other hand, to highlight some useful insights that compression-based approaches such as these can have, namely, when dealing with classification problems using one-dimensional signals. There is a tendency for machine learning methods to focus on benchmarks and, therefore, to ignore the computational costs and times involved on the proposed solutions. Unfortunately, on real world applications, those methods with high computational might not be applicable under most scenarios– a problem which these methods can solve, as they can easily be implemented on even low-cost hardware. Another important insight these methods can offer is the possibility of understanding how related different classes (assuming we are dealing with a classification problem) are, using easy to interpret measures, such as the normalized relative compression, amongst others, introduced in the second chapter of this document (Chapter 2). Even if compression-methods for classification tasks do not provide the state-of-the-art results, they can be a useful way to understand relations and patterns on data in a straightforward way.

Appendix A – Extended-Alphabet Finite Context models – Experimental Results (DNA Sequence Relative Similarity)

An approach for computing the similarity of a sequence relatively to other is to calculate the NRC using one of them as reference and the other as the target. In previous works, this has been done using FCM compressors [117, 118, 10, 119].

In order to show that the xaFCMs are also suitable to use in the manner, we ran some simulations using the human and chimpanzee DNA sequences, removing the unknown symbols (N). The idea was to use each chromosome of the human species as reference and then compress each chromosome of chimpanzee as the target, using exclusively the model from the reference. Since we know from evolution theory that these two species are closely related [120], it is expected that, when we are compressing homologous pairs of chromosomes, the NRC should be lower than on the other cases.

Table 1: CPU time and memory usage (RAM) of the experiments with DNA sequences.

Parameters (context k and depth d)	Average Time to Learn the Model	Average Time to Compress	Average Memory per model	Total Time to run the experiment
$k = 12, d = 1$	1649.6 sec	1580.5 sec	5043.2 MB	274.4 hours
$k = 12, d = 8$	2181.2 sec	269.5 sec	14350.3 MB	59.5 hours

To perform the experiment, we used the assembled human chromosomes 1 to 22, X and Y (3.1GB of data in total) and assembled chimpanzee chromosomes 1, 2a, 2b, 3 to 22, X and Y (3.2GB of data in total)[1]. We ran two different simulations: the first one, with a FCM of context $k = 12$; the other with a xaFCM with $k = 12$ and $d = 8$. All the experiments ran on a server with 16-cores 2.13GHz Intel Xeon CPU E7320 and 256GB of RAM, but the implementation used a single core.

Table 1 shows the average times taken by each experiment, as well as the average memory needed to store the xaFCM model to represent the human chromosomes.

It is clear from these results that the xaFCMs are almost d times faster than a FCM of the same

[1] All the assembled genome data were downloaded from ftp://ftp.ncbi.nlm.nih.gov/genomes/

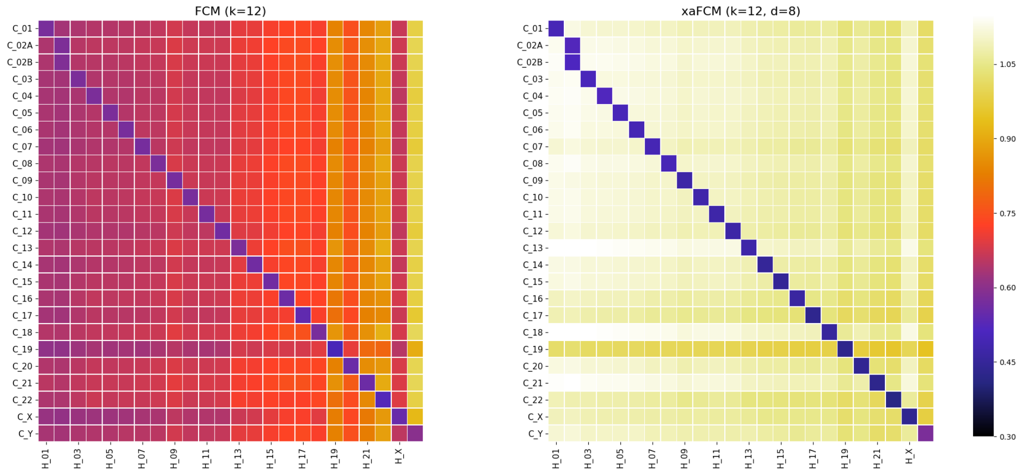


Figure 1: Normalized compression of the chimpanzee (C) chromosomes relatively to the Human (H), using: **(left)** FCM ($k = 12$); **(right)** xaFCM ($k = 12, d = 8$).

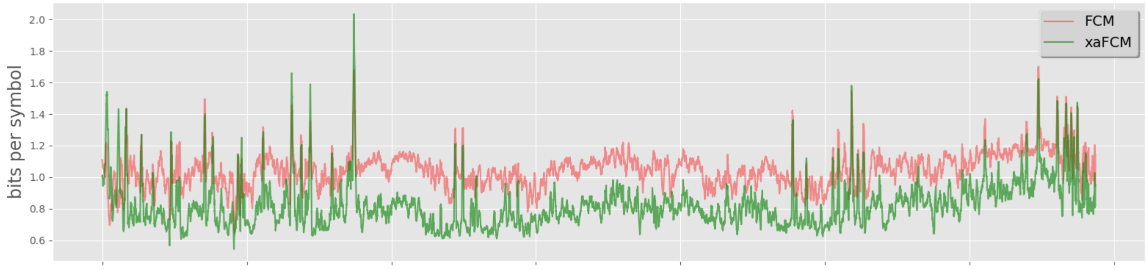


Figure 2: Profiles of information content of the chimpanzee chromosome 22 relatively to the human chromosome 22 using different models (FCM and xaFCM).

order k. Another advantage is that the memory needed for the xaFCMs does not increase exponentially with d.

