Laboratório de Modelagem, Análise e Controle de Sistemas Não-Lineares

Departamento de Engenharia Eletrônica

Universidade Federal de Minas Gerais

Av. Antônio Carlos 6627, 31270-901 Belo Horizonte, MG Brasil

Fone: +55 3499-4866 - Fax: +55 3499-4850

Feature Enrichment in Human Activity Recognition

Priscila Louise Ribeiro Aguirre

Dissertation submitted in partial fulfilment of the requirements for the degree of Master in Electrical Engineering in the Post Graduate Program in Electrical Engineering of the Federal University of Minas Gerais.

Supervisors: Prof. Leonardo Antônio Borges Torres, Ph.D.
Prof. André Paim Lemos, Ph.D.

Belo Horizonte, April 2018

Acknowlegments

I thank first and foremost my Lord Jesus, the Author of my salvation, through whom my sins are justified before God, who I can now call my Father.

I would like to express my joy and gratitude to my family for all the love and support, encouragement and fun times spent together. I am also thankful for my brothers and sisters in Christ, for the fellowship.

This experience has transported me from a simple deterministic context to a world of statistical uncertainties; it has matured my thinking as an engineer, broadened my view and equipped me with a new set of tools for problem solving; and above all, it has brought me back to working with data analysis and modelling, which although I much enjoy, I had not been able to work with in the previous years. For such an opportunity, I thank my supervisors, Leonardo and André. I likewise thank them for all their patience and willingness to help and teach me. I hereby express my sincere appreciation for guiding me thorough out this journey.

I wish to thank Samsung for the opportunity of being part of their HAR-Heath Project and for the financial support, as well as Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, CAPES. Likewise, I thank Itaú-Unibanco for granting me a day a week to work on my dissertation.

Part of the results presented in this dissertation were obtained through research on a project titled “HAR-HEALTH: Reconhecimento de Atividades Humanas associadas a Doenças Crônicas”, sponsored by Samsung Eletrônica da Amazônia Ltda. under the terms of Brazilian federal law No. 8.248/91.

"Praise God from Whom all blessings flow
Praise Him, all creatures here below
Praise Him above, ye heavenly hosts
Praise Father, Son and Holy Ghost
[. . .]
Praise to the King, His throne transcends
His crown and Kingdom never end
Now and throughout eternity
I'll praise the One Who died for me"

Doxology

Contents

Abstract

In the context of Ubiquitous Sensing, a field that investigates how to extract knowledge using pervasive sensors, the recognition of human activities is a task of great interest in surveillance, healthcare and entertainment systems. Although it has been explored for more than a decade, many issues, such as the collection of data under realistic conditions, the flexibility to support new users, implementations that meet processing and energy consumption requirements, etc, still motivate the development of new techniques in order to increase performance.

To the best of our knowledge, most researches consider recognition of activities that involve postural transitions, which are categorized as ambulation activities. Motivated mostly by the lack of studies which consider activities classified as daily living, which are more complex when compared with ambulation activities, this work investigates feature enrichment from wrist worn accelerometer data in order to refine the ability of recognizing such type of human activities, which tend to be rich in wrist and arm movement.

Based on the assumption that wrist orientation contains useful information that could increase recognition of daily life activities, we investigate the effects of applying a non-linear mathematical transformation to the tri-axial accelerometer signals in order to estimate wrist attitude roll and pitch angles. We employ, as features, coefficients of two-dimensional multivariate/vector autoregressive (AR) models obtained from raw acceleration signals and from estimated wrist attitude angles. It is shown that the simultaneous use of both types of models improves the overall accuracy about 10% when compared to recently published algorithms where only univariate AR models coefficients for each raw acceleration signal are employed.

Keywords: Human Activity Recognition; System Identification; Multivariate Autoregressive models; Euler's attitude angles; Machine Learning; Feature Enrichment; wrist-worn sensors; accelerometer.

Resumo

No contexto de Sensoriamento Ubíquo, uma área que investiga a extração de informação por meio de sensores pervasivos, o reconhecimento de atividade humanas é uma tarefa de grande interesse em sistemas de vigilância, assistência médica e entreterimento. Apesar de ter sido explorada por mais de uma década, muitas questões como a coleta de dados em condições realistas, a flexibiliade do sistema para novos usuários, implementações que atendem requesitos de processamento e de consumo de energia, etc, ainda motivam o desenvolvimento de novas técnicas para melhoria de performance de tais sistemas.

Ao nosso melhor conhecimento, a maioria dos trabalhos científicos na área consideram o reconhecimento de atividades que envolvem transições posturais, que são categorizados como atividades de locomoção. Motivados principalmente pela falta de estudos que consideram ativiades classificados como cotidianas, que são mais complexas quando comparadas com atividades de locomoção, esse trabalho investiga o enriquecimento de atributos de dados de acelerômetros usados no punho a fim de refinar a habilidade de reconhecimento de tal tipo de atividade humana, que são tipicamente ricos em movimentação de punho e braço.

Baseados no pressuposto que a orientação do pulso carrega informação útil que poderia melhorar reconhecimento de atividades do cotidiano, envestigamos os efeitos de transformações matematicas não-lineares aos sinais de acelerômetro triaxial para estimação dos ângulos de atitude, rolamento e guinada, do punho. Empregamos, como atributos, coeficientes de modelos autorregressivos (AR) multivariados bidimensionais obtidos de sinais crus de acelerômetro e dos ângulos estimados de atitude. Mostramos que o uso simultâneo de ambos os tipos de modelos melhora a acurácia geral em quase 10% quando comparado a algoritmos recentemente publicados nos quais empregou-se apenas coeficientes univariados de modelos AR a cada sinal de acelerômetro.

Palavras-chave: Reconhecimento de Atividade Humana; Identificação de Sistemas; Modelos Autorregressivos multivariados; Ângulos de Euler; Aprendizado de Máquina; Enriquecimento de atributos; sensores usados no punho; acelerômetros.

List of Figures

List of Tables

List of Symbols

Chapter 2.2

$u(k)$	Input signal of non-autonomous systems in time instant $k \in \mathbb{R}^1$;
$y(k)$	Output signal in time instance $k \in \mathbb{R}^l$;
$e(k)$	Additive noise in time instance $k \in \mathbb{R}^l$;
Z	Set of observed input and output data $\in \mathbb{R}^1$;
$\hat{\boldsymbol{\theta}}_{\mathrm{p}}$	Estimated parameters vector of transfer function $\in \mathbb{R}^{m\times 1}$;
$G(\cdot)$	Process transfer function;
$H(\cdot)$	Noise transfer function;
$\xi(k)$	Prediction error residues on time instant $k \in \mathbb{R}^1$;
$\hat{y}(k\|k-1)$	One-step ahead prediction on time instant $k \in \mathbb{R}^1$;
$\xi(k\|k-1,\hat{\boldsymbol{\theta}}_{\mathrm{p}})$	One-step ahead prediction error on time instant $k \in \mathbb{R}^1$;
arg min	Argument of the minimum;
$J(\cdot)$	Cost function;
$A(\cdot)$	Polynomial of autoregressive model;
$B(\cdot)$	Polynomial of autoregressive with exogenous inputs model;
n_i	Order of the polynomial associated to the i variable $\in \mathbb{R}^l$;
a_i	Autoregressive coefficient $\mathbb{R}^1$;
A_i	Autoregressive coefficient matrix $\mathbb{R}^{m\times m}$;
$\mathbf{e}(k)$	Additive noise vector in time instance $k \in \mathbb{R}^{m\times 1}$;
$\mathbf{y}(k)$	Output signal vector in time instance $k \in \mathbb{R}^{m\times 1}$;
$\hat{\boldsymbol{\Theta}}_{\mathrm{p}}$	Estimated parameters vector of transfer function $\in \mathbb{R}^{(m\,n_i)\times m}$;
$\boldsymbol{\psi}(k-1)$	Regressor vector in time instant $k-1 \in \mathbb{R}^{m\,n_i}$;
Ψ	Regressor matrix in time instant $k-1 \in \mathbb{R}^{(N-n_i)\times m\,n_i}$;
Y	Output signal matrix $\in \mathbb{R}^{(N-n_i)\times m}$;
$\mathrm{E}[\cdot]$	Mathematical expectation;
$\mathrm{Var}[\cdot]$	Variance;
$\mathrm{Cov}[\cdot]$	Covariance;

τ	Lag;
$r_x(\tau)$	Autocorrelation function of time series x;
$r_{xy}(\tau)$	Cross-correlation function between time series x and y;
$\delta(\tau)$	Kronecker's function;
$\overline{x(k)}$	Time average of $x(k)$;

Chapter 2.3

S	Training set;
n	Number of samples in set S;
p	Number of attributes;
$\mathbf{x}$	Attribute vector that describe a class $\in \mathbb{R}^p$;
y	Attribute's vector class label $\in \mathbb{Z}$;
$p(\cdot)$	Probability;
$p(\cdot\|\cdot)$	Conditional probability;
C_k	k^th class;
arg max	Arguments of the maxima;
k^*	Most probable class;
$d(\cdot)$	Discriminant function;
T	Test set;
$\hat{f}(\cdot)$	Estimator;
ϵ	Irreducible error;
V	Validation set;
n_c	Number of classes;
$\mathcal{P}$	Positive class;
$\mathcal{N}$	Negative class;
$\mathcal{R}_k$	Regions defined by discriminant function;
w	Normal vector to the hyperplane $\in \mathbb{R}^p$;
b	Bias term $\in \mathbb{R}^1$;
C	Regularization parameter in SVM model $\in \mathbb{C}$;
sgn	Sigmoidal function;
$\langle\cdot,\cdot\rangle$	Inner product;
α	Support vector coefficients;
$\psi(x)$	Feature space mapping function;

$\mathcal{K}$ Kernel function;
$\|\cdot\|$ Euclidian norm;
$\binom{\cdot}{\cdot}$ Binomial norm;

Chapter 3

$\vec{a}_{\mathrm{m}}$ Measured acceleration;
$\vec{a}_{\mathrm{t}}$ Translational acceleration $\in \mathbb{R}^3$;
$\vec{g}$ Gravity acceleration vector $\in \mathbb{R}^3$;
g_0 Gravity acceleration's magnitude $\in \mathbb{R}^1$;
$\vec{v}$ Measurement noise $\in \mathbb{R}^3$;
$\hat{\Xi}^m$ Gaussian Mixture Model of a motion primitive;
$\hat{\Xi}^g$ Gaussian Mixture Model of the gravity component;
$\hat{\Xi}^b$ Gaussian Mixture Model of the gravity component;
μ Mean average;
Σ Covariance matrix;
$d(\cdot,\cdot)$ Mahalanobis distance;
τ_m Probability threshold;
$|\cdot|$ Absolute value;
arcsin Inverse sine function;
$\tilde{a}_z$ Normalized component of signal a_z;
Φ Tilt angle $\in \mathbb{R}^m$;
Φ^* Tilt angle sample average $\in \mathbb{R}^1$;
arctan Inverse tangent function;
ϕ Euler angle of roll $\in \mathbb{R}^m$;
θ Euler angle of pitch $\in \mathbb{R}^m$;

List of Acronyms

AIC	Akaike's Information Criterion;
AR	Auto-Regressive Models
CBIC	Congresso Brasileiro de Inteligêcia Computacional;
CNN	Convolution Neural Networks;
CV	Cross-validation;
FN	False Negative;
FP	False Positive;
FPR	False Positive Rate;
GMM	Gaussian Mixture Models;
GMR	Gaussian Mixture Regression;
LOOCV	Leave-one-out cross-validation;
LS	Least-Squares;
MACSIN	Grupo de Modelagem, Análise e Controle de Sistemas Não-Lineares;
MCCV	Monte Carlo cross-validation;
MSE	Mean squared error;
OAA	One-against-All
OAO	One-against-One
SMA	Signal Magnitude Area;
SVM	Support Vector Machines;
TA	Tilt Average;
TN	True Negative;
TP	True Positive;
TNR	True Negative Rate;
TPR	True Positive Rate;
UFMG	Universidade Federal de Minas Gerais;
VAR	Vector Auto-Regressive Models;

Chapter 1

Introduction

We begin this work by describing the context and motivation in which it was conceived, as well as presenting some useful definitions and an overview of this dissertation.

1.1 Contextualization

In the past couple of decades, *Human Activity Recognition* (HAR) has sprouted as a field of high interest for applications that range from surveillance systems to healthcare, and it is also entering the field of entertainment systems and interactive applications.

Healthcare systems are usually related to patient monitoring and rehabilitation applications. It is often adopted to assess patients capability in performing, without assistance, activities of the daily living (ADL), defined by Dr. Sidney Kartz as an index of standardized measurement of a persons functional competence (Katz et al., 1970) and determine what type of long-time care and health coverage will be needed as one ages. Activity recognition in such cases is required so caregivers can keep track of the level of activities being performed in a regular basis by patients. The total level of daily physical activity holds a relationship not only with life quality but also with elderly's probability of falling (Gupta and Dallas, 2014).

The HAR field of study, which depends on extracting information from sensors data, has two main data sources: video based information (Poppe, 2010), and non-visual sensors' signals (Lara and Labrador, 2013), which are often classified as external and wearable sensors. The former refers to the devices placed in strategic fixed points in a room, and the latter refers to sensors that are attached to body of the user. Even though the use of cameras as external sensors has been widely researched and is especially suitable for security, the sensors approach to ubiquitous computing seems to be more promising. This is due to issues associated with video recordings, such as privacy, pervasiveness and complexity (Lara and Labrador, 2013).

External sensors are considered less intrusive in the context of pervasive sensing, but such systems still have limitations such as high cost associated to installation and maintenance and require users to interact with the system. Wearable sensors are rising

as the most adequate solution, for they are still less obtrusive than video and their associated processing techniques tend to require less computational power. Besides, it has been usual for people to carry devices, such as mobile phones and smartwatches, that have embedded various non-visual sensors, rendering the sensors data collection a potentially ubiquitous possibility (Shoaib et al., 2015).

Sensors measured attributes can usually be (1) environmental, such as temperature, humidity, pressure, sound, luminosity levels or GPS location; (2) related to user movements, often obtained from accelerometers and girometers; and (3) physiological signals that can range from heart rate and skin temperature to ECG signals.

With respect to the movements, even though there might be some sensor fusion to improve recognition performance, tri-axial accelerometers signals have been the commonly used sensor data for ambulation activities (e.g. walk, sit, climb stairs) (Allen et al., 2006; Bruno et al., 2013; He and Jin, 2008; Karantonis et al., 2006; Khan et al., 2008; Krassnig et al., 2010; Kratz et al., 2013). The highly fluctuating and oscillatory nature of this type of sensor signals renders it hard to be used to identify patterns from raw values (Lara and Labrador, 2013). For this reason, mathematically extracting attributes that belong to a so-called *feature space* in order promote separation and recognition of patterns is often employed in such systems. Features can be categorized as time-domain (mean, standard deviation, variance, correlation between axis, entropy, etc) and frequency-domain (Fourier Transform coefficients and Discrete Cosine Transform coefficients). Some other features found in literature are obtained by means of Principal Components Analysis, Linear Discriminant Analysis and Autoregressive Models (Lara and Labrador, 2013).

The work described in this dissertation is based on enhancement of features in order to refine recognition in the context of HAR problems, as defined in the following section.

1.2 The Human Activity Recognition Problem

Lara and Labrador (2013) defines the HAR problem as finding a mapping function such that windowed time series are evaluated and resulting scores are as similar as possible as the actual activity performed in such window. This is hard problem to be solved deterministically and therefore machine learning techniques are employed in activity recognition. The task of adequately identifying a proper temporal partition of the time series, i.e., a consistent data segmentation, is, however, non trivial, considering that in real case scenarios activity transitions are rather hazy and overlapping is sometimes unavoidable. Therefore, data segmentation poses as a challenge since a single window should be large enough to contain one activity transition. On top of that, the duration of a activity being carried out varies considerably depending on the type of activity.

According to the literature, there are seven different groups of activities, namely (i) ambulation (e.g. walking, running, sitting, laying, climbing stairs); (ii) transportation (e.g. riding a bus, driving, cycling); (iii) phone usage; (iv) daily activities (e.g. eating, drinking, brushing teeth, watching TV, vacuuming); (v) physical exercises (e.g. weight lifting, rowing, doing push ups); (vi) military (crawling, kneeling, situation assessment); (vii) upper body (e.g. chewing, speaking, moving head). The ideal position of accelerometer on user's body is strongly related to the application in question, i.e., the type of activities to be recognized. It is known that ambulation activities, which comprise postural changes, sensing devices positioned in waist or chest region is preferred, while for daily activities, which are rich in arm movement, wrist worn sensors are recommended (Lara and Labrador, 2013).

Another crucial factor in the study of a HAR problem that should be taken into consideration is the range of characteristics of the individuals in what concerns gender, age, height, weight and health conditions. Such pool of examples would guarantee a flexible model which supports new users (Lara and Labrador, 2013).

1.3 Motivation

As previously mentioned, accelerometers are probably the most frequently used sensors in HAR problems used to recognize ambulation activities. Reported accuracy results are often above 90%, and can go up to 98% (Khan et al., 2010c). However, it has been reported that the use of acceleration signals in the recognition of daily activities may lead to confusion due to arm movement (Lara and Labrador, 2013).

Moreover, it was noticed that more complex activities are usually not considered in most HAR works. Actually, it is very common to have accelerometer strapped either to the user's chest or located in a trouser pocket (Khan et al., 2008, 2010a,c), which mostly addresses ambulation activity recognition problems.

Considering the lack of works that actually deal with complex daily activities, which are rich in arm movements, this work aims at refining the recognition of this type of activities while also dealing with ambulation activities. Researchers have reported in (Ramos-Garcia et al., 2015; Dong et al., 2012) possible improvements in the overall accuracy when working with features that depend on attitude estimation of wrist-worn devices, because, in many cases, hand rotation carry useful information, as made evident in Figure 1.1. Following this approach, our research considers the feature extraction problem from wrist positioned accelerometer signals for recognition of daily activities that are rich in arm movements, by relying not only on raw acceleration signals, but also on estimated attitude angles.

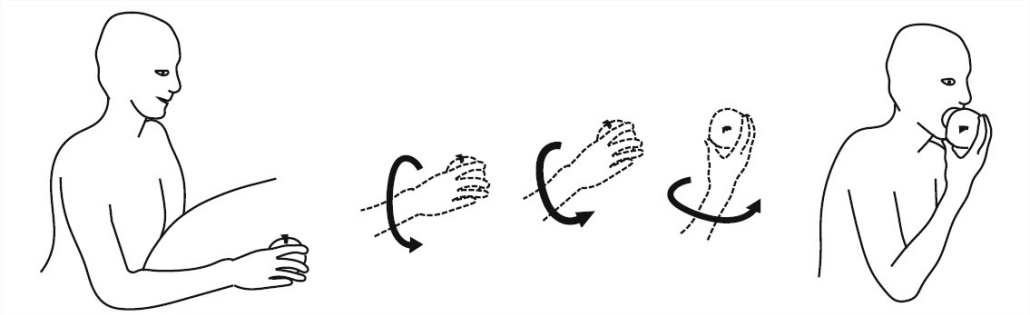


Figure 1.1: Wrist roll during taking a bite, from (Dong et al., 2012).

1.4 Objectives

Having the aforementioned context in mind, and motivated by the trend in the current literature, in the present work we aim:

1. To investigate if HAR accuracy of activities of daily living can be improved through wrist attitude estimation;

2. To investigate whether considering the temporal correlation between all signals from the accelerometer axes introduces useful information for recognition;

The main objective of this work is to propose a feature enrichment procedure associated with attitude estimation based on gravity projection. As a result of this procedure, a new set of features, derived from the accelerometer signals, can convey additional information to facilitate the human activity recognition by reducing intra-class variance and, at the same time, increasing inter-class variance, such that the data becomes more separable. In order to verify the new set of features effects in separability, we consider using a "less sophisticated" classification method, since using the state-of-the-art classification method in HAR might disguise the true feature enhancement ability of the proposed procedure.

1.5 Dissertation Outlook

The remainder of this dissertation is organized as follows. Chapter 2 presents conceptual and theoretical background on mathematical and statistical modelling, both for system identification and for machine learning, that will be necessary respectively for feature extraction and for validation of the resulting feature vector.

Following up, Chapter 3 reviews two cases of HAR problems in the literature that will be used as benchmarks in this study. The first reviewed work presents a study involving both ambulation and daily living activities and employees the same dataset

used in the present work. In the second one, some of the features extracted from accelerometer data are also used in the feature vector we propose in this work, by the end of Chapter 3.

In Chapter 4, the dataset used in this dissertation is described, as well as results reproduced from the chosen benchmarks. In that chapter all the feature extraction process proposed in this study is described as well as structural choices and results analysis.

Finally, Chapter 5 presents a retrospective overview of the dissertation along with future steps and possible new directions to enhance this study.

Chapter 2

Theoretical Foundation: Mathematical Modelling and Statistical Learning

In this chapter, the theoretical framework needed for this work is presented. We start in Section 2.1 by briefly considering model identification from time series, emphasising on autoregressive stochastic processes. We conclude with a brief introduction to statistical learning in Section 2.2, focusing on the supervised classification problem. Broadly speaking, both fields concern model estimation from available data, although the former usually applies to dynamical systems whilst the latter, to statistical learning. Thus, since conceptual and nomenclature overlapping is unavoidable, some key concepts will be clearly defined in the following sections.

2.1 System Identification

This section considers a couple of important definitions in the realm of stochastic processes, mathematical representations of linear systems and methods for parameter estimation. Aguirre (2015) and Pillonetto et al. (2014) were the primary references consulted on these subjects.

A general model structure for a linear time-invariant system with input $u(k)$ and output $y(k)$ affected by additive noise $e(k)$, which is a stochastic process, can be represented as

$$y(k) = G(q)u(k) + H(q)e(k) \tag{2.1}$$

where q is the shift operator and $G(\cdot)$ and $H(\cdot)$ are transfer functions, which are assumed to represent perfectly the process that generates the output $y(k)$ from the input $u(k)$.

2.1.1 Parameter Estimation

The parameter estimation problem in system identification refers to fitting a dynamic function to a given set of observed data $Z = \{u(1),y(1),\ldots,u(N),y(N)\}$ where $u(k)$ and $y(k)$ are observations of the input and respective output signals, in the instant k, of

a system or process we wish to model. This dynamic model can be written as

$$y(k) = G(q,\hat{\boldsymbol{\theta}}_{\mathrm{p}})u(k) + H(q,\hat{\boldsymbol{\theta}}_{\mathrm{p}})\xi(k), \tag{2.2}$$

where $\hat{\boldsymbol{\theta}}_p$ is the estimated parameters vector of the transfer function $G(q,\hat{\boldsymbol{\theta}}_{\mathrm{p}})$ and $H(q,\hat{\boldsymbol{\theta}}_{\mathrm{p}})$ and $\xi(k)$ is the prediction errors, referred to as *residues*. The residues carry all the information that the model was unable to express, be it due to process or measurement noise, or due to model's inappropriateness in describing the system. In the case of the one-step ahead prediction, the prediction in instant k estimated by information available until instant $k-1$ is indicated as $\hat{y}(k|k-1)$ and the one-step ahead prediction error is defined as

$$\xi(k|k-1,\hat{\boldsymbol{\theta}}_{\mathrm{p}}) = y(k) - \hat{y}(k|k-1). \tag{2.3}$$

If one wishes to minimize the expected quadratic prediction error, assuming that the expected value can be approximated by the quadratic prediction error mean value, the parameter estimation problem can be written as

$$\hat{\boldsymbol{\theta}}_{\mathrm{p}} = \arg\min_{\theta} J(\hat{\boldsymbol{\theta}}_{\mathrm{p}}) = \arg\min_{\theta} \frac{1}{N}\sum_{k=1}^{N} \xi(k|k-1,\hat{\boldsymbol{\theta}}_{\mathrm{p}})^2. \tag{2.4}$$

This optimization problem can be called *Prediction Error Method*, which coincides with the *Maximum Likelihood* method, if the noise source is Gaussian (Pillonetto et al., 2014).

Different representations of model (2.2) render different one-step ahead predictors, and different optimization criteria can require different parameter estimators. In the next section, we consider a common model representation in system identification and stochastic processes.

2.1.2 Autoregressive Models

A good approximation for linear time-invariant systems are autoregressive with exogenous input (ARX) models, in such a way that (2.1) can be expressed as

$$A(q)y(k) = B(q)u(k) + v(k) \tag{2.5}$$

where $A(q) = 1 + a_1 q^{-1} + \cdots + a_{n_a} q^{-n_a}$ and $B(q) = b_1 q^{-1} + \cdots + a_{n_b} q^{-n_b}$, and $v(k)$ is white noise signal. From Pillonetto et al. (2014), it is known that

$$\frac{B(q)}{A(q)} \to G(q), \quad \frac{1}{A(q)} \to H(q) \ \text{ as } \ n_a, n_b \to \infty. \tag{2.6}$$

For autonomous processes, where there are no inputs, the autoregressive (AR)

model becomes

$$y(k) = a_1 y(k-1) + a_2 y(k-2) + \cdots + a_{n_y} y(k-n_y) + e(k), \tag{2.7}$$

where $n_y = n_a$ is the process' order and $e(k) = \nu(k)/A(q)$, that is, the error is modelled as a white process filtered by an AR filter.

In the case of a multivariate system, we have that $\mathbf{y}(k) \in \mathbb{R}^{m\times 1}$, for which m is the number of measured outputs, and the autoregressive model can be referred to as *vector autoregressive model* (VAR), written as

$$\mathbf{y}(k) = A_1 \mathbf{y}(k-1) + \cdots + A_{n_y} \mathbf{y}(k-n_y) + \mathbf{e}(k) \tag{2.8}$$

for which $A_i \in \mathbb{R}^{m\times m}$ is the autoregressive coefficients matrix and $\mathbf{e}(k) \in \mathbb{R}^{m\times 1}$. The process (2.8) can be modelled as a linear regression problem such as

$$\mathbf{y}(k) = \boldsymbol{\Theta}_{\mathrm{p}}^{\mathrm{T}} \boldsymbol{\psi}(k-1) + \xi(k) \tag{2.9}$$

in which $\boldsymbol{\Theta}_{\mathrm{p}} = [A_1 \ \ldots \ A_{n_y}]^{\mathrm{T}}$ are the parameters to be estimated and $\boldsymbol{\psi}(k-1)$ is the regressor's vector, given by $\boldsymbol{\psi}(k-1) = [\mathbf{y}^{\mathrm{T}}(k-1) \ \ldots \ \mathbf{y}^{\mathrm{T}}(k-n_y)]$. In this case a single regressor vector is used to explain all m signals, such that $\boldsymbol{\Theta}_{\mathrm{p}} \in \mathbb{R}^{(m\,n_y)\times m}$ and $\boldsymbol{\psi}(k-1) \in \mathbb{R}^{m\,n_y}$. It is worth mentioning that a multivariate model can have other representations of the same dynamic order but with reduced number of parameters by attributing different regressor vectors to each of the m outputs.

The estimator that minimizes (2.4) for VAR models is the least-squares estimator, given by

$$\hat{\boldsymbol{\theta}}_{LS} = \left(\Psi^{\mathrm{T}}\Psi\right)^{-1} \Psi^{\mathrm{T}} Y \tag{2.10}$$

where $\Psi \in \mathbb{R}^{(N-n_y)\times m n_y}$ and $Y \in \mathbb{R}^{(N-n_y)\times m}$, in which

$$Y = \begin{bmatrix} \mathbf{y}_{n_y}^{\mathrm{T}}(k) \\ \mathbf{y}_{n_y+1}^{\mathrm{T}}(k) \\ \vdots \\ \mathbf{y}_{N}^{\mathrm{T}}(k) \end{bmatrix} \text{ and } \Psi = \begin{bmatrix} \boldsymbol{\psi}_{n_y-1}(k-1) \\ \boldsymbol{\psi}_{n_y}(k-1) \\ \vdots \\ \boldsymbol{\psi}_{N-1}(k-1) \end{bmatrix}.$$

2.1.3 Assessing Model Quality

The quality of a model estimated from a given set of data Z, which is referred to as *traning* set, is usually verified via simulation in a new set of data, called the *validation* set in the system identification context. In this context, to validate a model means to check if the model in question is general enough, that is, if it is able to explain new set of data observed from the same system operating under the same conditions.

As consequence of the parameter estimation based on algorithms that minimize the squared sum of the residues defined from a group of regressors, the one-step ahead prediction error will always be the smallest possible. However, one-step ahead predictors errors could not be good indicators of model's performance, specially when the identified model should be used for other purposes than one-step ahead predictions. Alternatively, the use of the error from a run-free prediction simulation, which starts with measured values and then reuses previous predictions values, sometimes can be a much better method for verifying model's generalization capability. However, since there is no closed form solution to the optimization problem associated with estimating parameters by minimizing the mean quadratic error in run free predictions, one can still consider using (2.10) to avoid significantly higher computation cost that would incur otherwise.

2.1.4 Residues Analysis

Residues analysis can be done in order to evaluate the estimator's performance and check if the one-step ahead simulation is statistically good. As for model validation, random and uncorrelated scalar residues, that is

$$\begin{aligned} \mathrm{E}[\xi(k)] &= 0, \\ \mathrm{Var}[\xi(k)] &= \sigma^2, \\ \mathrm{Cov}[\xi(k),\xi(j)] &= 0, \ \forall k \neq j, \end{aligned}$$

indicate that all explainable information from data was captured by the model. A set of values $\xi(k)$ is linearly white when its autocorrelation function is statistically null for all lag values different from zero, as expressed by

$$r_\xi(\tau) = \mathrm{E}\left[(\xi(k) - \overline{\xi(k)})(\xi(k-\tau) - \overline{\xi(k)})\right] = \delta(\tau) \tag{2.11}$$

In order to detect non-linearity, we consider the residues squared autocorrelation function, $r_\xi(\tau)$, and the residues and residues squared cross correlation function, $r_{\xi^2}(\tau)$, defined as

$$r_{\xi\xi^2}(\tau) = \mathrm{E}\left[(\xi(k) - \overline{\xi(k)})(\xi(k-\tau) - \overline{\xi(k)^2})\right] = 0 \tag{2.12}$$

$$r_{\xi^2}(\tau) = \mathrm{E}\left[(\xi^2(k) - \overline{\xi^2(k)})(\xi^2(k-\tau) - \overline{\xi^2(k)})\right] = \delta(\tau) \tag{2.13}$$

where $\overline{\xi(k)}$ is the time average of $\xi(k)$, $\delta(\tau)$ is Kronecker's function. For an ergodic process, the mathematical expectancy can be replaced by the temporal average.

Testing residues randomness is considered to be a good way of checking if a mathematical structure is complex enough to capture the deterministic dynamics in the

temporal evolution associated with a specific set of data. It does not, however, check model's generality.

2.2 Introduction to Statistical Learning

This section presents a brief introduction to statistical learning, considering James et al. (2013), Bishop (1995) and Suykens et al. (2002) as textbook references and overview material.

Broadly speaking, this field is based on searching, from given data, for models that are able to capture data structure organization, as in the clustering problem, or to predict target values from known attributes, be they continuous values, as in the regression problem, be they discrete, as in the classification problem. Even though most of the concepts presented in this section relate to both regression and classification problems, only the latter is explored.

2.2.1 Supervised Learning in Classification Problems

Supervised statistical learning consists of inferring functions that are able to predict outputs from given inputs. This can be achieved by finding statistical models which learn behaviours and patterns from known n samples of input-output pairs from a given training set $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, where $\mathbf{x}_i \in \mathbb{R}^p$ is a vector of attributes that describe a class and belongs to the input space, and $y_i \in \mathbb{Z}$ is a qualitative value indicating the class which the data point belongs to. The former is also referred to as the *feature vector* and the latter, as the *label*.

Thus, it is assumed that feature vectors and labels hold a relationship that can be generally described by

$$Y = f(X) + \epsilon, \tag{2.14}$$

where ϵ represents a random systematic and irreducible error term, with null mean and is uncorrelated to X. Therefore, the prediction problem seeks to estimate $\hat{f} : X \mapsto Y$ so that $\hat{Y} \approx \hat{f}(X)$, where $\hat{Y}$ is a prediction of Y. The estimate $\hat{f}(\cdot)$ introduces error that is directly correlated to model quality, which is discussed in Section 2.2.2.

For the classification problem, the searched model $\hat{f}(\cdot)$ is often associated with a decision boundary, that is, a rule that separates the feature space $\mathbb{R}^p$ into decision regions $\mathcal{R}_k$ so that each point in a region is assigned to a class C_k , $k = 1, \ldots, n_c$ where n_c, is the number of classes. By taking a Bayesian approach, the supervised classification problem can be expressed as

$$p(C_k \,|\, \mathbf{x}) = \frac{p(\mathbf{x} \,|\, C_k)\, p(C_k)}{p(\mathbf{x})}, \tag{2.15}$$

in which the prior $p(C_k)$ is the probability of having points belonging to the specific class C_k; the likelihood $p(\mathbf{x}\,|\,C_k)$ is the probability of a data point belonging to the given class; and the evidence $p(\mathbf{x})$ is the probability of observing a data point with respect to the whole set. The posterior probability $p(C_k\,|\,\mathbf{x})$ gives the probability of an observed data point $\mathbf{x}$ belonging to class C_k.

Hence, the classification process comprises two stages: (i) the inference stage refers to determining from data values the posterior probabilities, from which (ii) the decision making stage takes place by assigning new data points to the most probable possible class, such that

$$k^* = \underset{k=1,\ldots,n_c}{\arg\max}\ p(C_k\,|\,\mathbf{x}_i). \tag{2.16}$$

The decision of class membership can be made by simple comparison of probability magnitude so that $y_i = C_{k^*}$, or by reformulating the classification process by adding discriminant functions such that $y_i = d(\mathbf{x}_i)$, where $d(\mathbf{x}_i) \propto p(\mathbf{x}\,|\,C_k)\,p(C_k)$.

The performance of a model fitted to the given training set S can be evaluated by comparison with its predicted results, $\hat{f}(\mathbf{x}_j)$, and true label, y_j, in which $(\mathbf{x}_j,y_j) \in T$ where $T = \{(\mathbf{x}_{n+1},y_{n+1}),\ldots,(\mathbf{x}_N,y_N)\}$ is a given test set composed of data not used during the training of the fitted model. The "test set" in machine learning corresponds to the "validation set" in the context of system identification (Section 2.1).

2.2.2 Assessing Model Quality

Machine learning algorithms usually consist of two optimization problems. The first is referred to as *model selection*, in which parameters that configure the chosen mathematical representation, known as *hyper-parameters*, must be appropriately set prior to model training in order to maximize model generalization.

Once the set of hyper-parameters is defined, a model is then fit to a finite sample of data by solving a (hopefully convex) optimization problem. Thus, a model is trained by estimating its parameters such that a predefined loss function is minimized. Performance evaluation of the obtained model can be referred to as *model assessment*.

The Bias-Variance Trade-off

Hyper-parameters usually control model's complexity and directly affect the model and its performance. The error associated to an estimator $\hat{f}(\cdot)$ obtained in model selection is given in terms of bias, variance and irreducible error resulting from process and measurement uncertainties. The bias of $\hat{f}(\mathbf{x}_i)$ quantifies the error introduced by restricting a complex, real world problem to a simple model. On the other hand, the variance of $\hat{f}(\mathbf{x}_i)$ indicates the estimated model's sensitivity to the data set used for training, since it is fitted based on a specific finite sample of data. Mathematically, the

expected squared error of the estimator can be expressed by

$$\mathrm{E}\left[y_i - \hat{f}(\mathbf{x}_i)\right]^2 = \mathrm{Var}(\hat{f}(\mathbf{x}_i)) + \left[\mathrm{Bias}(\hat{f}(\mathbf{x}_i))\right]^2 + \mathrm{Var}(\epsilon), \tag{2.17}$$

that refers to the average error obtained if $\hat{f}(\cdot)$ were repeatedly estimated over a large training data set. And the overall error rate is obtained by averaging (2.17) over all possibilities of $\mathbf{x}_i$ in the test set. Usually, as flexibility and complexity of a statistical learning method increases, so does the variance, whereas the bias decreases. The opposite effect of bias and variance on the error rate are respectively described by the purple and green curves in Figure 2.1.

As model flexibility increases, the statistical learning algorithm begins to learn undesired information, that is, the algorithm begins to understand as relevant information the irreducible error. This is called *overfitting* and corresponds to the high variance scenario. The name given to the high bias scenario is *underfitting*, in which the model is too simple to capture the data's patterns and trends, as depicted in Figure 2.1.

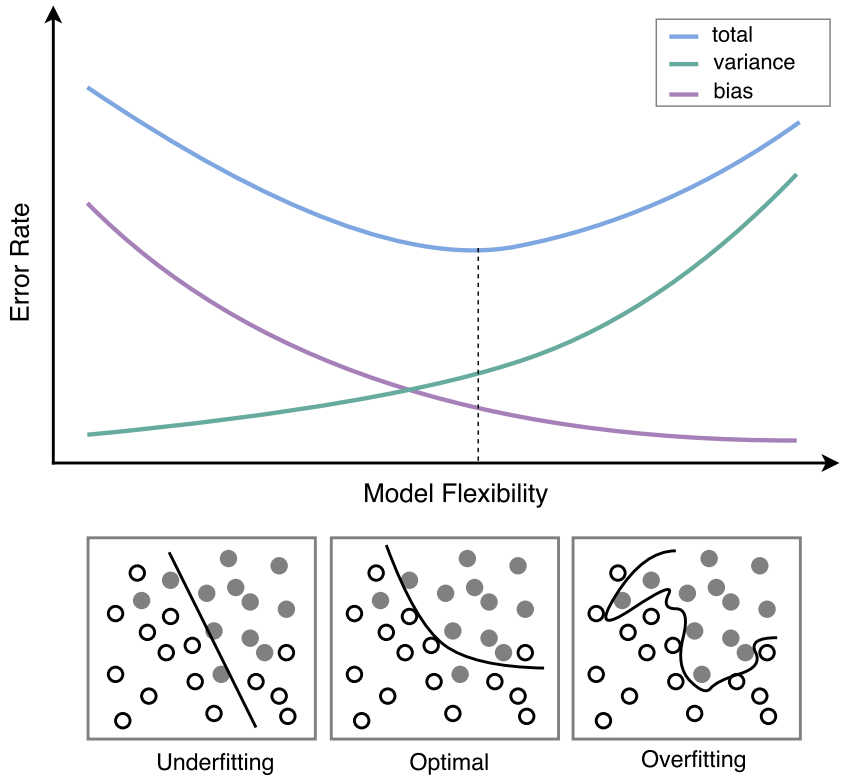


Figure 2.1: The bias-variance trade-off. High bias tends to result in model *underfitting* and high variance, *overfitting*. Minimum error rate renders the best fitting of the model.

In order to minimize the testing error, a statistical learning method that renders simultaneously low bias and low variance is needed. In practical terms, a model with

low variance is desired so that small changes in the training data set does not represent large changes in the fitted model, as well as a low bias is desired, for it indicates a good mathematical structure approximation to the considered problem. However, since variance and bias have opposite behaviours, a compromise must be achieved, which leads to the trade-off concept associated with bias and variance. The total expected error, therefore, exhibits a U-shape characteristic, decreasing at first, reaching a minimum and finally increasing again, as shown by the blue curve in Figure 2.1.

However, true test error rate is generally unknown in model selection since there is no knowledge of testing data at this point. For such reasons, an extra subset, called validation set V, is obtained by partitioning the training set S, such that $V \subset S$. The validation set is then used for model assessment during the model selection stage in order to estimate the test error rate. As shown in Figure 2.2, parameter tuning process can take place by minimizing the error with validation data, for the smallest error in validation indicates optimal choice of hyper-parameters, assuming that it provides a good estimate of the test error rate. Once model selection is complete, all data in S is then used for training the parametrized model.

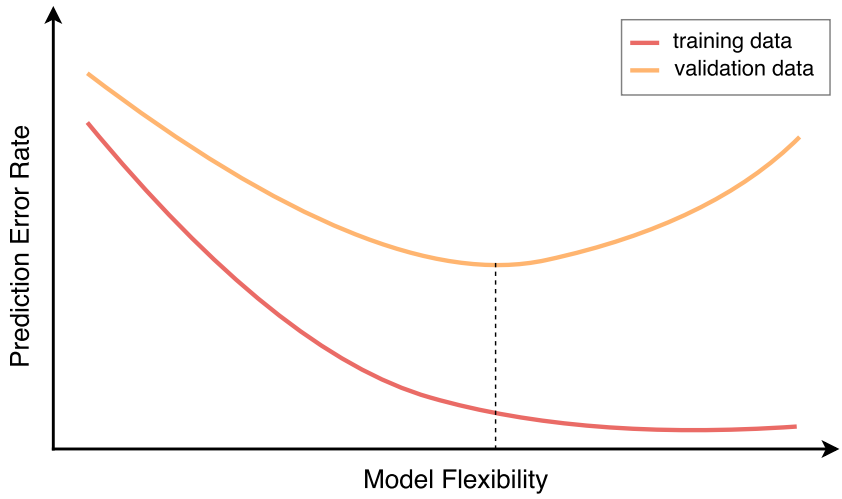


Figure 2.2: Typical curves of error rate in training and testing.

The just described validation set approach to model selection and performance evaluation, also known as the *hold-out* method, is an approximately unbiased estimator but may lead to highly variable test error rate since a particular data partitioning might favour one classifier over another. Resampling methods are techniques for variability reduction which the dataset is repeatedly partitioned into new subsets in order to obtain a more accurate and generalized performance estimation. Some approaches are described in Section 2.2.3.

Performance Metrics

To simplify visualization, the Bayesian approach to the classification problem considered in the beginning of this section will be restricted to a one-dimensional binary scenario, hence $p = 1$, $n_c = 2$ and $y \in \{0,1\}$. Therefore, y_i is boolean, that is, a data point $\mathbf{x}$ either belongs to a class, which we will call the *positive* ($\mathcal{P}$) class and for which $y_i = 1$, or it does not belong, which we will call *negative* ($\mathcal{N}$) class, for which $y_i = 0$. Figure 2.3 presents the joint probabilities $p(\mathbf{x}, C_k) = p(\mathbf{x}|C_k)p(C_k)$ as functions of a feature value $\mathbf{x}$ for classes C_1 and C_2, which are assumed to be Gaussian. The black vertical like is the decision boundary which determines if a randomly picked $x = x_i$ will be assigned either to C_1 or C_2. The choice for the threshold decision line is often driven by minimizing misclassification (when $\hat{y}_i \neq y_i$).

Often times, the variance between classes (inter-class variance) is smaller than the within class variance (intra-class variance), causing classes to overlap, which leads to misclassification. A common criteria to define model's quality and measure its performance is quantifying misclassification though different metrics.

There are four possible misclassification outcomes for a predicted value $\hat{y}$ for a binary classifier as depicted in Figure 2.3(a) namely: (i) the correct classification of an observation $\mathbf{x}_i \in \mathcal{P}$ is called true positive (TP); (ii) the correct classification of an observation $\mathbf{x}_i \in \mathcal{N}$ is called true negative (TN); (iii) the incorrect classification of an observation $\mathbf{x}_i \in \mathcal{N}$ is called false positive (FP); and (iv) the incorrect classification of an observation $\mathbf{x}_i \in \mathcal{P}$ is called false negative (FN).

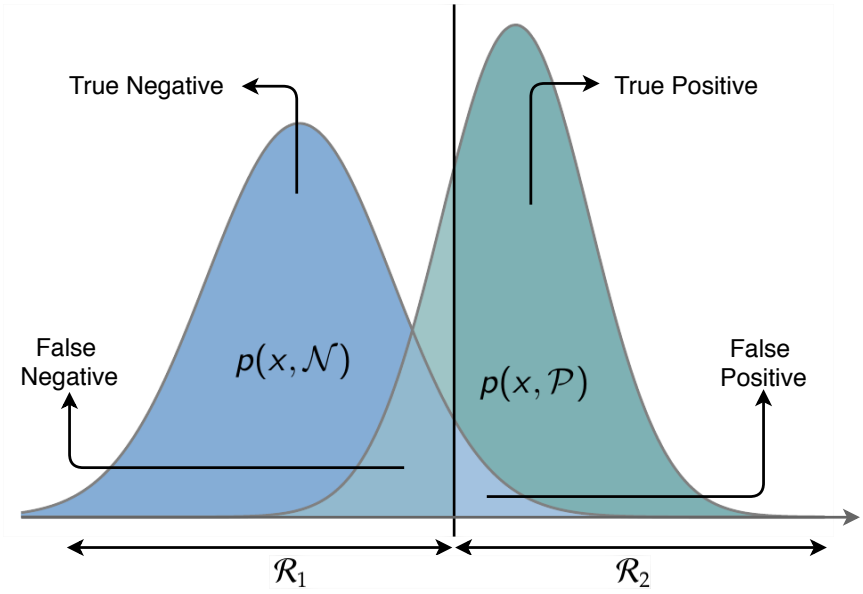


Figure 2.3: Representation of a binary class data's distribution (under Gaussian assumption); black vertical line represents classifier's decision boundary.