The NRC results for the two simulations, with $k = 12$, can be seen in Fig. 1.

It is possible to notice that the heatmap corresponding to the FCM shows better compressions on average. However, using the "perfect" relative compressor, we would expect the NRC values to be as low as possible on the diagonal of the matrix[1], since they represent related chromosomes. The other squares should have higher NRC values, as they have more variation. This is exactly what happens on the xaFCMs test (bottom one in Fig. 1).

This becomes even more clear when we are comparing the compression along the same sequence, as can be seen in Fig. 2.

[1] Not exactly the diagonal, because of the second chromosome of the chimpanzee is split into 2a and 2b, making the matrix not a square one.

Discussion

We have shown that xaFCMs, when compared with FCMs with the same memory usage, tend to obtain better accuracy ratios, using up to around 3-4 times less time to compute the NRC (depending on the choice of *d*) – an obvious advantage when dealing with large sequences of data. Our experiments show that it is possible to use them for DNA sequence pattern recognition, making them a suitable alternative to the traditional FCMs.

These are promising results, and it seems appropriate to infer that the xaFCMs can be suitable to some other applications where FCMs have proven suitable and memory usage or testing speed are crucial, such as biometric identification using the ECG [17, 121, 37], image pattern recognition [8, 69, 122] and authorship attribution [64].

Bibliography

[1] Khalid. Sayood. *Introduction to data compression.* Morgan Kaufmann Publishers, 2000, p. 636. ISBN: 1558605584.

[2] Yosra Hajjaji, Wadii Boulila, Imed Riadh Farah, et al. "Big data and IoT-based applications in smart environments: A systematic review". In: *Computer Science Review* 39 (2021), p. 100318. DOI: 10.1016/j.cosrev.2020.100318.

[3] Eamonn Keogh, Stefano Lonardi, and Chotirat Ann Ratanamahatana. "Towards parameter-free data mining". In: *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '04.* ACM Press, 2004. DOI: 10.1145/1014052.1014077.

[4] M.W Gardner and S.R Dorling. "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences". In: *Atmospheric Environment* 32.14-15 (1998), pp. 2627–2636. DOI: 10.1016/s1352-2310(97)00447-0.

[5] Les Atlas, Toshiteru Homma, and Robert Marks. "An Artificial Neural Network for Spatio-Temporal Bipolar Patterns: Application to Phoneme Classification". In: *Neural Information Processing Systems.* Ed. by D. Anderson. American Institute of Physics, 1988. URL: https://proceedings.neurips.cc/paper/1987/file/98f13708210194c475687be6106a3b84-Paper.pdf.

[6] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. "Learning Internal Representations by Error Propagation". In: *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations.* Ed. by David E. Rumelhart and James L. Mcclelland. Cambridge, MA: MIT Press, 1986, pp. 318–362.

[7] Michael I. Jordan. "Serial Order: A Parallel, Distributed Processing Approach". In: *Advances in Connectionist Theory: Speech.* Ed. by Jeffrey L. Elman and David E. Rumelhart. Hillsdale, NJ: Erlbaum, 1989.

[8] Armando J. Pinho and Paulo Jorge S. G. Ferreira. "Finding unknown repeated patterns in images". In: *Proceedings of the 19th European Signal Processing Conference, EUSIPCO 2011, Barcelona, Spain, August 29 - Sept. 2, 2011.* IEEE, 2011, pp. 584–588. URL: https://ieeexplore.ieee.org/document/7074121/.

[9] Armando J. Pinho, Diogo Pratas, and Paulo J. S. G. Ferreira. "Information Profiles for DNA Pattern Discovery". In: *2014 Data Compression Conference.* IEEE, 2014. DOI: 10.1109/dcc.2014.54.

[10] Diogo Pratas and Armando J. Pinho. "A Conditional Compression Distance that Unveils Insights of the Genomic Evolution". In: *2014 Data Compression Conference.* IEEE, 2014. DOI: 10.1109/dcc.2014.58.

[11] D Pereira Coutinho and A. T. Figueiredo. "Text Classification Using Compression-Based Dissimilarity Measures". In: *International Journal of Pattern Recognition and Artificial Intelligence* 29.05 (2015), p. 1553004. DOI: 10.1142/s0218001415530043.

[12] Diogo Pratas, Raquel M Silva, Armando J. Pinho, et al. "An alignment-free method to find and visualise rearrangements between pairs of DNA sequences." en. In: *Nature* (2015). ISSN: 2045-2322. DOI: 10.1038/srep10203. URL: http://www.nature.com/srep/2015/150518/srep10203/full/srep10203.html.