The number of each outcome occurrences can be organized into a *confusion matrix*, from which many common metrics are calculated. The true positive rate (TPR), also known as *hit rate*, *recall* and *sensitivity*, is defined as

$$TPR = \frac{\text{positives correctly classified}}{\text{total positives}} = \frac{\#TP}{\#TP + \#FN}, \tag{2.18}$$

as depicted in Figure 2.4(a) and expresses the classifiers ability to correctly detect a class. The true negative rate (TNR), likewise measures the classifiers ability to correctly reject a class, and is estimated by

$$TNR = \frac{\text{negatives correctly classified}}{\text{total negatives}} = \frac{\#TN}{\#TN + \#FP}, \tag{2.19}$$

and is also known as *specificity*. The complement of TNR is know as the false positive rate, and is given by

$$FPR = \frac{\text{negatives incorrectly classified}}{\text{total negatives}} = \frac{\#FP}{\#TN + \#FP}, \tag{2.20}$$

as shown in Figure 2.4(c). The positive predictive rate (PPR), or *precision*, measures the proportion, within all positive results, the correctly predicted outputs , defined as

$$PPR = \frac{\text{negatives incorrectly classified}}{\text{total negatives}} = \frac{\#FN}{\#TN + \#FP}, \tag{2.21}$$

as depicted in Figure 2.3(b) and 2.4(d). In other words, precision is the probability of a randomly selected positively labelled test observation is incorrectly classified as negative. It is a measure of exactness, or quality of the prediction.

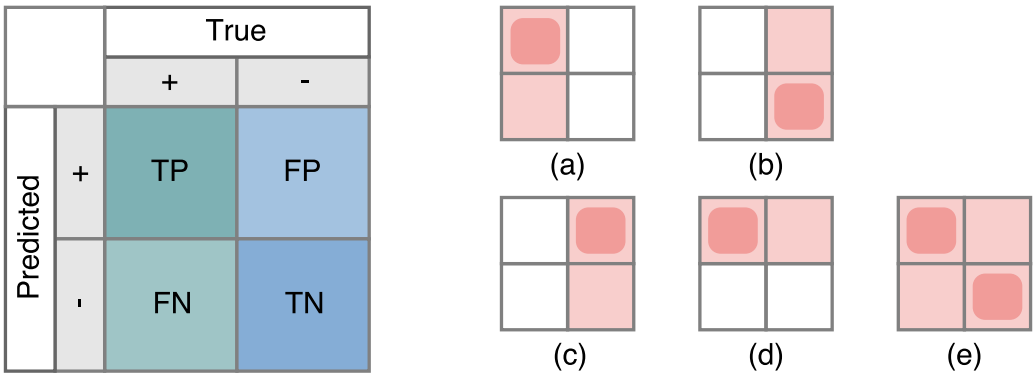


Figure 2.4: Confusion matrix and some common performance metrics: (a) true positive rate (TPR), (b) true negative rate (TNR), (c) false positive rate (FPR) and (d) positive prediction rate (PPR) (e) accuracy.

2.2.3 Cross-Validation

One of the most commonly used resampling methods is cross-validation (CV) and it is usually applied to model selection, as previously described, or model assessment, by determining performance statistics based on virtually increasing the test dataset and better evaluating model's performance in cases in which available data is limited. The most commonly used CV methods are described below.

Leave-One-Out Cross-Validation

The leave-one-out cross-validation (LOOCV) method splits the available training set, of size n, into a validation subset, composed of a single observation, and a training subset of size $n-1$, containing all the remaining observations. Model fitting is repeated n times, for each of the different data partitioning, so that at each observation is used once at validation step. The final result is given by averaging all n obtained results.

In contrast with the hold-out method, LOOCV always renders the same result, since all the available data is used for both validation and model fitting instead of being randomly split, which causes variations. However, LOOCV can be computationally expensive, and therefore is not ideal for large datasets, nor for models in which the fitting process is slow.

k-Fold Cross-Validation

Desiring to reduce the number of training steps and thus, the computational cost, an alternative to LOOCV is the k-fold cross-validation method for which the training dataset is randomly split into k groups instead of n groups.

In terms of test error estimate, LOOCV yields smaller bias than k-fold, since the number of observations for training in the former method, $n-1$, is larger than in the latter, $(k-1)(n/k)$. Nevertheless, in what variance is concerned, k-fold presents smaller variance than LOOCV, for in the former method training sets do not overlap as much as in the latter.

Repeated Random Sub-sampling Validation

This resampling method, also known as Monte Carlo cross-validation (MCCV) or *shuffle and split*, randomly splits the data in training and validation sets, as in the hold-out method, and a model is fit to the training data and performance estimate is computed with validation data. Such procedure is carried out repeatedly in a Monte Carlo loop, so that predictive accuracy is assessed by averaging the results over the splits.

Conceptually, this method resembles the k-fold CV, differing on the fact that the size of each fold does not depend on the number of iterations, which poses as an advantage. Yet, some samples might never be picked for the validation subset and some might be

picked more than once. In addition, a variation in results appears if different seeds are used for the random generator, that is, each time the Monte Carlo loop is run, a different set of splits is generated.

It is common practice to rearrange data and divide it in homogeneous subgroups, ensuring class representativity in each fold. In statistics, this method is called stratification sampling and is often used in Monte Carlo methods providing variance reduction, mainly for unbalanced datasets.

2.2.4 Support Vector Machines

Support Vector Machines (SVM) is a popular approach to classification, for they tend to perform well in a variety of problems, and is considered a powerful tool in machine learning. Although conceptually SVM models are supervised binary linear classifiers, this learning method can also be used for clustering, pattern recognition and multiclass classification problems.

The SVM learning problem is based on finding a plane in the feature space that separates two classes, with largest margin. This $(p-1)$-dimensional hyperplane and the points closest to it must satisfy $|\mathbf{w} \cdot \mathbf{x}_i + b| = 1$, where $\mathbf{w} \in \mathbb{R}^p$ is the normal vector to the hyperplane and b is a bias term. The margin is defined as the minimum, hence perpendicular, distance from each observations to the hyperplane and equals $2/\|\mathbf{w}\|_2$, as shown in Figure 2.5. Training a SVM involves solving the optimization problem

$$\begin{aligned} \underset{\mathbf{w},b,\xi}{\arg\min}\ J(\mathbf{w},\xi) \quad &= \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_{k=1}^{N}\xi_k \\ \text{subject to} \quad &\quad y_i\left[\mathbf{w}\cdot\mathbf{x}_i + b\right] \geq 1-\xi_i, \quad k = 1,\ldots,n,\ \xi_i \geq 0 \end{aligned}$$

in order to maximize the margin, which is equivalent to minimize $\|\mathbf{w}\|_2$, and simultaneously obey the constraints so as to prevent observations from falling on wrong side of the margin. This optimization problem resembles the regularized least-squares, where the latter minimizes the squared prediction error and the former, the distance between support vectors and the separating hyperplane. The terms ξ_i are slack variables representing the classifier's tolerance to misclassification, for the case in which classes overlap and therefore these classes are non separable. The tuning parameter C determines the trade-off between margin-size and number of observations on the incorrect side of the margin defined by the hyperplane. Smaller values of C lead to narrower margin and hence less margin violations are tolerated. Conversely, larger values of C implies wider margin, allowing more violations.

The training observations that affect classifier are those nearest to hyperplane, and are called *support vectors*. Since the hyperplane is determined based on a small subset of training data points, such that SVM classifiers are robust to observations far from the hyperplane. When considering the bias-variance trade-off discussed previously, larger

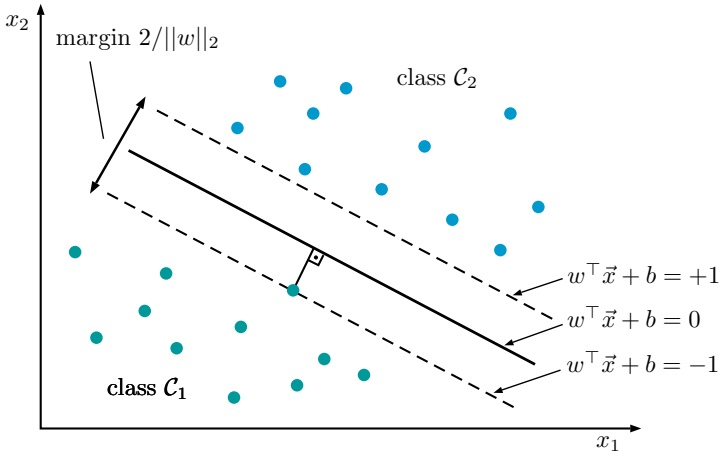


Figure 2.5: Linear binary classification for a separable case in two-dimension feature space. Unique hyperplane with maximum margin, the distance between the dashed lines.

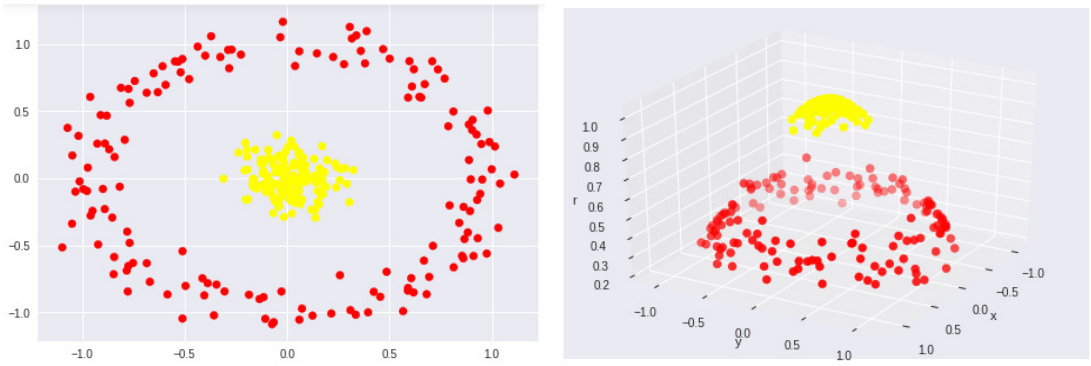
values of C result in more support vectors, which corresponds to a more biased model but with lower variance and therefore more robust to new observations.

The estimated function $\hat{f}(\cdot)$ for a linear SVM model can be represented as

$$\hat{f}(\mathbf{x}) = \text{sgn}\left[b + \sum_{i=1}^{n} \alpha_i \langle \mathbf{x}, \mathbf{x}_i \rangle\right], \tag{2.22}$$

where the coefficients α measure the degree of closeness of a new test point $\mathbf{x}$ to each point $\mathbf{x}_i$ and are nonzero only when $\mathbf{x}_i$ is a support vector. Thus only inner products of new data and support vectors affects the output. It is important to note that SVM are non-probabilistic, that is, the scores computed in (2.22) are measurements of distance between points and the hyperplane, not posterior probability estimates. Methods such as Platt's Scaling (Platt et al., 1999) can be applied in order to produce comparable scores, process often referred as *calibration* and is sometimes needed in multiclass problems.

For problems that require nonlinear decision boundaries, the so-called "kernel trick" (Boser et al., 1992) technique can be used. It is based on expanding the feature space to a higher dimensional space by using a non-linear mapping $\varphi(\mathbf{x})$ from the original feature space to a new one. For this formulation, the inner product in (2.22) is replaced by the kernel function $\mathcal{K}(\mathbf{x},\mathbf{x}_i) = \langle \varphi(\mathbf{x}), \varphi(\mathbf{x}_i) \rangle$. An example of a commonly used kernel is the Gaussian radial basis function, give by $\mathcal{K}(\mathbf{x},\mathbf{x}_i) = \exp\left(-\gamma \|\mathbf{x} - \mathbf{x}_i\|^2\right)$. Figure 2.6 shows an example of a problem that becomes linearly separable under nonlinear transformation with radial basis function.



(a) (b)

Figure 2.6: Example of nonlinear classification problem. Mapping to a higher dimension with a radial transformation allows for linear separation.

Approaches to the Multiclass Problem

The hyperplane concept that underlines the SVMs separating principle does not support multiclass classification problems, that is, when $n_c > 2$. The most popular approaches for such cases are the *one-against-one* and *one-against-all* classification paradigms. The preference for each approach is application driven, hence advantages and suitability of each strategy depends on the problem setting.

The one-against-one (OAO) is a "divide and conquer" approach by which the multiclass problem is decomposed into several binary sub-problems. For such, a classifier is trained for each pair of classes, that is, $\binom{n_c}{2}$ models are trained. A test observation is classified by each one of the $\binom{n_c}{2}$ pairs of classifiers. The final decision is given by a polling system for all pairwise classifiers, where the classification result is determined by the most voted class.

The one-against-all (OAA) approach, however, fits a model which must separate a given class from all the rest that does not belong to said class. Therefore, only n_c classifiers are trained, one for each class. A new pattern is then assigned to the class which produces largest score, usually computed as $b + \sum_{i=1}^{n} \alpha_i \langle \mathbf{x}, \mathbf{x}_i \rangle$. However, these values represent data points' distance to the hyperplane and are unbounded, therefore they are not directly comparable for the case of multiple classifiers. For such reason, calibration methods are applied for post posterior probabilities comparison.

The problem considered in this thesis is assumed to be a single-label problem, that is, no more than one label is assigned to each class.

Chapter 3

Human Activity Recognition from Accelerometer Data

As discussed in Chapter 1, accelerometers have proven to be the sensors that provide most useful information in the strategy of wearable sensing. The quality of the information does not depend solely on the sensor but is also related to its placement and the nature of the movements of interest, that is, if one wishes to detect changes in posture or hand gestures.

In this chapter, three different approaches to Human Activity Recognition (HAR) from accelerometer data are considered. Sections 3.2 and 3.3 review recent proposed works in literature, which use probabilistic models and autoregressive models, respectively. At last, we propose a new method in Section 3.4, as a variation of the approach presented in Section 3.3.

3.1 Accelerometer Data

The acceleration signal is due to the wearer's body movement and due to gravity, such that *measured* acceleration can be represented as

$$\vec{a}_m = \vec{a}_t - \vec{g} + \vec{v}, \tag{3.1}$$

where $\vec{a}_m = [a_x\, a_y\, a_z]^\top$; $\vec{a}_t \in \mathbb{R}^3$ is the translational acceleration; $\vec{g} \in \mathbb{R}^3$ is the gravity acceleration vector ($\|g\| = g_0 = 9.806\,\mathrm{m/s^2}$); and $\vec{v} \in \mathbb{R}^3$ represents additive zero mean measurement noise. The measured acceleration and gravity acceleration conventionally have opposite signs because of the sensor measurement principle that usually depends on inertial movement of a seismic mass inside the accelerometer Doebelin and Manik (2007).

In this work, such measurements are considered to be provided by wrist worn devices, as exemplified in 3.1.

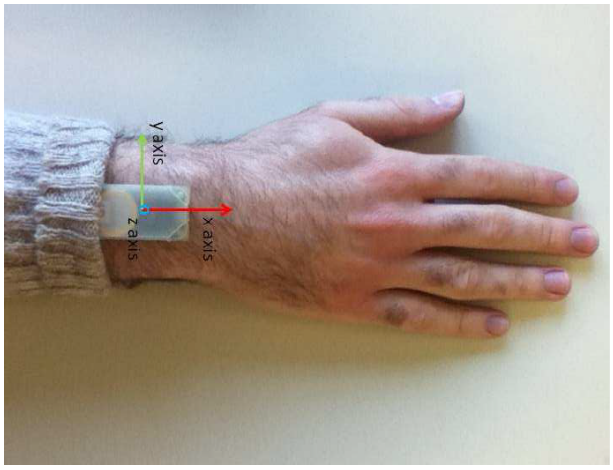


Figure 3.1: Wrist worn ad-hoc acceleration measuring device.

3.2 Recognition Using Gaussian Mixture Models

By considering accelerometer data provided by a wrist-placed ad-hoc device, Bruno et al. (2013) address the HAR problem by creating probabilistic models of the so called "motion primitives", which they define as being motions that uniquely identify an activity belonging to the larger group of Activities of Daily Living (ADL). The run-time data can then easily be compared to the built signal models, for they are in the same space of the raw data.

The proposed system's architecture comprises two modules: the model builder and the classifier, as seen in Figure 3.2. The former works *off-line* using Gaussian Mixture Models (GMM) and Gaussian Mixture Regression (GMR) to estimate models for each activity from a set of recordings, as done in (Mathie, 2003; Allen et al., 2006). The latter is intended to work *on-line* by classifying the run-time acceleration via comparison with the available models provided by the off-line module.

The model builder module starts by median filtering the raw tri-axial acceleration data, for noise reduction, followed up by feature extraction. This step is based on separating *gravity* and *body* translational components from the filtered data. The former is taken directly by applying a low-pass filter to the acceleration signals and the latter, by subtracting this low frequency component from the original signals. This way, features keep correlation information which has proven to result in better classification performance than uncorrelated features (Bruno et al., 2012). A set of these two features is shown in Figure 3.3(a), extracted from 20 trials of activity *drink*.

Authors' main purpose in adopting GMM trained using GMR for model building is to create a compact and general version of each feature for each activity. GMMs are a parametric representation based on the weighted sum of multivariate Gaussian distributions. Hence, a model associated to a motion primitive can be defined as

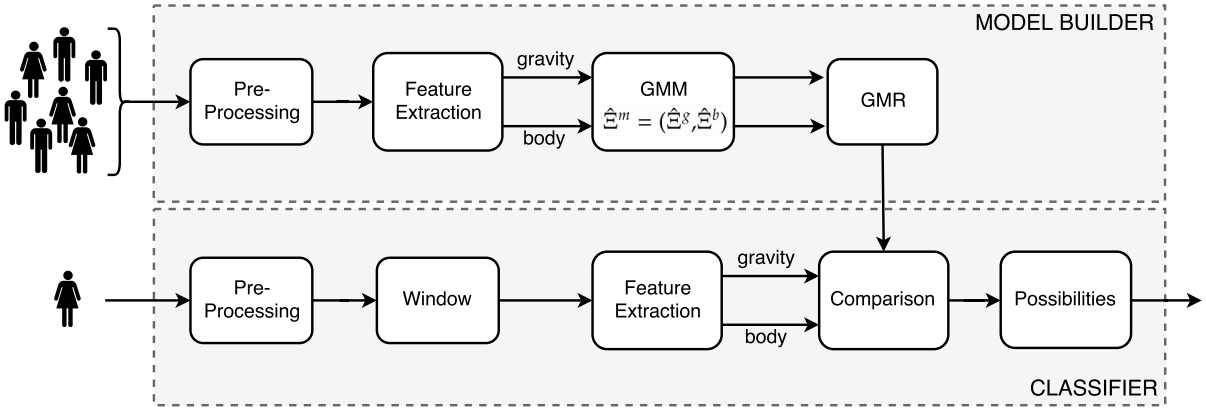


Figure 3.2: Proposed system architecture by Bruno et al. (2013), comprising the *off-line* model builder module and the *on-line* classifier module.

$\hat{\Xi}^m = (\hat{\Xi}^g, \hat{\Xi}^b)$, where $\hat{\Xi}^f = (\mu^f, \Sigma^f)$ for which μ^f is the expected curve modelling a feature f and Σ^f is the respective covariance matrix. By the GMM approach, often used for soft-assignment, each data point in run-time space is regarded as having been generated by a different Gaussian distribution. In order to estimate $\hat{\Xi}^m$, the following modelling procedure takes place:

1. GMM initialization: initial values $(\mu_0, \Sigma_0)_k$ for each of k Gaussian distributions are defined by applying the k-means algorithm to the run-time data. An optimized selection of the total number of Gaussians distributions in a model is proposed by the authors in Bruno et al. (2012). The computed centroids, which are assigned to the expected value μ_0, are shown in light green points, as well as their respective dispersions, in Figure 3.3(c), as an example of model initialization for the feature of activity *drink*;

2. GMM training: the probability of each data point cluster membership is estimated via Expectation Maximization algorithm (Candy, 2016), through which the modelling parameter values $(\mu, \Sigma)_k$ of each Gaussian distribution are also inferred.

3. GMR retrieval of expected curves: a smooth and generalized version of data can be retrieved from the trained GM models by the GM regression algorithm. The expected curves and respective covariances for both features, gravity and body acceleration temporal profiles, of activity *drink*, hence $\hat{\Xi}^{\text{drink}}$, are shown in Figure 3.3(b).

A step of uttermost importance refers to data segmentation. The described model building procedure relies on data synchronization through which all training trials have equal length and the performed gestures of all trials start roughly at the same sample, as observed in Figure 3.3(a). The main reason this is done is that the retrieved

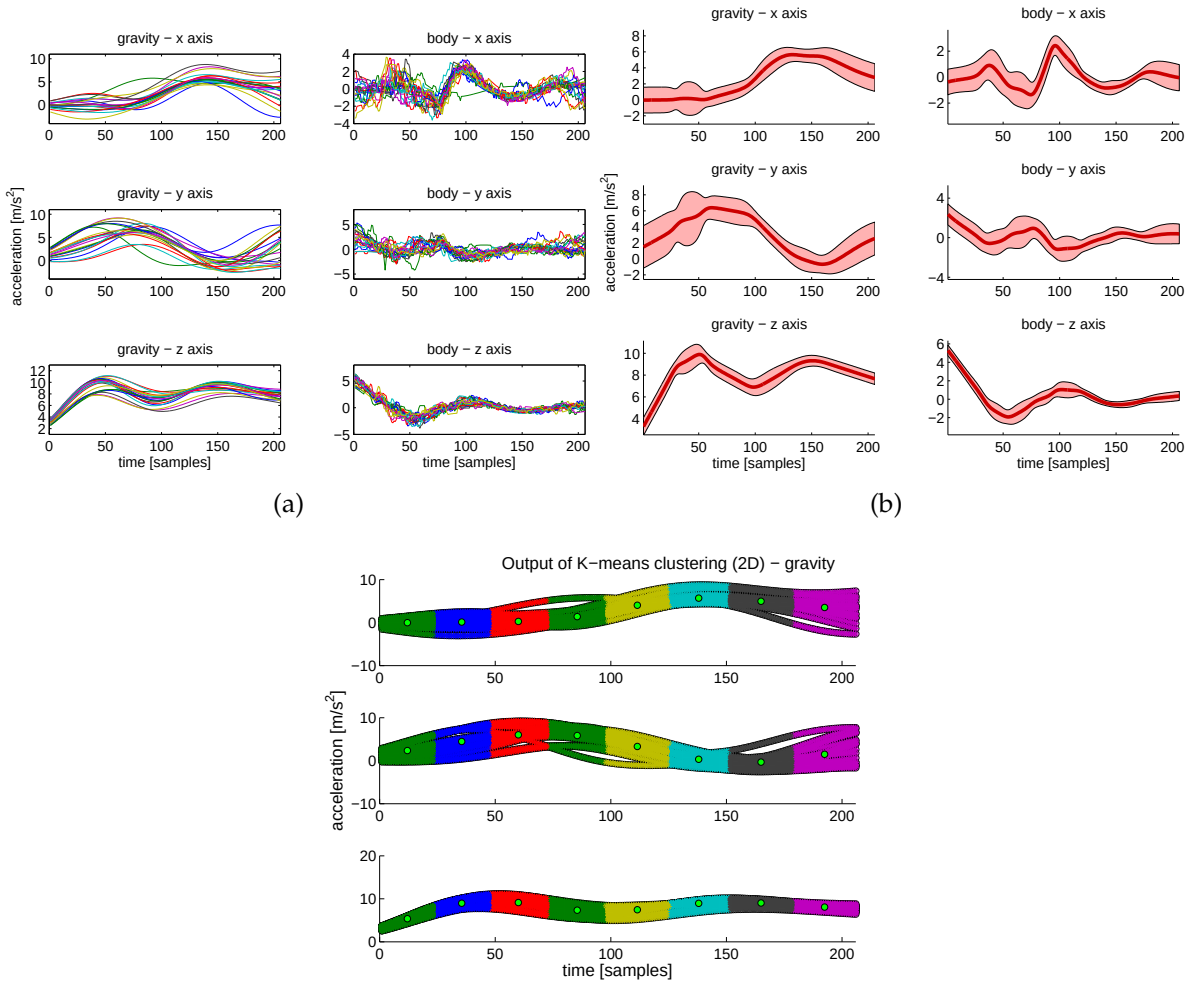


Figure 3.3: Outputs of model building procedure for 20 trials activity *drink*. (a) Feature extraction: acceleration components *gravity* and *body*; (b) Final model: expected curves for acceleration components, obtained by performing Gaussian Mixture Regression considering Gaussian Mixture Models; (c) Initial clusters (radial basis of each Gaussian) and respective centroids. As observed, the optimum number of Gaussians obtained for this case is 8.

models curves can have a general yet consistent form, capture the rise-fall pattern of each gesture and reduce the estimated covariance matrix associated to each curve, as seen in Figure 3.3(b).

In the classifier module, data filtering and features extraction are carried out just as in the model builder module so that new run-time data can be compared to the previously obtained probabilistic models. The goal in this module is to select amongst the obtained models the one closest to run-time acceleration data. For such, the following steps are taken:

1. an acceleration data trial is swept by a sliding window whose size is defined by the length of the longest estimated model;

2. the overall distance between the window data and each model, $d(\Xi_w,\hat{\Xi}^m)$, is computed as the average Mahalanobis distance of all feature component. A threshold mechanism is set up to discriminate between unknown and potentially known activities, in which the threshold τ_m is computed as the Mahalanobis distance between model mean components and the farthest curve of the model itself. The possibility of a window belonging to a motion primitive is defined as

$$p_{w\in\Xi} = 1 - d(\Xi_w,\hat{\Xi}^m)/\tau_m, \tag{3.2}$$

   that is, the complement of the ratio between overall distance and the threshold of same motion primitive. For when $p_{w\in\Xi}$ is negative, the possibility of the window belonging to the given model is defined as null;

3. the similarity between tested data and models is then ranked by the computed possibilities. The assignment of an activity to a given window is determined by the motion primitive that rendered the highest possibility value. Windows to which no activity is assigned are neglected;

4. the final off-line classification of a trial is given by the most frequently assigned motion primitive in the previous step. This was not explicity presented in Bruno et al. (2013), but it was discussed with main author, as registered in Annex A.

Authors concluded that automatic ADL recognition relying solely on two time-domain features is comparable to other systems depending on more features, for they belived that the reduced amount of attributes is compensated by the chosen approaches to modeling and comparison. Driven by the same goal of correctly identifing human activities from wrist worn acceleration data, the work in Bruno et al. (2013) was chosen as benchmark for the method proposed in the present thesis.

3.3 Recognition Using Autoregressive Models of Acceleration Data

An interesting approach to feature extraction from acceleration signals used in HAR is by approximating AR linear models, as presented in Section 2.1.2. The use of coefficients from AR models as features for HAR problems has been considered in previous works (He and Jin, 2008; Khan et al., 2008). As mentioned earlier, such models can be used as powerful one-step ahead predictors and their parameters can be considered as mixed time domain and frequency domain representations for the dynamic behaviour for a stochastic process, and therefore are appropriate to deal with the acceleration raw signals.

He and Jin (2008) proposed estimating AR models for each tri-axial accelerometer signal, placed in front trousers pockets. Even though a model structure selection based on Akaike's criterion (Aguirre, 2015) suggested best model order to be 3, the authors used 4^{th} order models to create feature vectors with dimension 12, so to compare results with traditional 12 dimensional time-domain features and 12 FFT features.

Authors in Khan et al. (2008) go further and propose augmenting the feature vector with the *signal magnitude area* (SMA) value, defined mathematically as

$$SMA = \frac{1}{N}\left(\sum_{i=1}^{N}|a_x(i)| + \sum_{i=1}^{N}|a_y(i)| + \sum_{i=1}^{N}|a_z(i)|\right), \tag{3.3}$$

in which N is the number of samples of the signal, and the *tilt angle* (TA) which is defined as the angle between the z-axis and the gravitational vector $\vec{g}$, mathematically expressed by

$$\Phi = \arcsin(\tilde{a}_z), \tag{3.4}$$

where $\tilde{a}_z$ is the normalized component of signal a_z. The idea is that SMA will enable distinguishing dynamic and static activities (e.g. standing and walk), as done in many other works (Karantonis et al., 2006; Krassnig et al., 2010), and TA allows distinguishing postures (e.g. standing and lying), as in Karantonis et al. (2006), by evaluating user's relative tilt of the body in space. Such features are often used in rule-based methods where their values are compared to predefined thresholds in order to infer performed activity. In the referenced work, Φ is a single scalar value, even though a_z is a time sequence. Therefore we have assumed that Φ is actually computed as

$$\Phi^* = \frac{1}{N}\sum_{i=1}^{N}\arcsin(\tilde{a}_z(i)), \tag{3.5}$$

because postural orientation can be defined by fixed angles ranges, as depicted in Figure 3.4, and so the exact value of Φ need not be calculated, but rather defined by established

thresholds (Karantonis et al., 2006). TA is related only to the gravitational component and relates to the low-frequency component "gravity", described in previous section, by representing body rotation.

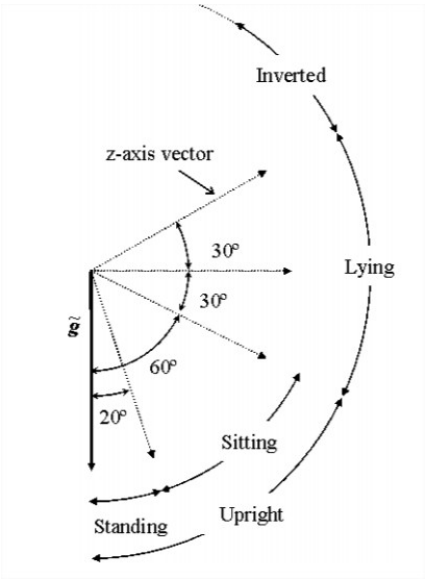


Figure 3.4: Method for determining the postural orientation from tilt angle (Karantonis et al., 2006).

The AR model's order was chosen by signal's autocorrelation analysis, leading a value of 3. As opposed to the approach presented in Section 3.2, cross correlation among all tri-axial acceleration signals was not taken into consideration. With respect to classification, authors in (Khan et al., 2008) used a multilayer perceptron (MLP) neural network to validate the good performance that could be achieved using their feature vector .

An extra step before classification was added in Khan et al. (2010c,a) in order to reduce large intra-class and low inter-class variance and therefore improve separability. High within-class variance can be related to lack of firm attachment of the wearable accelerometer and its different positioning, as in the case of embedded sensors in smartphones that might be in shirt pockets, trousers front or back pockets, facing upwards, facing downwards, etc. Thus, linear discriminant analysis (LDA) (Welling, 2005) was applied to the feature vector composed of AR coefficients and SMA so that recognition could be made independent of accelerometer position. However, kernel discriminant analysis (KDA) (Mika et al., 1999) proved to be more effective in reducing intra-class variance, as shown in Khan et al. (2010b). Another extra step in feature extraction was also applied in He (2010), where raw accelerometer signals were decomposed by computing Wavelet Transforms (Sifuzzaman et al., 2009) prior to AR model building. In a work by Khan and co-workers (Khan et al., 2013) a complete

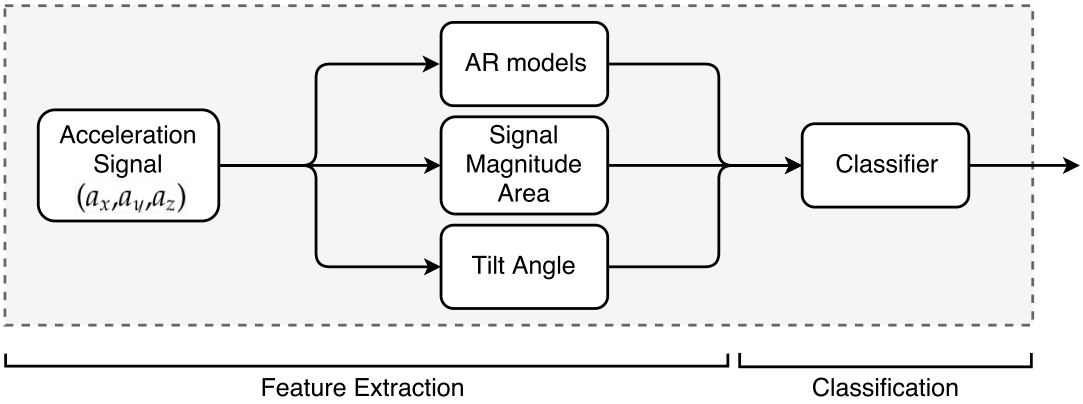


Figure 3.5: Proposed system architecture by Khan et al. (2008).

smartphone-based activity recognition system is presented, relying on AR models coefficients extraction from accelerometer signals followed by the use of KDA.

With respect to classifiers, authors in (Khan et al., 2008, 2010c,a; He, 2010) have used artificial neural networks (ANN), while He and Jin (2008); He (2010) have opted for SVM following a one-versus-one approach.

From all reviewed works using AR coefficients from tri-axial accelerometer data as features, the method proposed by Khan et al. (2008) was chosen to be considered as the reference to validate the method proposed in the next section, by means of performance. A schematic of the approach by Khan et al. (2008) to the HAR problem is presented in Figure 3.5.

3.4 Recognition Using Autoregressive Models of Attitude

In (Aguirre, 2017), which is a scientific contribution from the present work, it was also considered the use of AR linear autonomous models to capture, from available sensing data, the acceleration dynamics of each human activity. However, we propose estimating a multivariate model (VAR), as presented in (2.8), page 9 in Section 2.1.2, as opposed to separate univariate AR models for each signal, as considered in Khan et al. (2008, 2010c,a); He and Jin (2008); He (2010). The underlying assumption for following this multivariate approach is that the components of the measured acceleration of a human body limb during the execution of a movement are not independent from each other.

In addition, as mentioned earlier in this thesis (Section 1.4), wishing to investigate recognition of complex activities rich in arm movements, we propose feature enhancement by computing wrist's attitude angles from accelerometer data, instead of relying

only on raw data.

Authors in (Ramos-Garcia et al., 2015; Kratz et al., 2013; Dong et al., 2012) also consider features that depend on wrist attitude obtained from gyroscopes for wrist motion tracking in the gesture recognition problem. On devices without a gyroscope, rotation can be approximated using accelerometers alone by using the direction of gravity as a reference for tilt, and estimating the Euler angles of roll (ϕ) and pitch (θ) by trigonometric relations

$$\phi = \text{arctg}\left(\frac{a_y}{a_z}\right), \tag{3.6}$$

$$\theta = -\text{arcsin}\left(\frac{-a_x}{\sqrt{a_x^2 + a_y^2 + a_z^2}}\right). \tag{3.7}$$

In (3.6), arctangent with two arguments is used in order to avoid discontinuities due to division by zero.

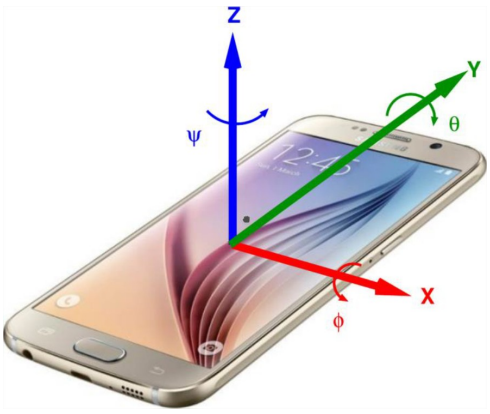


Figure 3.6: Euler angles representation of spacial orientation.

Figure 3.6 presents an example of inertial orientation of a smartphone, and the relationship between each Euler angle and the corresponding. Only two estimates are attainable from a tri-axial measurement since no information on yaw angle (ψ) can be computed from acceleration measurement resulting from projection of gravity.

It should be noted that equations (3.6) and (3.7) present an estimate of the rotational angles ϕ and θ by assuming that the contribution of the wrist translational acceleration is much smaller than that of the local gravity and that the level of noise is negligible, i.e. $\vec{a}_m \approx -\vec{g}$. It is also worth mentioning that approximations in (3.6) and (3.7) are not always reliable for when the device is perpendicular to gravity, no tilt nor rotation information can be inferred (Kratz et al., 2013), due to singularities in the rotational matrix.

The roll and pitch angles represent rotational change with respect to the gravitational vector $\vec{g}$, and therefore are comparable to the low-frequency component *gravity*, in Section 3.2, and to the tilt angle, in Section 3.3. Former works in the literature such as (Krassnig et al., 2010) also consider Euler angles but these angles are regarded as static features, similar to the TA in (Karantonis et al., 2006; Khan et al., 2008) where they are usually related to ambulatory activities and their role is mainly to distinguishes postures. On the other hand, by the computed attitude angles from accelerometer's data allow wrist motion tracking, for they are a more complete description of the wrist attitude.

Based on the premise that attitude signals carry useful information about wrist rotation, a new combined feature vector for the HAR problem can be proposed, by adding to the feature vector described in Section 3.3 coefficients of multivariate AR models (2.8) of the estimated attitude angles (3.6) and (3.7) such that

$$\begin{bmatrix}\phi(k)\\ \theta(k)\end{bmatrix} = A_1 \begin{bmatrix}\phi(k-1)\\ \theta(k-1)\end{bmatrix} + \cdots + A_n \begin{bmatrix}\phi(k-n)\\ \theta(k-n)\end{bmatrix} + \vec{e},$$

where $A_i = \begin{bmatrix} a^i_{11} & a^i_{12} \\ a^i_{21} & a^i_{22}\end{bmatrix}$ is the coefficient matrix, n is the model's order and $\vec{e}$ is the model error. Coefficients a^i_{12} and a^i_{21} quantify the effects of $\phi(k)$ on $\theta(k)$ and vice-versa, thus measuring the temporal correlation between signal axes.

Therefore, the proposed feature vector is comprised by coefficients of multivariate AR models from both tri-axial accelerometer signals and estimated attitude angles, and SMA value, as depicted in Figure 3.7.

For AR model estimation, the following assumptions were made: (i) system non-linearity can be neglected for the purposes of the present work; (ii) sampling frequency is constant; (iii) signals are second-order stationary processes, i.e., the time series are characterized by constant mean and variance (Madsen, 2007).

3.5 Related Works

The works mentioned in Sections 3.2 and 3.3 follow the conventional approach to HAR, which presentes handcrafted features to classifiers. In Section 3.4, we describe a new set of handcrafted features that is based on the features proposed in (Khan et al., 2008), since they report positive results. Tehrefore, the method we propose derives of this method by:

- maintaining the autoregressive coefficients, which expresses the activities dynamics, and the SMA value, which carries information of signals energy;
- substituting the single TA value by the autorregressive coefficients of the estimated attitude. In doing such, we intend increasing information related to

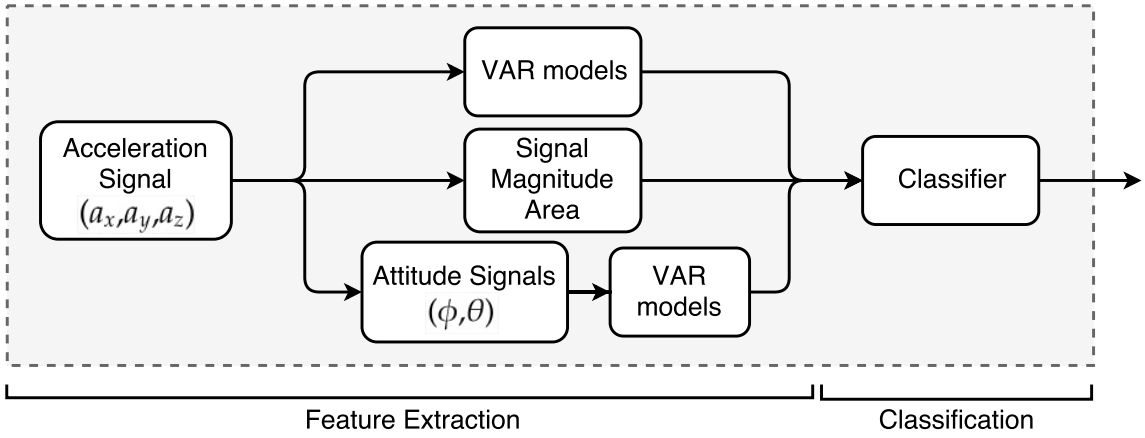


Figure 3.7: Proposed system architecture.

body's inclination and thus increase recognition of activities which are better characterized by rotation than translational displacement;

- adding between correlation information by using multivariate autoregressive models in place of the univariate approach used in (Khan et al., 2008).

Despite this conventional approach to HAR, recently an increasing number of works have employed deep architecture Convolution Neural Networks (CNN), which learns and extracts the best features for the specific task and avoids the need of domain specific knowledge about acceleration data. This approach to HAR presents state-of-the-art results in (Chen and Xue, 2015), outperforming common time domain features (mean, variance, short-time energy and correlation coefficients) and frequency domain features (Fast Fourier Transform and Discrete Cosine Transform) using SVM and Deep Belief Network (DBN). However, authors in Jordao et al. (2018) point out that results in (Chen and Xue, 2015) is limited to walking-based activities and propose enhancing CNN to improve data representation and increase activity discrimination. They propose improvements to the state-of-the-art method.

The use of such state-of-the-art classification method as a benchmark for the proposed feature extraction method would be, to say the leat, intuitive. However, due to the CNN's conceptual bias in image processing, the method proposed in (Chen and Xue, 2015) fixes the size of the windowed accelerometer data which is fed to the CNN. This fixed structure makes the process of aggregating new features impracticable. Besides, as mentioned in Chapter 1, the use of a powerful classification method could disguise the true ability of the proposed feature extracted method of increasing class separability and therefore defeating the purpose of quantifying the potential of new set of features. For such reasons, we opted for a less sophisticated classification system, as described on the following Section.

Chapter 4

Results for the WHARF Database

In the present chapter, a detailed description of the implementation and experimental results for the three HAR methods described in Sections 3.2, 3.3 and 3.4 are presented. The dataset used is first described followed up by the pipeline architecture used to verify the proposed method for feature extraction in Section 3.4. The same pipeline is also used for the reviewed method presented in Section 3.3. For validation purposes, results reviewed in Sections 3.2 and 3.3 are reproduced and reported, in order to enable performance comparisons with the proposed method presented in Section 3.4 and published in (Aguirre, 2017).

4.1 Dataset Description

The dataset used in this work was first presented in (Bruno et al., 2013), and subsequently made available by the authors. The *Wearable Human Activity Recognition Folder* (WHARF) dataset is a public collection of accelerometer recordings of different activities, segmented and labelled accordingly. A thorough description of the dataset is presented in (Bruno et al., 2014).

In this dataset tri-axial accelerometer measurements ranges from -1.5g to +1.5g, with 6 bits of digital resolution per axis and sampling rate of 32 Hz. The accelerometers were embedded in an ad-hoc sensing device attached to the right wrist of each one of the 17 volunteers (11 male, with age ranging from 19 to 81 years; and 6 female, with ages between 56 and 85 years), with x-axis pointing towards the hand, y-axis towards the left, and z-axis perpendicular to the hand's palm. Data transmission to the PC was wired, via a USB cable.

The human activities available in the WHARF's dataset are listed in Table 4.1 as well as their respective number of trials, which is not the same for each group, rendering the dataset unbalanced. The selected activities range from ambulation activities to those that depend on arm movements, which is consistent with the present work's main goal of verifying the ability of the described method in Section 3.4 to correctly identify daily human activities both rich in arm movement or not. For such purpose, a subset of the activities in this dataset are considered, as listed in Table 4.2. The choice was made over a pool of available synchronized groups, since this step is fundamental for the method

described in Section 3.2, which is reproduced here for comparison reasons. Each data sequence vary from 140 up to 1000 samples, depending on the activity performed.

Table 4.1: Available human activities in WHARFs's dataset

Activity	Number of trials	Activity	Number of trials
Brush Teeth	12	Lie down	28
Climb Stairs	112	Pour water	100
Comb hair	31	Sit down	109
Descend Stairs	51	Stand up	112
Drink glass	115	Walk	100
Get up from bed	101		

4.2 Results Using GMM

The procedure briefly described in Section 3.2 was reproduced in order to generate results as presented in (Bruno et al., 2013).

For the GMM with GMR model fitting, 20 synchronized and truncated trials of each activity were used. This was done manually by authors, carrying out steps described in the private communication registered in Annex A. The activities available with synchronized data sequences are listed in Table 4.2.