[13] M. Li, X. Chen, X. Li, et al. "The Similarity Metric". In: *IEEE Transactions on Information Theory* 50.12 (2004), pp. 3250–3264. DOI: 10.1109/tit.2004.838101.

[14] Nima Karimian, Damon Woodard, and Domenic Forte. "ECG Biometric: Spoofing and Countermeasures". In: *IEEE Transactions on Biometrics, Behavior, and Identity Science* 2.3 (2020), pp. 257–270. DOI: 10.1109/tbiom.2020.2992274.

[15] Jessica Lin, Eamonn Keogh, Stefano Lonardi, et al. "A symbolic representation of time series, with implications for streaming algorithms". In: *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery, DMKD 2003, San Diego, California, USA, June 13, 2003.* Ed. by Mohammed Javeed Zaki and Charu C. Aggarwal. ACM, 2003, pp. 2–11. DOI: 10.1145/882082.882086. URL: https://doi.org/10.1145/882082.882086.

[16] João P. Silva Cunha, Bernardo Cunha, António Sousa Pereira, et al. "Vital-Jacket: A wearable wireless vital signs monitor for patients' mobility in cardiology and sports". In: *2010 4th International Conference on Pervasive Computing Technologies for Healthcare.* IEEE, 2010, pp. 1–2. DOI: 10.4108/ICST.PERVASIVEHEALTH2010.8991.

[17] Susana Brás and Armando J. Pinho. "ECG biometric identification: A compression based approach". In: *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC).* 2015, pp. 5838–5841. DOI: 10.1109/EMBC.2015.7319719. URL: http://www.ncbi.nlm.nih.gov/pubmed/26737619.

[18] Jacqueline Ferreira, Susana Brás, Carlos F. Silva, et al. "An automatic classifier of emotions built from entropy of noise". In: *Psychophysiology* (2016). ISSN: 00485772. DOI: 10.1111/psyp.12808. URL: http://doi.wiley.com/10.1111/psyp.12808.

[19] Ary L Goldberger, Luis A N Amaral, Leon Glass, et al. "PhysioBank, PhysioToolkit, and PhysioNet". In: *Circulation* 101.23 (2000), e215–e220. DOI: 10.1161/01.CIR.101.23.e215. URL: https://www.ahajournals.org/doi/abs/10.1161/01.CIR.101.23.e215.

[20] R. Bousseljot, D. Kreiseler, and A. Schnabel. "Nutzung der EKG-Signaldatenbank CARDIODAT der PTB über das Internet". In: *Biomedizinische Technik/Biomedical Engineering* (2009), pp. 317–318. DOI: 10.1515/bmte.1995.40.s1.317.

[21] Foteini Agrafioti, Francis M. Bui, and Dimitrios Hatzinakos. "Medical biometrics in mobile health monitoring". In: *Security and Communication Networks* 4.5 (2010), pp. 525–539. DOI: 10.1002/sec.227.

[22] Foteini Agrafioti, Francis M. Bui, and Dimitrios Hatzinakos. "Medical biometrics: The perils of ignoring time dependency". In: *2009 IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems.* IEEE, 2009. DOI: 10.1109/btas.2009.5339042.

[23] S. Zahra Fatemian, Foteini Agrafioti, and Dimitrios Hatzinakos. "HeartID: Cardiac biometric recognition". In: *2010 Fourth IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS).* IEEE, 2010. DOI: 10.1109/btas.2010.5634493.

[24] Yuxi Liu and Dimitrios Hatzinakos. "Earprint: Transient Evoked Otoacoustic Emission for Biometrics". In: *IEEE Transactions on Information Forensics and Security* 9.12 (2014), pp. 2291–2301. DOI: 10.1109/tifs.2014.2361205.

[25] Shahrzad Pouryayevali, Saeid Wahabi, Siddarth Hari, et al. "On establishing evaluation standards for ECG biometrics". In: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, 2014. DOI: 10.1109/icassp.2014.6854307.

[26] Saeid Wahabi, Shahrzad Pouryayevali, Siddarth Hari, et al. "On Evaluating ECG Biometric Systems: Session-Dependence and Body Posture". In: *IEEE Transactions on Information Forensics and Security* 9.11 (2014), pp. 2002–2013. ISSN: 1556-6013. DOI: 10.1109/TIFS.2014.2360430. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6910235.

[27] Mariana S. Ramos, João M. Carvalho, Armando J. Pinho, et al. "On the Impact of the Data Acquisition Protocol on ECG Biometric Identification". In: *Sensors* 21.14 (2021), p. 4645. DOI: 10.3390/s21144645.

[28] Christian Tronstad, Gorm Krogh Johnsen, Sverre Grimnes, et al. "A study on electrode gels for skin conductance measurements". In: *Physiological Measurement* 31.10 (2010), pp. 1395–1410. DOI: 10.1088/0967-3334/31/10/008.

[29] Tiago Bastos. "Vitals recorder: Sistema móvel para apoiar a realização de estudos de psicofisiologia". In: (2018), pp. 1–33. DOI: http://hdl.handle.net/10773/25979. URL: https://ria.ua.pt/bitstream/10773/25979/1/DISSERTACAO{_}71770.pdf.

[30] Klaus-Peter Hoffmann and Roman Ruff. "Flexible dry surface-electrodes for ECG long-term monitoring". In: *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2007. DOI: 10.1109/iembs.2007.4353650.

[31] Foteini Agrafioti and Dimitrios Hatzinakos. "ECG biometric analysis in cardiac irregularity conditions". In: *Signal, Image and Video Processing* 3.329 (2009). DOI: https://doi.org/10.1007/s11760-008-0073-4.

[32] Filipa Barros, Cláudia Figueiredo, Susana Brás, et al. "Multidimensional assessment of anxiety through the State-Trait Inventory for Cognitive and Somatic Anxiety (STICSA): From dimensionality to response prediction across emotional contexts". In: *PLOS ONE* 17.1 (2022). Ed. by Claudio Imperatori, e0262960. DOI: 10.1371/journal.pone.0262960.

[33] Gisela Pinto, João M. Carvalho, Filipa Barros, et al. "Multimodal Emotion Evaluation: A Physiological Model for Cost-Effective Emotion Classification". In: *Sensors* 20.12 (2020), p. 3510. ISSN: 1424-8220. DOI: 10.3390/s20123510. URL: http://dx.doi.org/10.3390/s20123510.

[34] Susana Brás, João M. Carvalho, Filipa Barros, et al. "An Information-Theoretical Method for Emotion Classification". In: *IFMBE Proceedings*. Springer International Publishing, 2019, pp. 253–261. DOI: 10.1007/978-3-030-31635-8_30.

[35] João M. Carvalho, Susana Brás, and Armando J. Pinho. "Compression-Based Classification of ECG Using First-Order Derivatives". In: *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. Springer International Publishing, 2019, pp. 27–36. DOI: 10.1007/978-3-030-16447-8_3.

[36] João M. Carvalho, Susana Brás, Diogo Pratas, et al. "Extended-alphabet finite-context models". In: *Pattern Recognition Letters* 112 (2018), pp. 49–55. ISSN: 0167-8655. DOI: 10.1016/J.PATREC.2018.05.026. URL: https://www.sciencedirect.com/science/article/pii/S0167865518302095.

[37] João M. Carvalho, Susana Brás, Jacqueline Ferreira, et al. "Impact of the Acquisition Time on ECG Compression-Based Biometric Identification Systems". In: *Pattern Recognition and Image Analysis. IbPRIA 2017. Lecture Notes in Computer Science, vol 10255. Springer, Cham.* Springer, Cham, 2017, pp. 169–176. DOI: 10.1007/978-3-319-58838-4_19. URL: http://link.springer.com/10.1007/978-3-319-58838-4_19.

[38] João M. Carvalho, Armando J. Pinho, and Susana Brás. "An Exploratory Study on ECG Biometric Bias Using Compression Algorithms". In: *Proceedings of the 27th RecPad.* 2021.

[39] João M. Carvalho, Susana Brás, and Armando J. Pinho. "ECG Biometric Identification using Relative Compression". In: *4th Workshop on Maximum Entropy and its Applications.* Department of Mathematics, University of Aveiro, Portugal. July 2021.

[40] João M. Carvalho, Susana Brás, and Armando J. Pinho. "Compression-based Tools for Non-Symbolic Data". In: *Research Summit 2021.* University of Aveiro, Portugal. 2021.

[41] João Carvalho, Susana Brás, and Armando J. Pinho. "Entropy-Based ECG Biometric Identification". In: *Proceedings of Entropy 2021: The Scientific Tool of the 21st Century.* MDPI, 2021. DOI: 10.3390/entropy2021-09795.

[42] João M. Carvalho, Susana Brás, and Armando J. Pinho. "Myocardial Infarction Detection Using a Compression-Based Approach". In: *Special Issue - Statistics on Health Decision Making: state of the art.* Vol. 1. 1. May 2019.

[43] Filipa Barros, C. Costa, C. Figueiredo, et al. "Cognitive and somatic trait anxiety: subjective and psychophysiological responses." In: *27th European Congress of Psychiatry, Warsaw, Poland,* Apr. 2019.

[44] Filipa Barros, Cláudia Figueiredo, Susana Brás, et al. "The role of somatic trait anxiety and alexithymia on the physiological and subjective response to emotional stimuli". In: *19th WPA World Congress of Psychiatry.* Aug. 2019.

[45] João M. Carvalho, Susana Brás, Jacqueline Ferreira, et al. "Impact of the Acquisition Time on ECG Compression-based Biometric Identification Systems". In: *Encontro com a Ciência e Tecnologia em Portugal, Centro de Congressos de Lisboa.* June 2017.

[46] João M. Carvalho, Susana Brás, Jacqueline Ferreira, et al. "Impact of the Acquisition Time on ECG Compression-based Biometric Identification Systems". In: *Research Day 2017.* University of Aveiro, Portugal. July 2017.

[47] João M. Carvalho, Susana Brás, and Armando J. Pinho. "The Impact of Noise Removal on a Compression-based ECG Biometric Identification System". In: *Proceedings of the 23rd RecPad.* 2017.

[48] João M. Carvalho, Armando J. Pinho, and Susana Brás. "Irregularity Detection in ECG signal using a semi-fiducial method". In: *Proceedings of the 22nd RecPad.* 2016, pp. 75–76.