The data files used for model fitting were excluded from the main data pool, to prevent training and testing subsets from overlapping each other. Likewise, those trials whose total number of samples was smaller than the size of the sliding window in classification were also excluded from the testing subset. The number of trials considered in this step are shown in Table 4.2.

Considering the implementation made available by (Bruno et al., 2013), Table 4.3 shows the results obtained for true positive rate (TPR) and true negative rate (TNR). Since these results were reproduced, in principle, using same code as the authors, it was not expected that the TPR values would be 10-30% smaller than those presented in their article. No evident reason for such low results was found. A possibility would be the scaling factor used for the calculus of the threshold τ_m which neither its value nor how it is chosen is described in the article. Results shown in Table 4.3 were computed using the value available in code. However, manually changing its value did not result in significant change in obtained TPR values. Low results for activities *sit down* and *stand up*, both in this thesis and in (Bruno et al., 2013) can be attributed to fact that such

Table 4.2: Selected subset of activities, based on available synchronized data sequences. Number of trials with length larger than sliding window.

Activity	Number of trials	
	training	testing
Climb Stairs	20	63
Drink glass	20	52
Get up from bed	20	70
Pour water	20	54
Sit down	20	15
Stand up	20	13
Walk	20	89

transitional activities have a much shorter time sequence than the chosen window and therefore, very few instances of the sliding window is compared to the activity's model.

Figure 4.1 presents the final confusion matrix. Almost all trials of activities *climb stairs* and *sit down* were confused with *walk*. Likewise, *climb stairs*, stand up from chair and *walk* were greatly confused with *get up* from bed. Some confusion is also observed between *drink* and *pour water*. It must be noted that 9 out of the 356 trials listed in Table 4.2 were not classified for no similarity to the activities GMM models was found, that is, the computed possibilities with (3.2) were less than zero.

Table 4.3: Performance metrics for GMM modeling method proposed by Bruno et al. (2013)

(%)	Climb stairs	Drink	Get up	Pour water	Sit down	Stand up	Walk
TPR	3.17	78.85	70.00	87.04	0	0	57.30
TNR	97.95	96.71	78.67	89.74	100.00	100.00	81.65
threshold	16.85	133.62	97.65	108.14	35.58	63.87	46.11

Predicted \ True	1	2	3	4	5	6	7
1	2	0	0	0	0	0	6
2	0	41	3	(7)	0	0	0
3	(19)	0	49	0	0	(12)	(30)
4	2	(10)	14	47	2	1	2
5	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0
7	(39)	0	0	0	(10)	0	51

Figure 4.1: Confusion matrix for classification with GMM models. Unclassified trials were not counted. Activities: (1) climb stairs, (2) drink, (3) get up from bed, (4) pour water, (5) sit down, (6) stand up from chair, and (7) walk. Observation: the 9 trials not classified are not considered here.

4.3 Pipeline Description

In order to investigate feature extraction methods presented in Section 3.3 and the proposed method presented in Section 3.4, we propose the processing pipeline described below.

4.3.1 Feature Extraction

The first step in the pipeline is feature extraction as depicted in Figures 3.5 (page 28) and 3.7 (page 31). It is important to note that no pre-processing takes place before feature extraction, that is, the data sequences are not truncated nor filtered.

The feature extraction step performs different mathematical transformations to each acceleration sequence in order to obtain 5 different features, namely

1. signal magnitude area (SMA), by using equation (3.3) to acceleration data;

2. tilt angle (TA), calculated by using equation (3.5) to acceleration data;

3. univariate autoregressive model of the type (2.7) for tri-axial acceleration data;

4. multivariate autoregressive model of the type (2.8) for tri-axial acceleration data;

5. multivariate autoregressive model of the type (2.8) for attitude angles $\phi(k)$ and $\theta(k)$, estimated from acceleration data by using equations (3.6) and (3.7).

The autoregressive models, univariate or multivariate, were estimated by the least-squares (LS) approach presented by equation 2.10, even though authors in (Khan et al., 2008) use Burg's method. Since Burg's Method and Yule-Walker method return similar results for large frame sizes (mathworks), we apply LS to both cases.

As described in Section 3.3, the feature vector is composed by features 1, 2 and 3 from the above list. And the feature vector used in the proposed method presented in Section 3.4 is comprised by features 1, 4 and 5. An extra feature vector, composed by features 1, 3 and 5 is considered in order to attempt to distinguish between the benefits due to estimated wrist orientation and due to the use of a VAR model for accelerometer data instead of an univariate model. The proposed classification procedure applied to each approach to feature extraction is the same and is described as follows.

The first step before classification is data scaling. Only the unbounded SMA and TA were normalized by shifting and scaling according to $\tilde{x} = (x - \mu)/\sigma$, for which $\mu = 0.5[\max(x) + \min(x)]$ and $\sigma = 0.5[\max(x) + \min(x)]$, with $\tilde{x}$ a normalized value, and x the corresponding original value.

4.3.2 Classifier

In order to verify the capability of the extracted features in representing each human activity considered, linear SVMs were employed, since they seem to be powerful and robust algorithms, with wide margin, and yet they could be considered "simple" classifiers, in the sense that not many parameters need to be tuned prior to training. Overfitting in both training and model selection becomes more severe as the number of data samples becomes smaller. For such reason, simpler models with less hyper-parameters to tune or even eliminating model selection altogether is recommended (Cawley and Talbot, 2010).

Yet, it cannot be ignored that the kernel decision and hyper-parameter tuning moves the overfitting problem from parameter optimization to model selection. Since we chose using a linear kernel for its simplicity, the overfit issue is related mostly to hyper-parameter tuning, which is treated in the proposed experimental protocol as described in the following subsection.

One-against-one (OAO) strategy proved to be a better option for our problem setting mainly due to the presence of unbalanced classes, for such classes only aggravates the tendency of classifiers in the OAA approach of seeing many more negative samples then positive. Although such nuisance can be softened by sub-sampling the over-represented class or over-sampling the under-represented class, both solutions present down-sides. Thus, OAO is rendered more practical for the present problem. Besides, the chosen strategy does not require extra parameter tuning, as would if we used the OAA approach with Platt's calibration method (Platt et al., 1999). Added to that, the

OAO process has higher computational cost as the number of classes becomes greater. Yet, since the models are simpler to train for they select only a pair of classes at a time, this approach to the multiclass problem tends to be more practical.

4.3.3 Experimental Protocol

To define the experimental protocol, we consider the cross-validation approach most appropriate to our problem that could reliably produce generalization performance estimates. From the options mentioned in Section 2.2.3, the LOOCV method presents itself as a more appropriate choice than the *k*-fold approach, for the number of iterations in a multifold cross-validation method is tied to the size of each fold and since the amount of data samples in the chosen dataset is limited and insufficient to be split into reasonable sized folds.

However, the LOOCV method does not seem to be a fair approach for a multi-class problem. Besides, as discussed by Xu et al. (2004), LOOCV often causes overfitting and typically underestimates the prediction error. MCCV, although not so commonly used in machine learning, has proven to be consistent and more often selects the best model. The probability of selecting the better prediction model increases because MCCV leaves out a larger amount of samples for validation at a time than LOOCV. In addition, there is no restriction for the number of iterations in the Monte Carlo approach. Figure 4.2 summarizes such aspects.

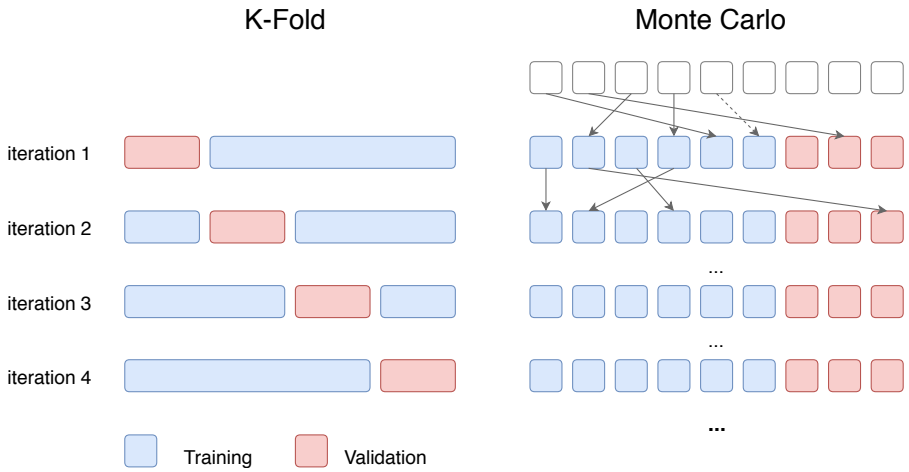


Figure 4.2: The Monte Carlo cross-validation (MCCV), in which data is repeatedly shuffled and proportionately split, allows a large number of iterations than the more common K-fold cross-validation, in which the number of iterations is tied to the size of each validation set.

The arbitrary splitting of the data introduces randomness. Consequently, the empirically obtained optimum hyper-parameter is not necessarily the true optimum. Biased

experiment protocol tends to cause overfitting in model selection and creates a subsequent optimistic bias in performance evaluation. This effect, however, is undesirable mainly for the case in which different learning algorithms performance are compared, and is not meaningful in the case of isolated algorithms performance (Cawley and Talbot, 2010).

For such reasons, in order to produce a fair comparison between the studied algorithms, we have implemented nested (or internal) cross-validation for an unbiased performance protocol. In this procedure, the model selection process is performed within each partitioning of the resampling procedure and therefore provides a more realistic estimate of the performance of the final selected and trained model (Cawley and Talbot, 2010). In contrast, in an external protocol, model selection is performed separately, as an initial step, using all training data and after the entire dataset is repeatedly re-partitioned for performance evaluation. As discussed by Cawley and Talbot (2010), the re-use of test data in an external protocol causes a consistently optimistic bias, even when number of hyper-parameters is small.

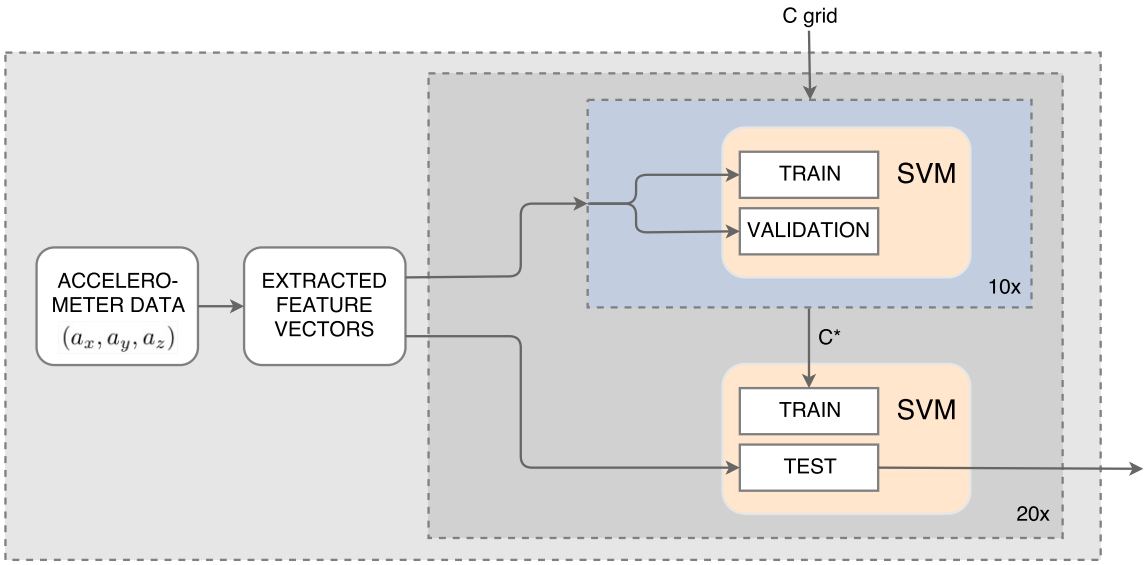


Figure 4.3: Experimental protocol schematic. Input data is randomly shuffled and split, 75% for training, which is used for the nested cross-validation, and 25% for testing. The internal procedure is repeated 10 times, as well as the whole shuffled and split procedure.

Finally, the stratified MCCV approach was chosen as the generalization performance estimator for both model selection and model assessment. Therefore, the experimental protocol takes place as follows:

1. input feature vectors are randomly reshuffled at each iteration and split into two sets: 75% for training, and the remaining 25% for testing;

2. the training subset is fed to the nested cross-validation algorithm, in which the optimum value for the hyper-parameter C is selected, as it will be explained later;

3. following up, all training data and selected value C^* are used for training the final classifier, which is then tested with the testing data.

The whole procedure is then carried out repeatedly, as depicted in Figure 4.3. Multiple randomised partitionings with model selection performed for each partition is time-consuming. So, we opted for 20 Monte Carlo iterations, in order to stablish a compromise between processor time and good performance evaluation.

In the model selection loop, we perform *grid-search* to carry out parameter tuning. For each value of C in a predefined grid, the received data is once again randomly reshuffled and split into two sets: 75% for training and 25% for validation, which is done repeatedly in a Monte Carlo loop. Thus, for each value in the grid, it is possible to compute an average error rate, as exemplified in Figure 4.4. We automated the hyper-parameter selection by detecting the curve's elbow computing the shortest distance between each point and a fitted line passing through the curve's extremes. Thus, the farthest point is most likely the curve's elbow and can be deemed as the optimum hyper-parameter for the considered data partition. This idea follows the method proposed by Satopaa et al. (2011).

This proposed approach to automatically choose the value for C is simple yet allows for a fair empirical estimate of the hyper-parameter value. In principle, this scheme is still effective when the mean error rate curve presents local minima. However, in order to be effective, the error rate curve must present a ridge shape. The closer either of the extremes of the estimated curve are from the searched elbow, the more the curve takes a hyperbolic shape and thus shifts the point with largest distance to the line. Hsu et al. (2016) suggests defining a coarse grid by an exponential growing sequence (for example, $C = 2^{-5}, 2^{-3}, \ldots, 2^{15}$) and then refining the grid around the found minimum. The searched grid was defined experimentally, prior to the classification procedure depicted in Figure 4.3.

4.4 Results Using the Autoregressive Models of Acceleration Data

As previously described in Sections 3.3 and 4.3.1, we consider human activity recognition by building a feature vector composed of SMA, TA and AR coefficients of three univariate models, one for each acceleration signal, as exemplified by a trial of activity *drink* in Figure 4.5.

The choice of activities to be classified was based on the available synchronized options in order to compare performance of this method with the GMM approach, as

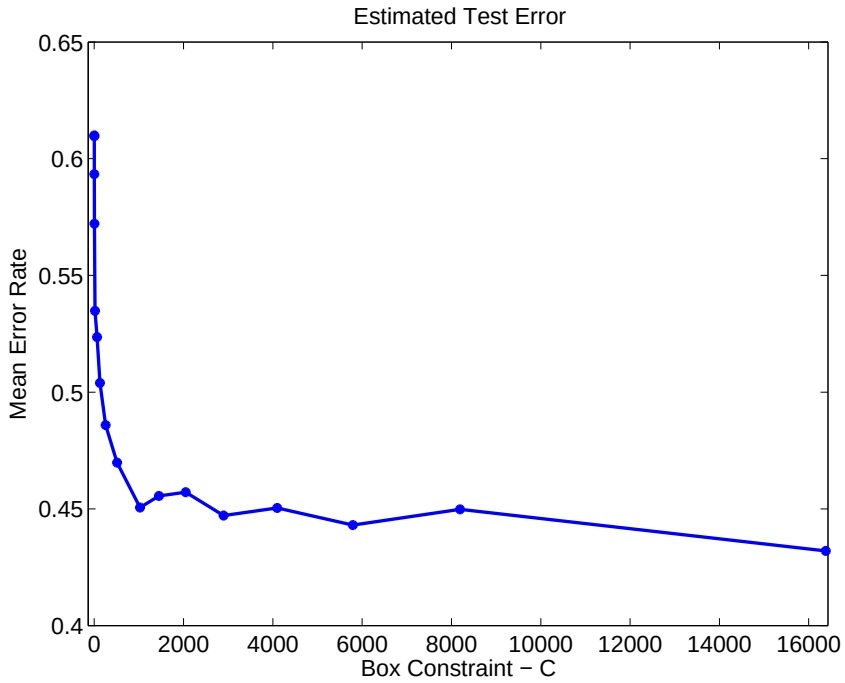


Figure 4.4: Total error rate averaged over 40 Monte Carlo iterations for grid-search. Even though the minimum is given by $C = 2^{14} = 16384$, the mean error rate for $C = 2^{10} = 1024$ upwards are statistically the same (for a standard deviation of ≈ 0.04). Hence, the empirical optimum hyper-parameter value is $C = 2^{10}$, at the curve's elbow.

discussed in Section 4.2. The results for classification procedure described in Section 4.3 are presented in Table 4.4.

Table 4.4: Performance metrics for feature vector proposed by Khan et al. (2008) (accelerometer AR coefficients + accelerometer SMA + accelerometer TA)

(%)	Climb stairs	Drink	Get up	Pour water	Sit down	Stand up	Walk
TPR	72.68	74.48	47.60	78.20	58.33	57.32	63
σ_{TPR}	9.06	8.12	11.59	6.26	9.51	9.21	9.11
TNR	94.18	92.06	92.72	96.11	95.03	93.59	95.06
σ_{TNR}	1.95	2.05	2.05	1.69	1.83	1.81	1.51
Accuracy	64.68 ± 4.51						

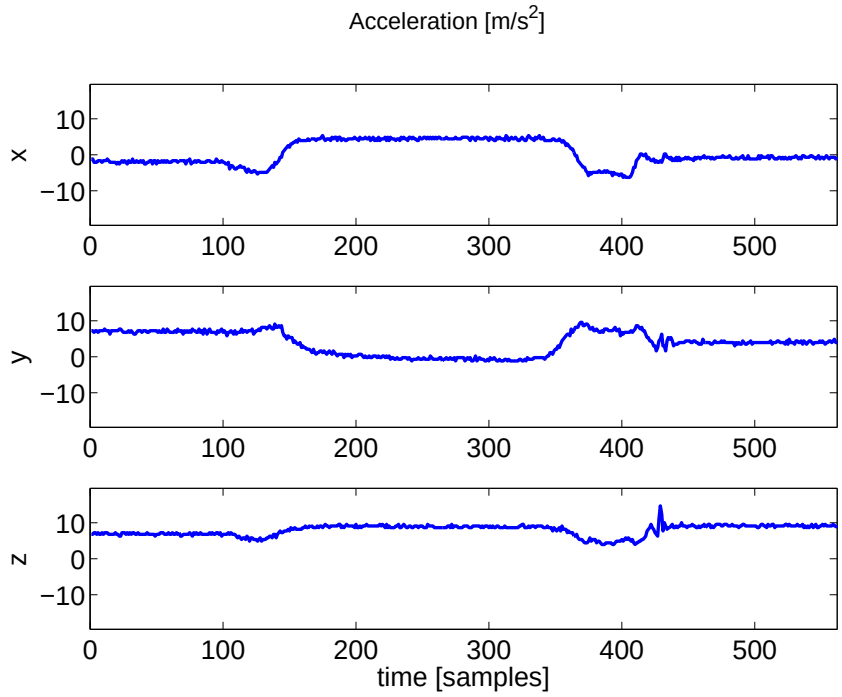


Figure 4.5: Tri-axial acceleration data sequence, with sampling rate of 32 Hz, for a trial of activity *drink*.

4.5 Results Using Proposed Method

The proposed method, described in Sections 3.4 and 4.3.1, is here investigated and discussed.

4.5.1 Attitude Estimation

Feature extraction is initiated by applying the trigonometric relation (3.6) and (3.7) to all data sequences in the considered dataset. No data pre-processing is done prior to this step. An example of roll and pitch signals is presented in Figure 4.6, which were obtained from accelerometer signals shown in Figure 4.5 for a trial of activity *drink*.

4.5.2 System Identification

Exploratory Data Analysis

Our approach to system identification assumes that a one-step ahead predictor is a good predictor to capture systems dynamics, which could be a reasonable assumption if accelerometer signals are not oversampled. In order to verify the effects of oversam-

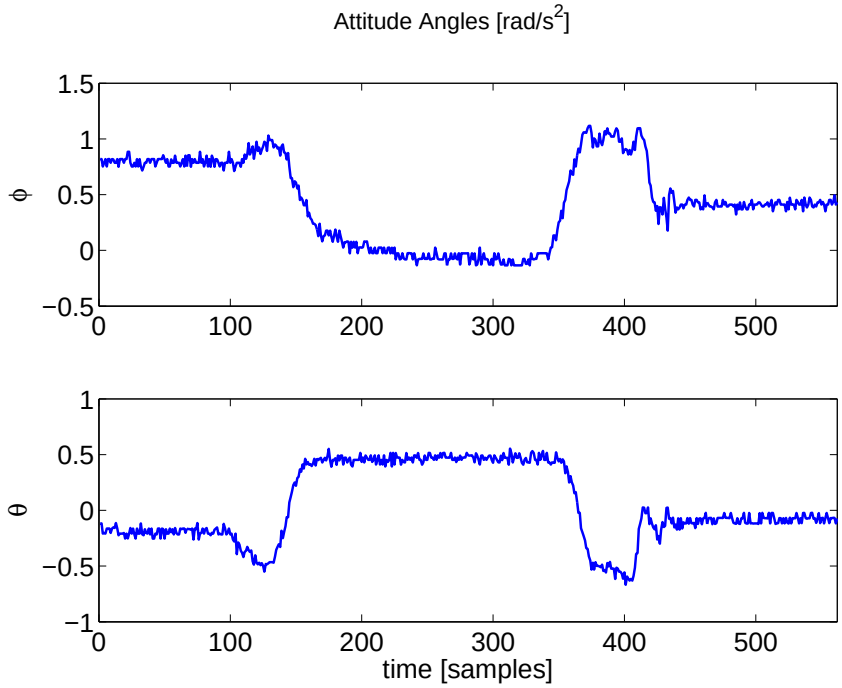
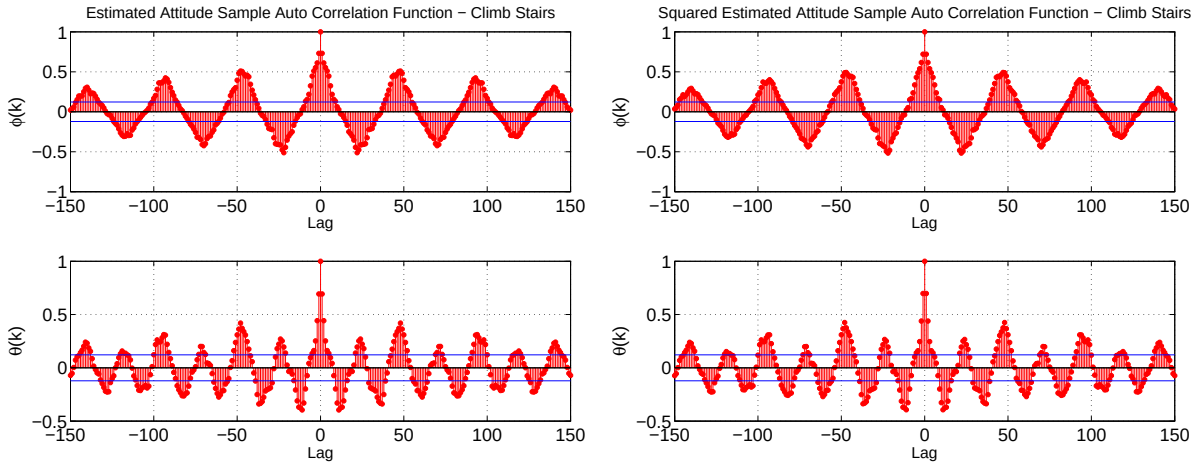


Figure 4.6: Roll (ϕ) and pitch (θ) angles data sequence for a trial of activity *drink*.

pling, the autocorrelation function of the signal and it's square are computed according to equations (2.11) and (2.13) and are shown in Figures 4.7 and 4.8.