[49] Geoffrey Grimmett David Stirzaker. *Probability and Random Processes.* Oxford University Press, July 3, 2020. 688 pp. URL: https://www.ebook.de/de/product/39317453/david_stirzaker_geoffrey_grimmett_probability_and_random_processes.html.

[50] Stanley Rabinowitz and Stan Wagon. "A Spigot Algorithm for the Digits of pi". In: *The American Mathematical Monthly* 102.3 (1995), p. 195. DOI: 10.2307/2975006.

[51] Ming Li and Paul Vitányi. *An introduction to Kolmogorov complexity and its applications.* 3rd. Springer, 1997. ISBN: 9780387339986. DOI: 10.1016/S0898-1221(97)90213-3.

[52] C.H. Bennett, P. Gacs, Ming Li, et al. "Information distance". In: *IEEE Transactions on Information Theory* 44.4 (1998), pp. 1407–1423. DOI: 10.1109/18.681318.

[53] R. Cilibrasi and P.M.B. Vitanyi. "Clustering by Compression". In: *IEEE Transactions on Information Theory* 51.4 (2005), pp. 1523–1545. DOI: 10.1109/tit.2005.844059.

[54] Stephanie Wehner. "Analyzing Worms and Network Traffic Using Compression". In: *J. Comput. Secur.* 15.3 (2007), 303–320. ISSN: 0926-227X.

[55] Efstathios Stamatatos. "A survey of modern authorship attribution methods". In: *Journal of the American Society for Information Science and Technology* 60.3 (2009), pp. 538–556. DOI: 10.1002/asi.21001.

[56] M. Nykter, N. D. Price, M. Aldana, et al. "Gene expression dynamics in the macrophage exhibit criticality". In: *Proceedings of the National Academy of Sciences* 105.6 (2008), pp. 1897–1900. DOI: 10.1073/pnas.0711525105.

[57] Andrew R Cohen, Francisco L A F Gomes, Badrinath Roysam, et al. "Computational prediction of neural progenitor cell fates". In: *Nature Methods* 7.3 (2010), pp. 213–218. DOI: 10.1038/nmeth.1424.

[58] Matti Nykter, Nathan D. Price, Antti Larjo, et al. "Critical Networks Exhibit Maximal Information Diversity in Structure-Dynamics Relationships". In: *Physical Review Letters* 100.5 (2008). DOI: 10.1103/physrevlett.100.058702.

[59] Anton Bardera, Miquel Feixas, Imma Boada, et al. "Compression-based Image Registration". In: *2006 IEEE International Symposium on Information Theory.* IEEE, 2006. DOI: 10.1109/isit.2006.261706.

[60] Xian Zhang, Yu Hao, Xiaoyan Zhu, et al. "Information distance from a question to an answer". In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '07.* ACM Press, 2007. DOI: 10.1145/1281192.1281285.

[61] J. Ziv and A. Lempel. "A universal algorithm for sequential data compression". In: *IEEE Transactions on Information Theory* 23.3 (1977), pp. 337–343. DOI: 10.1109/tit.1977.1055714.

[62] J. Ziv and A. Lempel. "Compression of individual sequences via variable-rate coding". In: *IEEE Transactions on Information Theory* 24.5 (1978), pp. 530–536. DOI: 10.1109/tit.1978.1055934.

[63] J. Ziv and N. Merhav. "A measure of relative entropy between individual sequences with application to universal classification". In: *IEEE Transactions on Information Theory* 39.4 (1993), pp. 1270–1279. DOI: 10.1109/18.243444.

[64] Armando J. Pinho, Diogo Pratas, and Paulo J. S. G. Ferreira. "Authorship attribution using relative compression". In: *Data Compression Conference*. 2016. DOI: 10.1109/DCC.2016.53.

[65] Armando J. Pinho, Antonio J. R. Neves, Daniel A. Martins, et al. "Finite-Context Models for DNA Coding". In: *Signal Processing*. Ed. by Sebastian Miron. Rijeka: IntechOpen, 2010. Chap. 6. DOI: 10.5772/8539. URL: https://doi.org/10.5772/8539.

[66] Diogo Pratas. "Compression and analysis of genomic data". PhD thesis. Universidade de Aveiro, 2016. DOI: http://hdl.handle.net/10773/16286.

[67] S.P. Garcia, J.M.O.S. Rodrigues, S. Santos, et al. "A Genomic Distance for Assembly Comparison Based on Compressed Maximal Exact Matches". In: *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 10.3 (2013), pp. 793–798. ISSN: 1545-5963. DOI: 10.1109/TCBB.2013.77. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6552202.

[68] Diogo Pratas, Armando J. Pinho, and Sara P. Garcia. "Computation of the Normalized Compression Distance of DNA Sequences Using a Mixture of Finite-Context Models". In: *Proceedings of the International Conference on Bioinformatics Models, Methods and Algorithms*. SciTePress - Science, and Technology Publications, 2012. DOI: 10.5220/0003780203080311.

[69] Armando J. Pinho and Paulo J. S. G. Ferreira. "Image similarity using the normalized compression distance based on finite context models". In: *2011 18th IEEE International Conference on Image Processing*. IEEE, 2011. DOI: 10.1109/icip.2011.6115866.