(a) (b)

Figure 4.7: Signal's $\phi(k)$ and $\theta(k)$ autocofunction and signal's squared $\phi^2(k)$ and $\theta^2(k)$ autocorrelation function of a trial of activity *climb* stairs.

According to Aguirre (2015), the largest frequency of interest can be is found in the first minimum of the autocorrelation function (τ_y,τ_{y^2}), and we choose to work with the smallest lag of all, that is, $\tau_m = \min[\tau_y,\tau_{y^2}]$. He proposes as a rule of thumb that the minimum lag τ_m^* of a decimated signal $y^*(k) = y(\Delta k)$ obeys the relationship $10 \leq \tau_m^* \leq 20$, which is equivalent to $10\Delta \leq \tau_m \leq 20\Delta$.

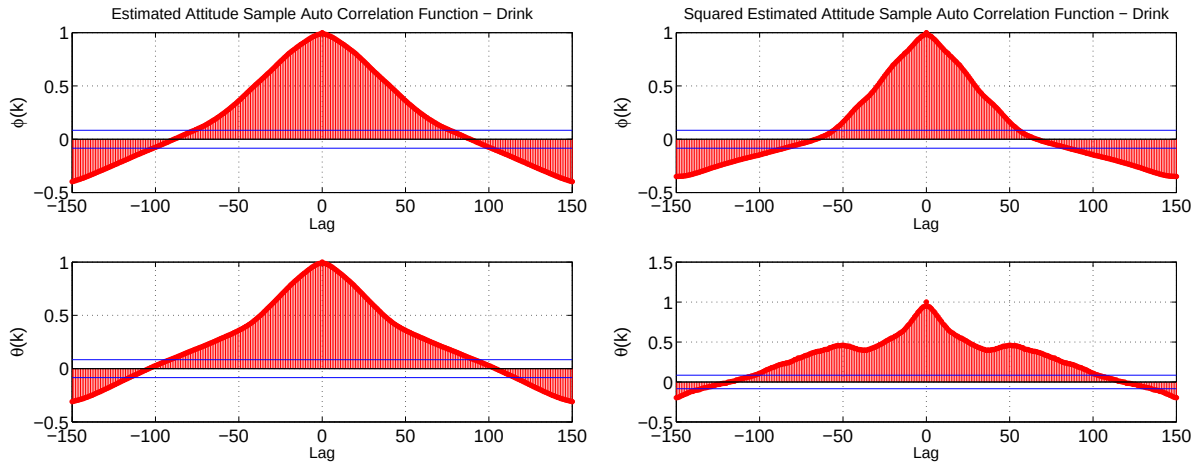


Figure 4.8: Signal's $\phi(k)$ and $\theta(k)$ autocofunction and signal's squared $\phi^2(k)$ and $\theta^2(k)$ autocorrelation function of a trial of activity *drink*.

As it can be seen in Figure 4.7, $\tau_m = 10$ which indicates that decimation is not needed. However, when observing Figure 4.8, $\tau_m \approx 70$ and so such signal is oversampled and could be decimated by a factor of 4. It can be concluded that activities such as climbing stairs and walking present a high frequency component for which the adopted sampling rate is adequate, as opposed to activities involving longer movements, such as drink and pour water. Nevertheless, this analysis leads us to believe that the measure acceleration signals are not too oversampled and that the one-step ahead prediction is still a reasonable predictor for our models.

It is important to mention that since the number of available data sequences is relatively large, the examples chosen in this Section are representative of the class as a whole but, yet, the main pattern described in the presented plots may vary considerably.

Model Structure and Validation

Many strategies to determine the autoregressive models' order were considered. The Akaike Information Criterion (AIC)[1] was useful to hint on a range of most common

[1] $AIC(n_\theta) = N\ln[\sigma^2_{error}(n_\theta)] + 2n_\theta$ where N is the number of data samples available, $\sigma^2_{\text{error}}(n_\theta)$ is the residues variance and $n_\theta = \dim[\hat{\boldsymbol{\theta}}]$ (Aguirre, 2015).

values. However, to directly apply the AIC algorithm, the multivariate autoregressive problem was adapted to a univariate problem. Since the AIC method is dependent on the order in which regressors are included (Aguirre, 2015), such changes on the VAR problem suggests that AIC results for our problem are not reliable.

The final choice for model's order was made by residues analysis, as defined in Section 2.1.4. The almost white behaviour of residues autocorrelation function, as shown in Figure 4.9(a), suggests that a 4^{th} order model is able to represent system's linear information. Hence, the resulting regressor vector was chosen to be

$$\psi^\top(k-1) = \left[\mathbf{y}^\top(k-1)\ \ \mathbf{y}^\top(k-2)\ \ \mathbf{y}^\top(k-3)\ \ \mathbf{y}^\top(k-4)\right], \tag{4.1}$$

where $\mathbf{y}^\top(k) = \begin{bmatrix}\phi(k) & \theta(k)\end{bmatrix}$ for the proposed model of the attitude angles. As pointed out earlier in this section, since a model is fitted for each trial and due to the amount of available trials, the general observed behaviour was taken into account for the decision making process. Thus, for the batch problem here considered, the regressor matrix is

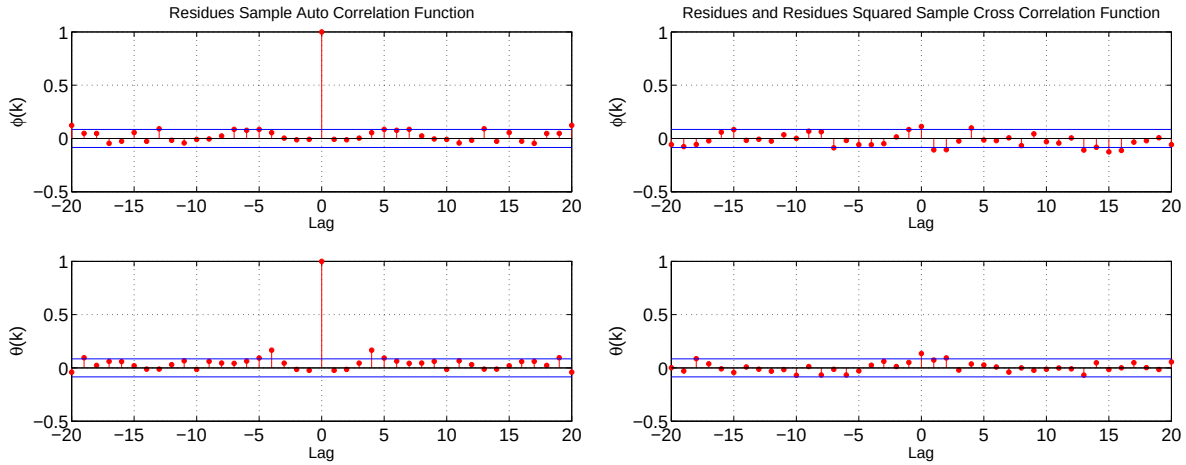
$$\Psi = \begin{bmatrix} \mathbf{y}(4) & \mathbf{y}(3) & \mathbf{y}(2) & \mathbf{y}(1) \\ \mathbf{y}(5) & \mathbf{y}(4) & \mathbf{y}(3) & \mathbf{y}(2) \\ \vdots & \ddots & & \vdots \\ \mathbf{y}(N-1) & \mathbf{y}(N-2) & \mathbf{y}(N-3) & \mathbf{y}(N-4) \end{bmatrix}, \tag{4.2}$$

where N is the length of a given data sequence, as defined in Chapter 2. A 4^{th} order model, for a bi-dimensional multivariate VAR model, produces 4×4 coefficients. As an example, the obtained model for the estimated attitude signals in Figure 4.6 is

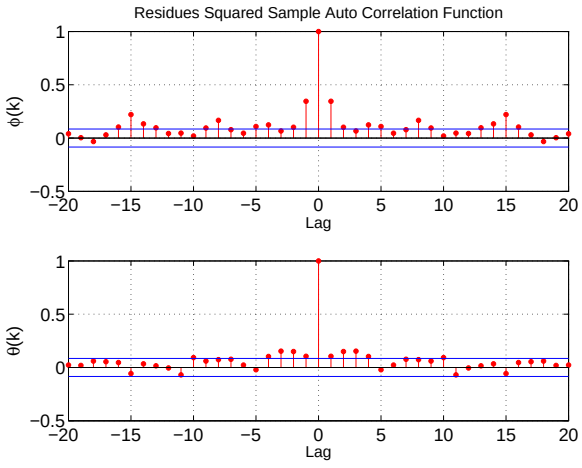
$$\begin{aligned}\begin{bmatrix}\phi(k)\\ \theta(k)\end{bmatrix} = &\begin{bmatrix}0.6249 & -0.0134\\ -0.0487 & 0.5984\end{bmatrix}\begin{bmatrix}\phi(k-1)\\ \theta(k-1)\end{bmatrix} + \begin{bmatrix}0.2993 & 0.0717\\ -0.0289 & 0.4510\end{bmatrix}\begin{bmatrix}\phi(k-2)\\ \theta(k-2)\end{bmatrix} + \\ &+\begin{bmatrix}0.1818 & -0.0442\\ -0.0393 & 0.1053\end{bmatrix}\begin{bmatrix}\phi(k-3)\\ \theta(k-3)\end{bmatrix} + \begin{bmatrix}-0.1112 & 0.0157\\ 0.0360 & -0.1642\end{bmatrix}\begin{bmatrix}\phi(k-4)\\ \theta(k-4)\end{bmatrix}.\end{aligned}$$

The statistically null crossed correlation function shown in Figure 4.9(b) indicates absence of correlation between residues and squared residues. Yet, by observing 4.9(c), the correlation lags $\tau = 1, 4, 8, 14$ and 15 for $\phi(k)$ and $\tau = 1, 2, 3$ and 4 for $\theta(k)$ greater than the confidence interval shows that residues still carry non-linear information, which is not expressed by the linear AR estimated model. Examples of the auto and crossed correlation functions for other activities are presented in Appendix A.

Validation in system identification, differently from the context of machine learning, refers to testing the retrieved model's generality by checking it's performance when presented with new data. The best way to do so is by run-free simulation. In the case of autonomous autoregressive models, this was achieved by defining the simulated initial



(a) (b)



(c)

Figure 4.9: Residues analysis for one-step ahead estimated 4$^{\text{th}}$ order model from a trial of activity *drink* (a) $r_{\xi}(\tau)$ (2.11) (b) $r_{\xi\xi^2}(\tau)$ (2.11) (c) $r_{\xi^2}(\tau)$ (2.12).

conditions as the initial values of the measured new data and continuously exciting the model with $e(k)$, in (2.7), considered to be Gaussian random process, with zero mean and variance chosen as 1% of the measured signal's range. The estimated model evaluated in Figure 4.9 was simulated with new data. By observing simulation results shown in Figure 4.10, it is evident that the estimated model is not general enough, mainly when signal $\theta(k)$ is considered.

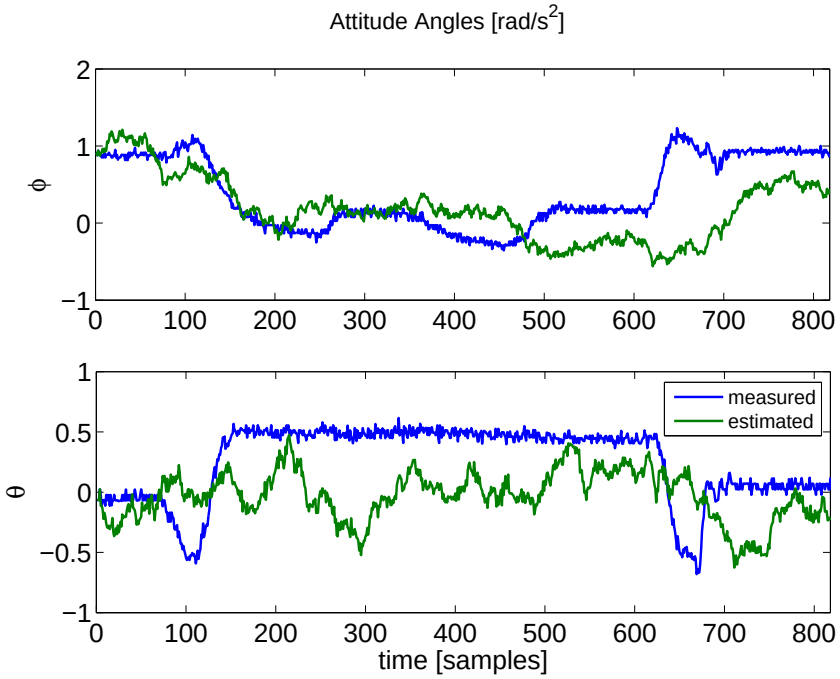


Figure 4.10: Measured and estimated by run-free simulation values for roll (ϕ) and pitch (θ) angles data sequence for a trial of activity *drink*.

However, since the problem considered in this work is classification, and not specifically system identification, validation on new data was not taken into account. In other words, as mentioned in Chapter 1, our main goal is finding a mathematical transformation which maps information to a different dimensional space in which intra-class variance is reduced and inter-class variance is increased, hence data becomes more separable.

4.5.3 Classification Results

Considering once again the experimental protocol defined in Section 4.3, 4 simulations were carried out:

- classification considering only VAR coefficients for estimated attitude angles (results in Table 4.5);

- classification for feature vector composed by attitude VAR coefficients, accelerometer VAR coefficients and accelerometer SMA (which will be referred to as *case* 1; results in Table 4.6 and in Figure 4.11);
- classification for feature vector composed by attitude VAR coefficients, accelerometer AR coefficients and accelerometer SMA (which will be referred to as *case* 2; results in Table 4.7).
- and classification for feature vector composed by raw accelerometer VAR coefficients, accelerometer SMA and accelerometer TA (which will be referred to as *case* 3; results in Table 4.8)

Table 4.5: Average and standard deviation of performance metrics considering information only from estimated attitude angles (feature vector composed only of attitude VAR coefficientes)

(%)	Climb stairs	Drink	Get up	Pour water	Sit down	Stand up	Walk
TPR	38.75	82.93	50.80	83.20	31.30	34.82	61.20
σ_{TPR}	7.26	7.11	9.13	6.013	10.11	9.71	9.72
TNR	91.73	94.11	90.40	96.48	93.09	90.79	90.31
σ_{TNR}	2.19	1.51	2.56	1.19	1.93	2.21	3.15
Accuracy	54.49 ± 2.79						

Since a simple average can be a biased estimator, due mainly to outliers, a more complete understanding of the hit rate can be achieved by observing the true positives box plot for case 1, presented in Figure 4.11.

An example of confusion matrix is shown in Figure 4.12. As pointed out, a relatively high level of confusion is seen between activities *climb stairs* and *walk*, and also between activities *sit down* and *stand up*. It is also seen that *sit down* and *walk* are also confused with *get up*. It must be noted that this result is not general, for it results from one Monte Carlo iteration only and therefore results from one specific data partitioning. Figure 4.13 presents histograms of the chosen values for the hyper-parameter C throughout the Monte Carlo loop and their respective range for each case.

4.6 Analysis

The obtained results for the method proposed in (Bruno et al., 2013) summarized in Table 4.3 indicate that such method favours some activities more than others. The

Table 4.6: Average and standard deviation of performance metrics for feature vector case 1 (attitude VAR coefficients + accelerometer VAR coefficients + accelerometer SMA)

(%)	Climb stairs	Drink	Get up	Pour water	Sit down	Stand up	Walk
TPR	73.57	82.93	59.60	85.40	48.15	41.61	75.60
σ_{TPR}	6.62	7.35	9.95	7.19	7.41	6.71	7.26
TNR	94.31	94.97	91.51	95.99	94.06	93.59	96.54
σ_{TNR}	1.84	1.38	2.56	1.75	1.90	2.23	1.08
Accuracy	66.55 ± 3.06						

Table 4.7: Average and standard deviation of performance metrics for feature vector case 2 (attitude VAR coefficients + accelerometer AR coefficients + accelerometer SMA)

(%)	Climb stairs	Drink	Get up	Pour water	Sit down	Stand up	Walk
TPR	70.54	81.55	53.80 84.60	54.82	52.14	68.80	
σ_{TPR}	6.86	7.09	7.64	5.70	8.33	9.61	8.63
TNR	93.27	95.57	90.90	96.91	94.88	94.50	95.06
σ_{TNR}	1.09	1.50	2.46	1.43	1.76	1.95	2.00
Accuracy	66.6310 ± 2.83						

Table 4.8: Average and standard deviation of performance metrics for feature vector proposeb by Khan et al. (2008) using VAR (accelerometer VAR coefficients + accelerometer SMA + accelerometer TA)

(%)	Climb stairs	Drink	Get up	Pour water	Sit down	Stand up	Walk
TPR	74.29	80.52	60.40	80.40	50.19	48.57	74.40
σ_{TPR}	8.19	9.21	9.11	7.89	7.81	9.20	9.83
TNR	93.93	94.81	91.45	96.88	94.00	93.87	96.42
σ_{TNR}	1.79	1.71	1.95	1.44	2.46	2.13	1.71
Accuracy	66.90 ± 3.89						

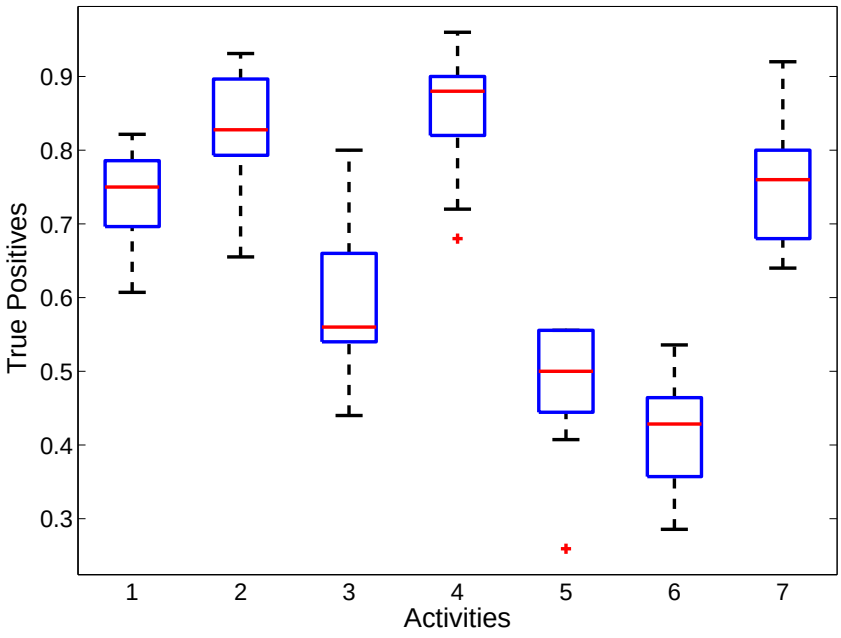
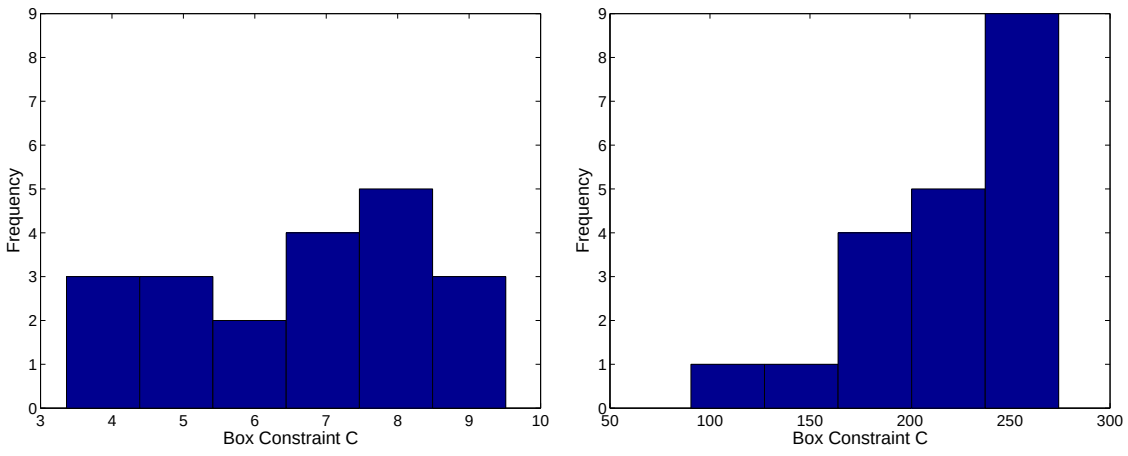


Figure 4.11: True positives box plot for case 1 (attitude VAR coefficients + accelerometer VAR coefficients + accelerometer SMA). 20 Monte Carlo iterations. Activities: (1) climb stairs (2) drink (3) get up from bed (4) pour water (5) sit down (6) stand up (7) walk.

		True						
		1	2	3	4	5	6	7
Predicted	1	20	0	3	1	2	0	(5)
	2	1	23	2	1	0	3	0
	3	0	1	13	0	(6)	1	(4)
	4	0	3	1	22	1	1	0
	5	0	2	3	1	14	(6)	1
	6	2	0	3	0	(4)	17	1
	7	(5)	0	0	0	0	0	14

Figure 4.12: An exemple of confusion matrix of one Monte Carlo iterration in case 2 (attitude VAR coefficientes + accelerometer AR coefficients + accelerometer SMA). Activities: (1) climb stairs (2) drink (3) get up from bed (4) pour water (5) sit down (6) stand up (7) walk.