[70] Susana Brás, Sandra C Soares, Ricardo Moreira, et al. "BeMonitored: Monitoring psychophysiology and behavior using Android in phobias". In: *Behavior research methods* (2015), pp. 1–9. ISSN: 1554-3528. DOI: 10.3758/s13428-015-0633-9. URL: http://www.ncbi.nlm.nih.gov/pubmed/26174715.

[71] Jose Venegas and Roger Mark. *HST.542J Quantitative Physiology: Organ Transport Systems*. Massachusetts Institute of Technology: MIT OpenCourseWare. URL: https://ocw.mit.edu/courses/health-sciences-and-technology/hst-542j-quantitative-physiology-organ-transport-systems-spring-2004/.

[72] Gari D. Clifford, Francisco Azuaje, and Patrick McSharry. *Advanced Methods And Tools for ECG Data Analysis*. USA: Artech House, Inc., 2006. ISBN: 1580539661.

[73] G. Stouffer, M.S. Runge, C. Patterson, et al. *Netter's Cardiology E-Book*. Netter Clinical Science. Elsevier Health Sciences, 2018. ISBN: 9780323547277. URL: https://books.google.pt/books?id=fRNkDwAAQBAJ.

[74] T. W. Shen, W. J. Tompkins, and Y. H. Hu. "One-lead ECG for identity verification". In: *Proceedings of the Second Joint 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society] [Engineering in Medicine and Biology*. Vol. 1. 2002, 62–63 vol.1. DOI: 10.1109/IEMBS.2002.1134388.

[75] Leif Sörnmo and Pablo Laguna. "Chapter 6 - The Electrocardiogram—A Brief Background". In: *Bioelectrical Signal Processing in Cardiac and Neurological Applications*. Ed. by Leif Sörnmo and Pablo Laguna. Biomedical Engineering. Burlington: Academic Press, 2005, pp. 411 –452. ISBN: 978-0-12-437552-9. DOI: https://doi.org/10.1016/B978-012437552-9/50006-4. URL: http://www.sciencedirect.com/science/article/pii/B9780124375529500064.

[76] F. Buenda-Fuentes, M. A. Arnau-Vives, A. Arnau-Vives, et al. "High-Bandpass Filters in Electrocardiography: Source of Error in the Interpretation of the ST Segment". In: *ISRN Cardiology* 2012 (2012), pp. 1–10. DOI: 10.5402/2012/706217.

[77] K.L. Venkatachalam, Joel E. Herbrandson, and Samuel J. Asirvatham. "Signals and Signal Processing for the Electrophysiologist". In: *Circulation: Arrhythmia and Electrophysiology* 4.6 (2011), pp. 965–973. DOI: 10.1161/circep.111.964304.

[78] Foteini Agrafioti, Jiexin Gao, and Dimitrios Hatzinakos. "Heart Biometrics: Theory, Methods and Applications". In: *Biometrics*. Ed. by Jucheng Yang. Rijeka: IntechOpen, 2011. Chap. 10. DOI: 10.5772/18113. URL: https://doi.org/10.5772/18113.

[79] Foteini Agrafioti. "ECG in Biometric Recognition: Time Dependency and Application Challenges". PhD thesis. Department of Electrical and Computer Engineering, University of Toronto, 2011.

[80] Leif Sörnmo and Pablo Laguna. "Chapter 7 - ECG Signal Processing". In: *Bioelectrical Signal Processing in Cardiac and Neurological Applications*. Ed. by Leif Sörnmo and Pablo Laguna. Biomedical Engineering. Burlington: Academic Press, 2005, pp. 453 –566. ISBN: 978-0-12-437552-9. DOI: https://doi.org/10.1016/B978-012437552-9/50007-6. URL: http://www.sciencedirect.com/science/article/pii/B9780124375529500076.

[81] K.L. Venkatachalam, Joel E. Herbrandson, and Samuel J. Asirvatham. "Signals and Signal Processing for the Electrophysiologist". In: *Circulation: Arrhythmia and Electrophysiology* 4.6 (2011), pp. 974–981. DOI: 10.1161/circep.111.964973.

[82] Franco Parola and Javier García-Niebla. "Use of High-Pass and Low-Pass Electrocardiographic Filters in an International Cardiological Community and Possible Clinical Effects". In: *Advanced Journal of Vascular Medicine* 2 (Dec. 2017), pp. 34–38.

[83] Steven A. Israel, John M. Irvine, Andrew Cheng, et al. "ECG to identify individuals". In: *Pattern Recognition* 38.1 (2005), pp. 133–142. DOI: 10.1016/j.patcog.2004.05.014.

[84] A. A. Fedotov. "Selection of Parameters of Bandpass Filtering of the ECG Signal for Heart Rhythm Monitoring Systems". In: *Biomedical Engineering* 50.2 (2016), pp. 114–118. DOI: 10.1007/s10527-016-9600-8.

[85] B. A. Shenoi. *Introduction to Digital Signal Processing and Filter Design*. 1st. Wiley-Interscience, Oct. 2005. ISBN: 978-0471464822.