(a) (b)

Figure 4.13: Chosen values for hyper-parameter C through grid-search. 20 Monte Carlo iterations. (a) Case 1 (b) Case 2.

very low hit rate for activities *climb stairs*, *sit down* and *stand up* can be attributed to the fact that these models are shorter, that is, their estimated curves have less points in time than the other models, since the execution of such activities is faster than the other ones. Authors in (Allen et al., 2006) already reported that their GMM system had great difficulty distinguishing the transitions sit-stand and stand-sit, even though using data from a waist worn accelerometer.

We first compare classification results obtained using features extracted from acceleration sequences only (Table 4.4), as proposed by Khan et al. (2008), and results obtained using features extracted from estimated Euler angles only (Table 4.5). The increase of almost 8% of the TPR average for activities *drink* and *pour water* indicate that the considered wrist attitude attributes in fact enhances recognition of activities rich in arm movement. On the other hand, the decrease of almost 20% in TPR average for ambulation activities (except for *walk*, whose resulting average is practically the same) implies that wrist orientation alone is not sufficient to properly discriminate such type of human activity. We consider that the high difference in between activities' recognition hit rates is related to the intrinsic frequency associated to each case. Activities such as *climb stairs* and *walk* can be regarded as recurrent activities with a comparatively high frequency, where as *drink* and *pour water* are also recurrent, however with much lower frequencies. Activities *get up from bed*, *stand up* and *sit down* are transitional, and there fore resemble an impulse response.

The addition of the *acceleration* VAR coefficients and the SMA value to the attitude VAR coefficients feature vector practically doubles the TPR average for *climb stairs* and increases the result for other ambulation activities in 10-15%, with respect to results using estimated Euler angles features only. When compared to results in Table 4.4, the increase of almost 10% for *drink*, *pour water*, *get up* and *walk* is accompanied by a 10% hit rate decrease for activities *sit down* and *stand up*. Nevertheless, a decrease of 2-3% is observed for the hit rate variability when considering the expanded feature vector.

In order to quantify the effects of taking axes correlation into account, we consider the simulation of Case 2 and 3, which the results are expressed in Tables 4.7 and 4.8. By comparing with Table 4.6, Case 2 actually presents higher hit rates for *sit down*, *stand up* and *walk* (about 10%) and *drink* and *pour water* (about 2%) than Case 1. Activities *climb stairs* and *get up* present higher hit rates (about 5%) in Case 1 than Case 2. Even though Case 2 seams to have better TPR averages than Case 1, it's variability is higher. For the GMM+GMR scenario, authors report in (Bruno et al., 2012) that the approach which considers correlation between axes also offers increased accuracy. Results in Table 4.8 accentuate the effects of the use of VAR coefficients by replacing the AR coefficients for raw accelerometer by the VAR estimated models. They indicate increase in practically all activities when compared with results of Table 4.4, which suggests that the better performance for the proposed feature vector is due mostly to the information embedded in axis correlation.

Chapter 5

Conclusions

This chapter reviews the work described in this dissertation and gives the main conclusions, addresses the main difficulties and shortcomings encountered throughout this research, and suggests possible directions for future works.

5.1 Dissertation Overview

In this work, we considered the automated recognition of human activities. We have focused on the problem of feature enrichment considering activities of daily living where arm movements have a more important role in their execution.

Based on studies which show that rotation contains relevant information for the HAR problem, and considering scenarios in which only accelerometer measurements are available, we estimate attitude Euler angles from available data, assuming that the changes in the measured acceleration are mainly due to changes in the projection of gravity, as described in Section 3.4. Test results have shown that wrist orientation alone is not enough to properly discriminate ambulation activities, as shown in Table 4.5, but increases recognition rate of activities rich in arm movement, as seen by comparing results in Table 4.4 and 4.6.

The use of multivariate/vector autoregressive models instead of univariate autoregressive models, commonly used in previous works, as review Section 3.3, enabled the incorporation of the temporal correlation between components of the measured wrist's acceleration. The increase on the performance for when such attributes were taken into account, as seen when comparing Tables 4.6 and 4.7, indicates that such correlation contain useful information and contributes to class separability.

Choices regarding pipeline design and approaches to testing protocol, as described in Section 4.3.3, were mostly made based on the limited number of samples available in the chosen dataset. The class imbalance introduced by some activities which have much fewer samples was dealt with by sample stratification and by adopting the one-against-one approach to the multiclass problem.

5.2 Main Results

Reviewing the main objectives of this study listed in Chapter 1, the following conclusions are worth registering:

1. It was verified that the overall hit rate for activities rich in arm movements have increased up to 10% when, in addition to raw acceleration features, new ones extracted from estimated wrist attitude were considered;

2. The effects of not neglecting the correlation between the components of the measured acceleration vector were verified by the use of multivariate model. Performance results increased for *most* of ambulation activities, which suggests that correlation does carry useful information. Yet, results for the activities of daily living did not vary, which confirms that estimated Euler angles are sufficient to identify rich in arm movement activities;

3. It was verified that while it is possible to attain reasonable activity recognition results when considering both ambulation and daily living activities, some activities namely *get up*, *sit down* and *stand up* were poorly identified. Such can be attributed mainly to fact that these are transitional activities;

4. Not only the overall hit rate was evaluated, but also it was quantified the variability of our main results by applying Monte Carlo cross-validation;

5. It was observed that linear autoregressive models are sufficient to represent some of human activities, but not adequate to represent others, such as *sit down* and *stand up* which present stronger non-linearities (see Figures A.4(a) and A.5(a)). The use of different model structures for different activities is not a viable option for automatic classification, since once a new observation is received no prior information is available to decide which model structure to be used.

5.3 HAR: Encountered Issues and Future Perspectives

The issues raised throughout the work here described lead to different directions and possibilities to be explored in the HAR problem. We list here some possible approaches to the issues we came across with and consider some other analyses that would make results more comprehensive, such as carrying out a more systematic comparison of hit rate by applying hypothesis testing to verify whether the observed difference in performance results between methods are actually statistically significant.

The greatest obstacle encountered throughout this work, apart from the limited number of available trials in dataset, is related to data prepping. The approach proposed by Bruno et al. (2013) relies completely on the training data being the same length for each motion and having the gesture starting approximately at the same sample. In this

work, the data used for autoregressive modelling were not synchronized beforehand, for such would require separate analysis of each data sequence. Automating the cropping process was a solution tested, where 15% of the beginning and end of each time sequence was removed but proved to make classification more troublesome. Manual data segmentation where information of the only motion is selected might increase classification performance, which is expected since prior information is introduced to the process.

In view of the dependency of data segmentation and the eventual application of the proposed method in on-line context, it would be interesting to investigate the possibility of classification on run-time, by using recursive least-squares to estimate the autoregressive coefficients. The use of windows on run-time data does not, however, exclude the fact that training is done off-line and that segmentation is still crucial to ensure good motion representativeness.

The order of the autoregressive model in our classification problem can be considered a hyper-parameter and could have been chosen via grid-search, as described in Section 2.2.2. A more thorough and systematic study is suggested for the choice of the VAR model order and its effects on classification. A possible starting point could be discarding the vectorized approach to the multivariable problem, in which the regressor vector for both signals are the same. By such, it would be possible to adopt different model structures for each autoregressive model. Moreover, this vectorized approach to autoregressive modelling and the fixed a priori model order increases the number of attributes and might introduce redundant information. For such a reason, is left as a suggestion for further reaserch a study of feature selection as described in Guyon and Elisseeff (2003).

An issue considered, yet not treated here, is the subject-to-subject variability. The autoregressive coefficients obtained from time series carry information of duration and speed in which each activity is performed and since each person performs differently, such dependency might cause model bias and is aggravated when most of training samples belong to the same agent or when the they do not compose a sufficiently diverse enough population. In order to reduce such influence, we considered the possibility of resampling the $\phi(k)$ and $\theta(k)$ signals on the phase plane $\phi \times \theta$ by creating a virtual grid and substituting the continues signal values by the value of the center of the respective grid unit which contains the signal. In this case, only transitions are taken into account and, in principle, the time factor is no longer relevant.

However, this data processing step introduces an extra hyper-parameter to be adjusted: the grid coarseness. Its effects are similar to those of sampling frequency, because a very coarse phase plane grid can lead to loss of information of signal dynamics and a finer grid will cause the resampled signals to "oversample" and tend to the original signal. The optimal grid coarseness is not the same for each activity since it is related to the dominant frequency, or still, the movement duration, which is different for each activity as previously seen. This idea was not deeply explored and should be

further investigated, together with other solutions.

Authors in Bruno et al. (2013) dealt with the difference of speed between trials, due either to irregular sampling or individual characteristics, by using the Dynamic Time Warping technique, which computes the similarity between two numerical sequences and therefore takes into account variations in time and speed. However, no significant improvement in results were observed and when real-time requirements must be met, this method is not ideal because it is costly computational wise.

The presented work was developed in the context of a R&D project which aims to recognize human activities in real time from sensory data provided by inertial measurement unit (IMU) of smartwatch and smartphone. This problem setting is followed by limitations associated with processing capability and battery consumption. Moreover, the system is priority driven, hence is not task dedicated and sampling frequency is not constant. This becomes a hindrance when one attempts to work with sensor fusion for observations of time sequences from different sensors might not correspond to the same time instant of a performed activity. Such fact does not affect performance too harshly if the shift between time sequences is much smaller than the window size and for statistical features like mean and variance. However, the lack of synchronization might be an issue for some cases like for the method proposed in Section 3.4, for AR models assume uniform sampling.

Considering the context of this project in which both accelerometer and gyrometer measurements are available, an interesting study involving sensor fusion could be done. It would be interesting to compare activity recognition performance and computational performance for the case where rotational information is obtained from gyromenter with the case in which it is estimated from accelerometer, as the in the method proposed in this work.

The merging of system identification and machine learning approach introduces interesting questions and widens research possibilities. An initial idea for research was to focus on the optimization problem embedded in the modelling process and to investigate cost functions that would capture the need for both the autoregressive model and the classifier. However, this would demand a much longer time to be accomplished, and therefore it is left as a suggestion for a future research project.

Part of the results presented in this dissertation were obtained through research on a project titled “HAR-HEALTH: Reconhecimento de Atividades Humanas associadas a Doenças Crônicas”, sponsored by Samsung Eletrônica da Amazônia Ltda. under the terms of Brazilian federal law No. 8.248/91.

Bibliography

Aguirre, L. A. (2015). In *Introdução a Identificação de Sistemas: Técnicas Lineares e Não Lineares: Teoria e Aplicação*, pages 221–252. Editora UFMG.

Aguirre, P. L. R. (2017). Autoregressive modeling of wrist attitude for feature enrichment in human activity recognition. *Congresso Brasileiro de Inteligência Computacional*.

Allen, F. R., Ambikairajah, E., Lovell, N. H., and Celler, B. G. (2006). Classification of a known sequence of motions and postures from accelerometry data using adapted gaussian mixture models. *Physiological measurement*, 27(10):935.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Clarendon Press.

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, pages 144–152, New York, NY, USA. ACM.

Bruno, B., Mastrogiovanni, F., and Sgorbissa, A. (2014). A public domain dataset for adl recognition using wrist-placed accelerometers. In *Robot and Human Interactive Communication, 2014 RO-MAN: The 23rd IEEE International Symposium on*, pages 738–743. IEEE.

Bruno, B., Mastrogiovanni, F., Sgorbissa, A., Vernazza, T., and Zaccaria, R. (2012). Human motion modelling and recognition: A computational approach. In *Automation Science and Engineering (CASE), 2012 IEEE International Conference on*, pages 156–161. IEEE.

Bruno, B., Mastrogiovanni, F., Sgorbissa, A., Vernazza, T., and Zaccaria, R. (2013). Analysis of human behavior recognition algorithms based on acceleration data. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 1602–1607. IEEE.

Candy, J. V. (2016). *Bayesian signal processing: classical, modern, and particle filtering methods*, volume 54. John Wiley & Sons.

Cawley, G. C. and Talbot, N. L. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11(Jul):2079–2107.

Chen, Y. and Xue, Y. (2015). A deep learning approach to human activity recognition based on single accelerometer. In *Systems, man, and cybernetics (smc), 2015 ieee international conference on*, pages 1488–1492. IEEE.

Doebelin, E. O. and Manik, D. N. (2007). Measurement systems: application and design.

Dong, Y., Hoover, A., Scisco, J., and Muth, E. (2012). A new method for measuring meal intake in humans via automated wrist motion tracking. *Applied psychophysiology and biofeedback*, 37(3):205–215.

Gupta, P. and Dallas, T. (2014). Feature selection and activity recognition system using a single triaxial accelerometer. *IEEE Transactions on Biomedical Engineering*, 61(6):1780–1786.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182.

He, Z. (2010). Activity recognition from accelerometer signals based on Wavelet-AR model. In *Progress in Informatics and Computing (PIC), 2010 IEEE International Conference on*, volume 1, pages 499–502. IEEE.

He, Z.-Y. and Jin, L.-W. (2008). Activity recognition from acceleration data using AR model representation and SVM. In *Machine Learning and Cybernetics, 2008 International Conference on*, volume 4, pages 2245–2250. IEEE.

Hsu, C.-W., Chang, C.-C., and Lin, C.-J. (2016). A practical guide to support vector classification.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning*. Springer.

Jordao, A., Torres, L. A. B., and Schwartz, W. R. (2018). Novel approaches to human activity recognition based on accelerometer data. *Signal, Image and Video Processing*.

Karantonis, D. M., Narayanan, M. R., Mathie, M., Lovell, N. H., and Celler, B. G. (2006). Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring. *IEEE transactions on information technology in biomedicine*, 10(1):156–167.

Katz, S., Downs, T. D., Cash, H. R., and Grotz, R. C. (1970). Progress in development of the index of adl. *The gerontologist*, 10(1_Part_1):20–30.

Khan, A., Lee, Y., and Lee, S. (2010a). Accelerometer's position free human activity recognition using a hierarchical recognition model. In *e-Health Networking Applications and Services (Healthcom), 2010 12th IEEE International Conference on*, pages 296–301. IEEE.

Khan, A. M., Lee, Y.-K., and Kim, T.-S. (2008). Accelerometer signal-based human activity recognition using augmented autoregressive model coefficients and artificial neural nets. In *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*, pages 5172–5175. IEEE.

Khan, A. M., Lee, Y.-K., Lee, S., and Kim, T.-S. (2010b). Human activity recognition via an accelerometer-enabled-smartphone using kernel discriminant analysis. In *Future Information Technology (FutureTech), 2010 5th International Conference on*, pages 1–6. IEEE.

Khan, A. M., Lee, Y.-K., Lee, S. Y., and Kim, T.-S. (2010c). A triaxial accelerometer-based physical-activity recognition via augmented-signal features and a hierarchical recognizer. *IEEE transactions on information technology in biomedicine*, 14(5):1166–1172.

Khan, A. M., Siddiqi, M. H., and Lee, S.-W. (2013). Exploratory data analysis of acceleration signals to select light-weight and accurate features for real-time activity recognition on smartphones. *Sensors*, 13(10):13099–13122.

Krassnig, G., Tantinger, D., Hofmann, C., Wittenberg, T., and Struck, M. (2010). User-friendly system for recognition of activities with an accelerometer. In *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2010 4th International Conference on-NO PERMISSIONS*, pages 1–8. IEEE.

Kratz, S., Rohs, M., and Essl, G. (2013). Combining acceleration and gyroscope data for motion gesture recognition using classifiers with dimensionality constraints. In *Proceedings of the 2013 international conference on Intelligent user interfaces*, pages 173–178. ACM.

Lara, O. D. and Labrador, M. A. (2013). A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys and Tutorials*, 15(3):1192–1209.

Madsen, H. (2007). *Time series analysis*. CRC Press.

Mathie, M. (2003). *Monitoring and Interpreting Human Movement Patterns Using a Triaxial Accelerometer*. PhD thesis, Faculty of Engineering. Sydney, Australia, University of New South Wales.

Mika, S., Ratsch, G., Weston, J., Scholkopf, B., and Mullers, K.-R. (1999). Fisher discriminant analysis with kernels. In *Neural networks for signal processing IX, 1999. Proceedings of the 1999 IEEE signal processing society workshop.*, pages 41–48. Ieee.

Pillonetto, G., Dinuzzo, F., Chen, T., De Nicolao, G., and Ljung, L. (2014). Kernel methods in system identification, machine learning and function estimation: A survey. *Automatica*, 50(3):657–682.

Platt, J. et al. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.

Poppe, R. (2010). A survey on vision-based human action recognition. *Image and Vision Computing*, 28(6):976–990.

Ramos-Garcia, R. I., Muth, E. R., Gowdy, J. N., and Hoover, A. W. (2015). Improving the recognition of eating gestures using intergesture sequential dependencies. *IEEE journal of biomedical and health informatics*, 19(3):825–831.

Satopaa, V., Albrecht, J. R., Irwin, D. E., and Raghavan, B. (2011). Finding a "kneedle" in a haystack: Detecting knee points in system behavior. *2011 31st International Conference on Distributed Computing Systems Workshops*, pages 166–171.

Shoaib, M., Bosch, S., Incel, O. D., Scholten, H., and Havinga, P. J. (2015). A survey of online activity recognition using mobile phones. *Sensors*, 15(1):2059–2085.

Sifuzzaman, M., Islam, M., and Ali, M. (2009). Application of wavelet transform and its advantages compared to fourier transform.

Suykens, J. A., Van Gestel, T., De Brabanter, J., De Moor, B., and Vandewalle, J. (2002). *Least Squares Support Vector Machines*. World Scientific.

Welling, M. (2005). Fisher linear discriminant analysis. *Department of Computer Science, University of Toronto*, 3(1).

Xu, Q.-S., Liang, Y.-Z., and Du, Y.-P. (2004). Monte carlo cross-validation for selecting a model and estimating the prediction error in multivariate calibration. *Journal of Chemometrics*, 18(2):112–120.

Appendix A

Results for WHARF's Database

A.1 Estimated Attitude and Respective Residues Whiteness Test for Each Activity Considered

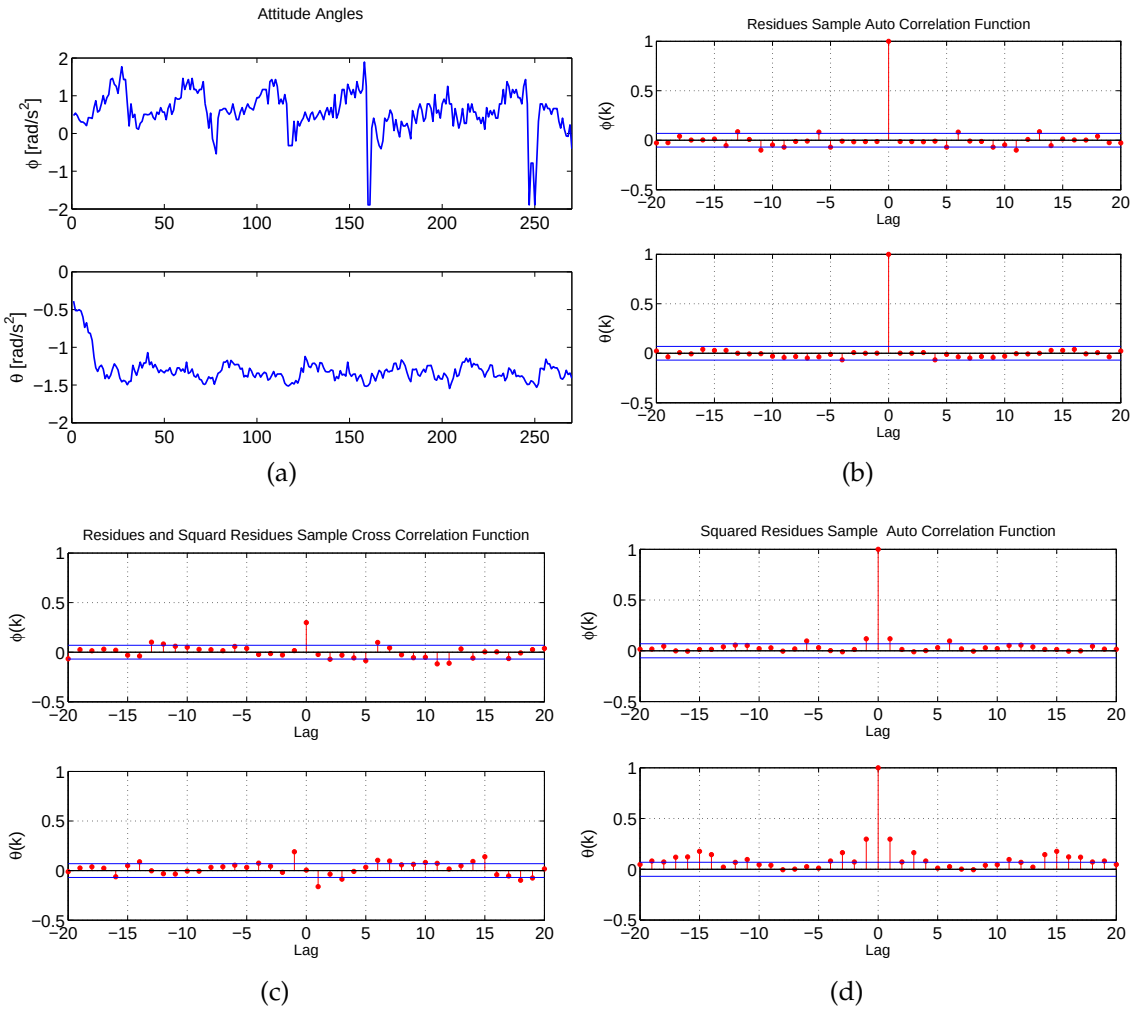


Figure A.1: (a) Roll (ϕ) and pitch (θ) angles data sequence for a trial of activity *climb stairs*. Residues analysis for estimated 4^{th} order model (b) $r_{\xi}(\tau)$ (2.11) (c) $r_{\xi\xi^2}(\tau)$ (2.11) (d) $r_{\xi^2}(\tau)$ (2.12).