[86] P. Kathirvel, M. Sabarimalai, S. R. M. Prasanna, et al. "An Efficient R-peak Detection Based on New Nonlinear Transformation and First-Order Gaussian Differentiator". In: *Cardiovascular Engineering and Technology* 2.4 (2011), pp. 408–425. ISSN: 1869-408X. DOI: 10.1007/s13239-011-0065-3. URL: http://link.springer.com/article/10.1007/s13239-011-0065-3/fulltext.html.

[87] Rafael Gonzalez C. and Richard E. Woods. *Sampling and Quantization*. Prentice Hall PTR, 2007. ISBN: 9780470870075. DOI: 10.1002/0470870109.ch3.

[88] Eamonn Keogh and Michael J. Pazzani. *Knowledge Discovery and Data Mining. Current Issues and New Applications*. en. Ed. by Takao Terano, Huan Liu, and Arbee L. P. Chen. Vol. 1805. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 122–133. ISBN: 978-3-540-67382-8. DOI: 10.1007/3-540-45571-X. URL: http://link.springer.com/chapter/10.1007/3-540-45571-X{_}14.

[89] S. Lloyd. "Least squares quantization in PCM". In: *IEEE Transactions on Information Theory* 28.2 (1982), pp. 129–137. DOI: 10.1109/tit.1982.1056489.

[90] J. Max. "Quantizing for minimum distortion". In: *IEEE Transactions on Information Theory* 6.1 (1960), pp. 7–12. DOI: 10.1109/tit.1960.1057548.

[91] Jessica Lin, Eamonn Keogh, Stefano Lonardi, et al. "Finding Motifs in Time Series". In: *Proceedings of the 2nd Workshop on Temporal Data Mining*. 2002.

[92] Richard Larsen. *An introduction to mathematical statistics and its applications*. Boston: Prentice Hall, 2012. ISBN: 9780321693945.

[93] Binoy B. Nair, V. P. Mohandas, Nikhil Nayanar, et al. "A Stock Trading Recommender System Based on Temporal Association Rule Mining". In: *SAGE Open* 5.2 (2015), p. 2158244015579941. DOI: 10.

1177/2158244015579941. eprint: https://doi.org/10.1177/2158244015579941. URL: https://doi.org/10.1177/2158244015579941.

[94] Zoran Peric and Jelena Nikolic. "An Effective Method for Initialization of Lloyd–Max's Algorithm of Optimal Scalar Quantization for Laplacian Source". In: *Informatica* 18.2 (2007), pp. 279–288. DOI: 10.15388/informatica.2007.177.

[95] Foteini Agrafioti and Dimitrios Hatzinakos. "ECG Based Recognition Using Second Order Statistics". In: *6th Annual Communication Networks and Services Research Conference (cnsr 2008)*. 2008, pp. 82–87. DOI: 10.1109/CNSR.2008.38.

[96] Ikenna Odinaka, Po-Hsiang Lai, Alan D. Kaplan, et al. "ECG Biometric Recognition: A Comparative Analysis". In: *IEEE Transactions on Information Forensics and Security* 7.6 (2012), pp. 1812–1824. ISSN: 1556-6013. DOI: 10.1109/TIFS.2012.2215324. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6287579.

[97] Denis Chabot, Max Bayer, and André de Roos. "Instantaneous heart rates and other techniques introducing errors in the calculation of heart rate". In: *Canadian Journal of Zoology* 69.4 (1991), pp. 1117–1120. DOI: 10.1139/z91-156.

[98] Nima Karimian, Paul A. Wortman, and Fatemeh Tehranipoor. "Evolving authentication design considerations for the internet of biometric things (IoBT)". In: *Proceedings of the Eleventh IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis - CODES '16*. 2016. DOI: 10.1145/2968456.2973748. URL: http://dl.acm.org/citation.cfm?doid=2968456.2973748.

[99] Yassin Kortli, Maher Jridi, Ayman Al Falou, et al. "Face Recognition Systems: A Survey". In: *Sensors* 20.2 (2020), p. 342. DOI: 10.3390/s20020342.

[100] Puja S. Prasad, B. Sunitha Devi, M. Janga Reddy, et al. "A Survey of Fingerprint Recognition Systems and Their Applications". In: *Lecture Notes in Electrical Engineering*. Springer Singapore, 2018, pp. 513–520. DOI: 10.1007/978-981-13-0212-1_53.

[101] Sunil. S. Harakannanavar and Veena I Puranikmath. "Comparative survey of iris recognition". In: *2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT)*. IEEE, 2017. DOI: 10.1109/iceeccot.2017.8284682.

[102] Sreenivas Sremath Tirumala, Seyed Reza Shahamiri, Abhimanyu Singh Garhwal, et al. "Speaker identification features extraction methods: A systematic review". In: *Expert Systems with Applications* 90 (2017), pp. 250–271. DOI: 10.1016/j.eswa.2017.08.015.

[103] Imed Bouchrika. "A Survey of Using Biometrics for Smart Visual Surveillance: Gait Recognition". In: *Surveillance in Action*. Springer International Publishing, 2017, pp. 3–23. DOI: 10.1007/978-3-319-68533-5_1.

[104] Jarina B Mazumdar and SR Nirmala. "Retina Based Biometric Authentication System: A Review". In: *International Journal of Advanced Research in Computer Science* 9.1 (2018). DOI: https://doi.org/10.26483/IJARCS.V9I1.5322.

[105] Rahul C Bakshe and AM Patil. "Hand geometry techniques: a review". In: *International Journal of Modern Communication Technologies and Research (IJMCTR)* 2.11 (2014), p. 265750. ISSN: 2321-0850.

[106] Emanuele Maiorana, Himanka Kalita, and Patrizio Campisi. "Mobile keystroke dynamics for biometric recognition: An overview". In: *IET Biometrics* (2021).

[107] Anwar Yahya Ebrahim, Hoshang Kolivand, Amjad Rehman, et al. "Features selection for offline handwritten signature verification: State of the art". In: *International Journal of Computational Vision and Robotics* 8.6 (2018), pp. 606–622.

[108] Shenda Hong, Yuxi Zhou, Junyuan Shang, et al. "Opportunities and challenges of deep learning methods for electrocardiogram data: A systematic review". In: *Computers in Biology and Medicine* 122 (2020), p. 103801. DOI: 10.1016/j.compbiomed.2020.103801.

[109] M Bassiouni and W Khalefa. "A study on the Intelligent Techniques of the ECG-based Biometric Systems". In: *Recent Advances in Electrical Engineering* (2015). URL: http://www.inase.org/library/2015/crete/COCI.pdf{\#}page=26.

[110] Foteini Agrafioti, Dimitrios Hatzinakos, and A. K. Anderson. "ECG Pattern Analysis for Emotion Detection". In: *IEEE Transactions on Affective Computing* 3.1 (2012), pp. 102–115. ISSN: 1949-3045. DOI: 10.1109/T-AFFC.2011.28. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5999653.

[111] Roberto Grossi, Costas S Iliopoulos, Robert Mercas, et al. "Circular sequence comparison: algorithms and applications". In: *Algorithms for molecular biology : AMB* 11 (2016), p. 12. DOI: 10.1186/s13015-016-0076-6. URL: http://www.ncbi.nlm.nih.gov/pubmed/27168761http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4862142.

[112] Warren Gay. *Raspberry Pi Hardware Reference.* Apress, Nov. 2014. 248 pp. ISBN: 1484208005. URL: https://www.ebook.de/de/product/23236482/warren_gay_raspberry_pi_hardware_reference.html.

[113] Christophe Giraud-Carrier. "A note on the utility of incremental learning". In: *AI Communications* 13.4 (2000), pp. 215–223.

[114] Shachar Kaufman, Saharon Rosset, Claudia Perlich, et al. "Leakage in data mining: Formulation, detection, and avoidance". In: *ACM Transactions on Knowledge Discovery from Data* 6.4 (2012), pp. 1–21. ISSN: 15564681. DOI: 10.1145/2020408.2020496. URL: http://dl.acm.org/citation.cfm?doid=2382577.2382579.

[115] Dana H. Ballard. "Modular Learning in Neural Networks". In: *Proceedings of the Sixth National Conference on Artificial Intelligence - Volume 1.* AAAI'87. Seattle, Washington: AAAI Press, 1987, 279–284. ISBN: 0934613427.

[116] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. "Learning Internal Representations by Error Propagation". In: *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations.* Cambridge, MA, USA: MIT Press, 1986, 318–362. ISBN: 026268053X.

[117] Armando J. Pinho, Diogo Pratas, and Paulo J. S. G. Ferreira. "Bacteria DNA sequence compression using a mixture of finite-context models". In: *IEEE Workshop on Statistical Signal Processing Proceedings* (2011), pp. 125–128. DOI: 10.1109/SSP.2011.5967637.

[118] Diogo Pratas and Armando J. Pinho. "Exploring deep Markov models in genomic data compression using sequence pre-analysis". In: *2014 22nd European Signal Processing Conference (EUSIPCO).* 2014, pp. 2395–2399.

[119] Diogo Pratas, Armando J. Pinho, and Paulo J. S. G. Ferreira. "Efficient compression of genomic sequences". In: *Data Compression Conference.* 2016. DOI: 10.1109/DCC.2016.60.

[120] A. Hobolth, J. Y. Dutheil, J. Hawks, et al. "Incomplete lineage sorting patterns among human, chimpanzee, and orangutan suggest recent orangutan speciation and widespread selection". In: *Genome Research* 21.3 (2011), pp. 349–356. ISSN: 1088-9051. DOI: 10.1101/gr.114751.110. URL: http://www.ncbi.nlm.nih.gov/pubmed/21270173http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC3044849http://genome.cshlp.org/cgi/doi/10.1101/gr.114751.110.

[121] Susana Brás, Jacqueline H. T. Ferreira, Sandra C. Soares, et al. "Biometric and Emotion Identification: An ECG Compression Based Method". In: *Frontiers in Psychology* 9 (2018), p. 467. ISSN: 1664-1078. DOI: 10.3389/fpsyg.2018.00467. URL: https://www.frontiersin.org/article/10.3389/fpsyg.2018.00467.

[122] Armando J. Pinho, Diogo Pratas, and Paulo J. S. G. Ferreira. "A New Compressor for Measuring Distances among Images". In: *Image Analysis and Recognition.* Ed. by Aurélio Campilho and Mohamed Kamel. Cham: Springer International Publishing, 2014, pp. 30–37. ISBN: 978-3-319-11758-4.