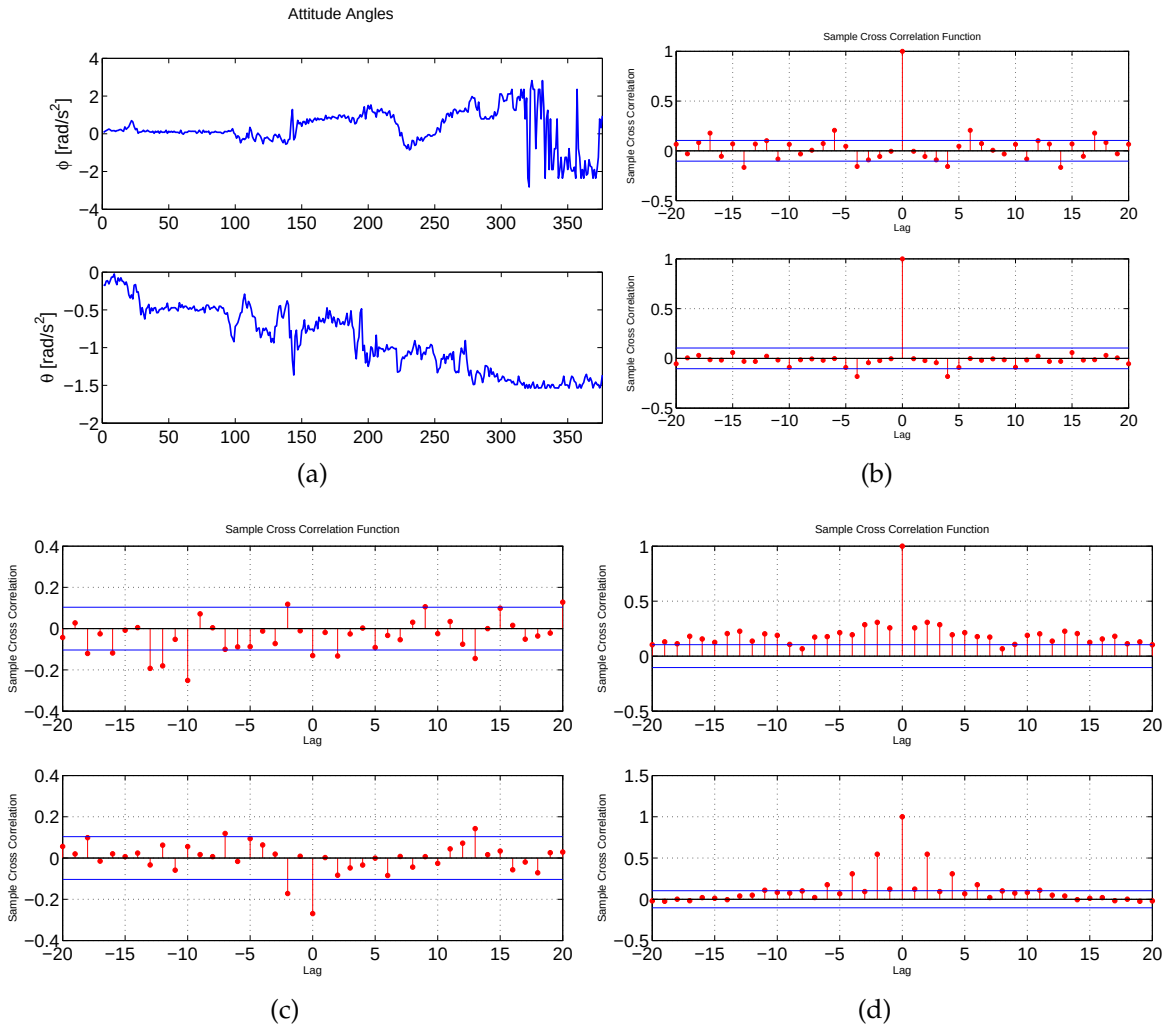


Figure A.2: (a) Roll (ϕ) and pitch (θ) angles data sequence for a trial of activity *get up* from bed. Residues analysis for estimated 4$^{\text{th}}$ order model (b) $r_{\xi}(\tau)$ (2.11) (c) $r_{\xi\xi^2}(\tau)$ (2.11) (d) $r_{\xi^2}(\tau)$ (2.12).

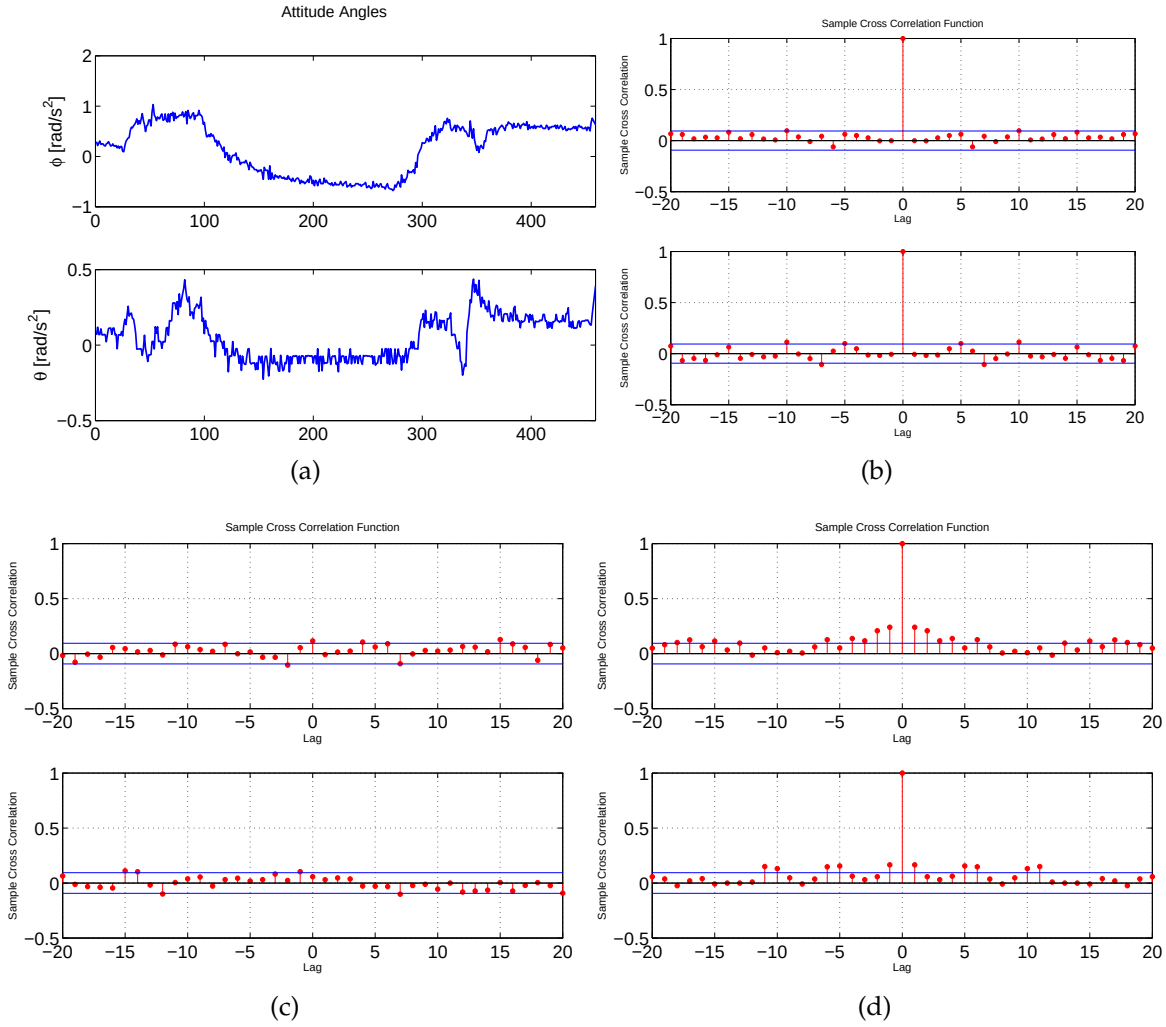


Figure A.3: (a) Roll (ϕ) and pitch (θ) angles data sequence for a trial of activity *pour water*. Residues analysis for estimated 4^{th} order model (b) $r_{\xi}(\tau)$ (2.11) (c) $r_{\xi\xi^2}(\tau)$ (2.11) (d) $r_{\xi^2}(\tau)$ (2.12).

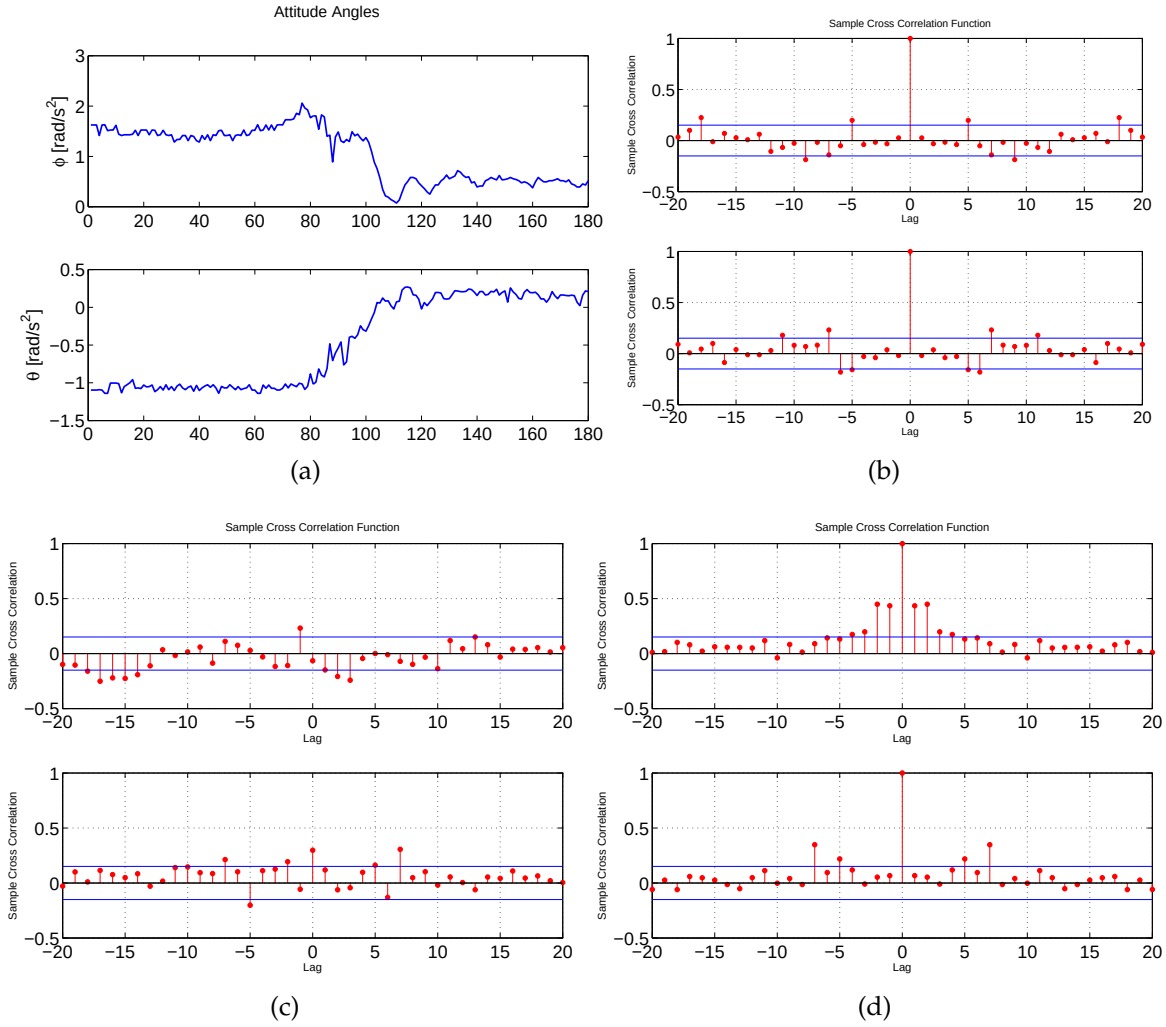


Figure A.4: (a) Roll (ϕ) and pitch (θ) angles data sequence for a trial of activity *sit down chair*. Residues analysis for estimated 4$^{\text{th}}$ order model (b) $r_{\xi}(\tau)$ (2.11) (c) $r_{\xi\xi^2}(\tau)$ (2.11) (d) $r_{\xi^2}(\tau)$ (2.12).

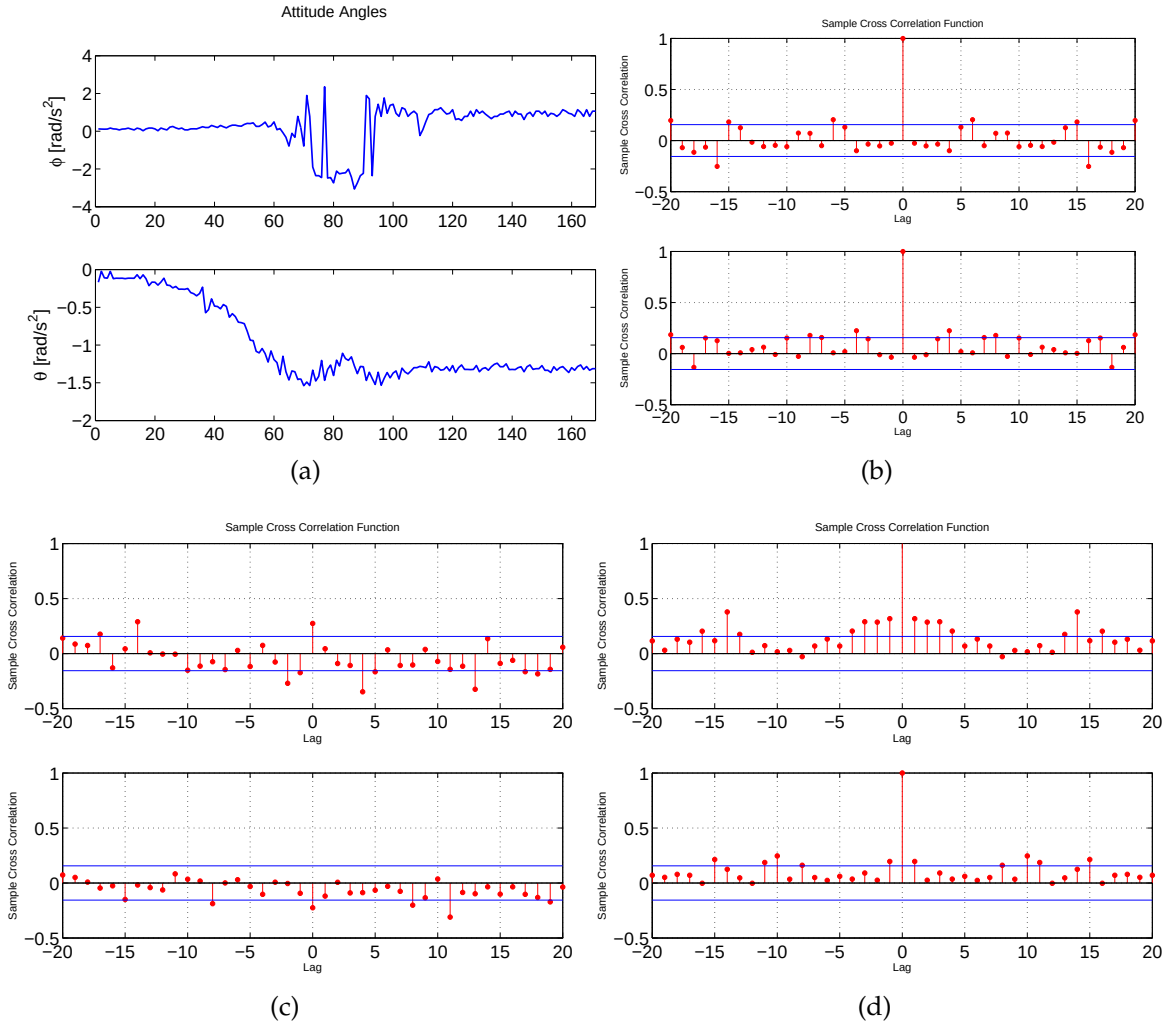


Figure A.5: (a) Roll (ϕ) and pitch (θ) angles data sequence for a trial of activity *stand up chair*. Residues analysis for estimated 4^{th} order model (b) $r_{\xi}(\tau)$ (2.11) (c) $r_{\xi\xi^2}(\tau)$ (2.11) (d) $r_{\xi^2}(\tau)$ (2.12).

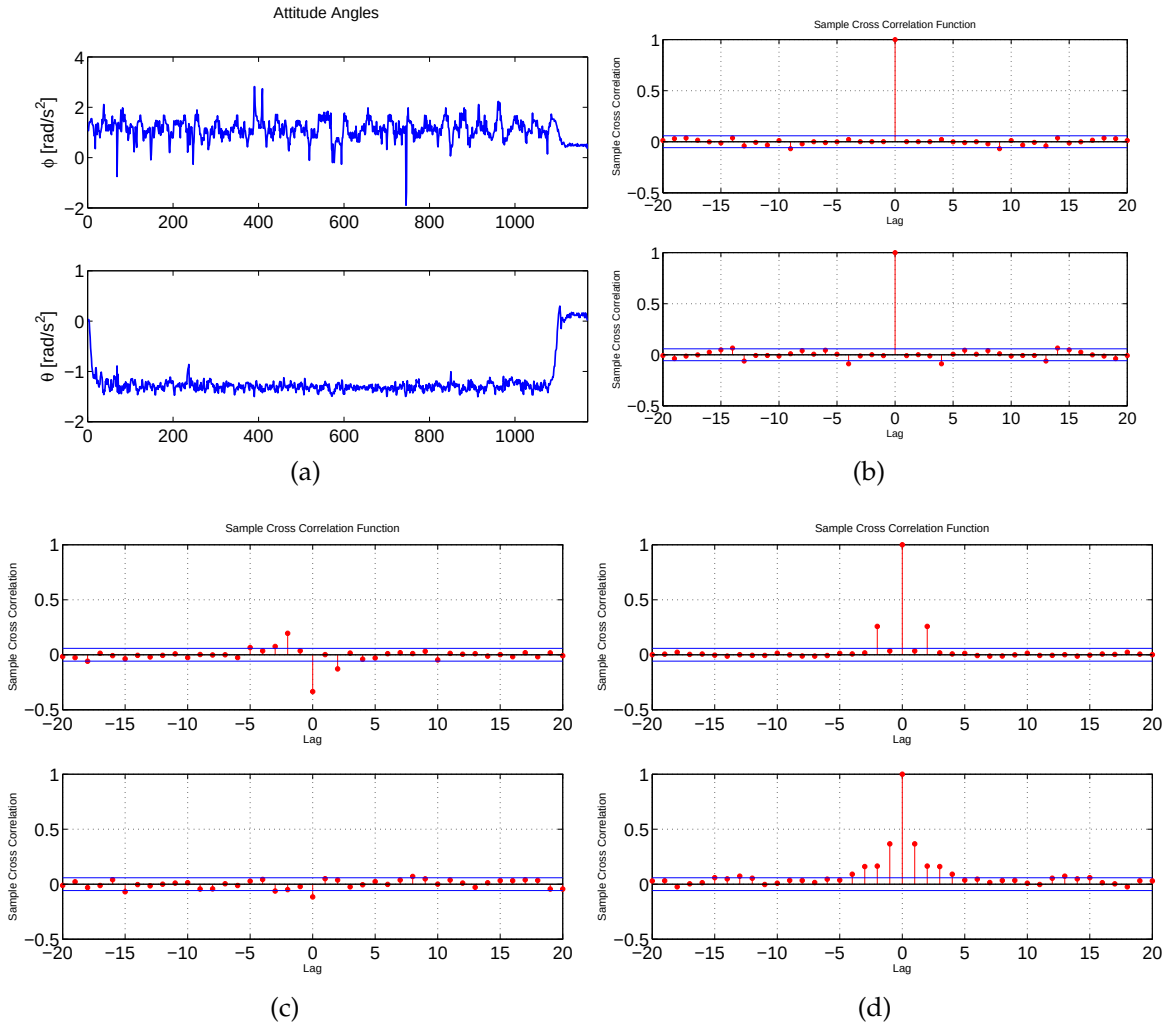


Figure A.6: (a) Roll (ϕ) and pitch (θ) angles data sequence for a trial of activity *walk*. Residues analysis for estimated 4^{th} order model (b) $r_{\xi}(\tau)$ (2.11) (c) $r_{\xi\xi^2}(\tau)$ (2.11) (d) $r_{\xi^2}(\tau)$ (2.12).

Appendix A

Annex

Gmail **Priscila Aguirre <priaguirre@gmail.com>**

Questions related to article

barbara.bruno@unige.it <barbara.bruno@unige.it> 27 de novembro de 2017 08:52
Para: Priscila Aguirre <priaguirre@gmail.com>

Dear Priscila,

thank you for your interest in our work!
I apologize for the long delay in my response, I have been away and unfortunately with very little time to devote to emails. I hope the delay did not have a negative big impact on your work!

In that article, classification was performed off-line and an analysis of the possibilities pattern was not yet considered. By the way, if you're interested in that, you can find more information on the paper "Using fuzzy logic to enhance classification of human motion primitives": https://link.springer.com/chapter/10.1007/978-3-319-08855-6_60

In the article you mention, a sliding window moves over the trial to recognize and yields a classification, i.e., it returns the label of the activity which more closely matches the data within the window (the possibility based on Mahalanobis distance). The whole trial is classified as an instance of the activity with more labels (i.e., the one that more closely matched the highest number of windows).
We could follow this rationale because the trials used for evaluation were cut so that they only contained one execution of one activity.

In later articles, once we introduced the idea of analysing the rise-fall pattern of the possibilities, the system gave a label only upon the recognition of a rise-fall pattern and we used all the labels to compute the performance of the system.

Concerning the synchronization of trials, you are very right, it is a crucial step for the accuracy of the modelling! What we did is the following:

1. manually synchronize 3/4 trials (i.e., cutting them to have equal length and to have the gesture starting more or less at the same sample)
2. build a model with those trials
3. take a recording that you want to add to the modelling dataset and classify it against the model (using as sliding window size the size of the model)
4. identify the window which is closest to the model and add it to the modelling set.
5. repeat from step 2 until enough trials have been added to the model.

I hope I have answered your question, don't hesitate to contact me if you have additional questions, or thoughts about the work that you would like to share!

Have a nice day,
Barbara

-

Inviata da Windows Mail

Da: Priscila Aguirre
Data invio: giovedì 9 novembre 2017 12:21
A: Barbara Bruno