Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Cybernetics

Use of data from smart hospital bed

Bachelor thesis

Charlotte Panušková

Bc. programme: Open Informatics
Branch of study: Computer and Informatic Science
Supervisor: Ing. Martin Macaš, Ph.D.

Prague, May 2018

Thesis Supervisor:
Ing. Martin Macaš, Ph.D.
Czech Institute of Informatics, Robotics and Cybernetics
Czech Technical University in Prague
Jugoslávských partyzánů 1580/3
160 00 Praha 6, Dejvice
Czech Republic

BACHELOR'S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Panušková Charlotte** Personal ID number: **456154**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Cybernetics**

Study program: **Open Informatics**

Branch of study: **Computer and Information Science**

II. Bachelor's thesis details

Bachelor's thesis title in English:

Use of Data from Smart Hospital Bed

Bachelor's thesis title in Czech:

Využití dat z chytrého nemocničního lůžka

Guidelines:

1. Explore structure of data obtained from smart hospital bed LINET and perform preliminary demonstration measurements and evaluation (what to measure, what is the reliability, under which conditions etc.).
2. Propose and discuss possible ways of utilization of those data in a real environment. (Possible scenarios are fall risk detection and prediction, recognition of particular patients or some changes, anomaly detection, risky states detection, activity recognition, etc.).
3. Choose a particular way of usage, which is realizable and testable under current circumstances (experimental smart home at CIIRC, current state of the provided smart bed, its sensors and equipment). Realize and evaluate the selected proposal. Final realization must cover also experimental data acquisition and real implementation (e.g. in Matlab).

Bibliography / sources:

[1] Manuál k nemocničnímu lůžku LINET.
[2] Sánchez, D., Tentori, M., & Favela, J. (2008). Activity recognition for the smart hospital. IEEE intelligent systems, 23(2).
[3] Wong, S., & Tan, C. S. (2013). Smart hospital bed.
[4] Nguyen, H. H. (2016). Advanced assistive control strategies for smart hospital beds (Doctoral dissertation).

Name and workplace of bachelor's thesis supervisor:

Ing. Martin Macaš, Ph.D., Cognitive Neurosciences, CIIRC

Name and workplace of second bachelor's thesis supervisor or consultant:

Date of bachelor's thesis assignment: **11.01.2018** Deadline for bachelor thesis submission: **25.05.2018**

Assignment valid until: **30.09.2019**

Ing. Martin Macaš, Ph.D. Supervisor's signature	doc. Ing. Tomáš Svoboda, Ph.D. Head of department's signature	prof. Ing. Pavel Ripka, CSc. Dean's signature

III. Assignment receipt

The student acknowledges that the bachelor's thesis is an individual work. The student must produce her thesis without the assistance of others, with the exception of provided consultations. Within the bachelor's thesis, the author must state the names of consultants and include a list of references.

.

Date of assignment receipt Student's signature

Declaration

I declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the methodical instruction for observing the ethical principles in the preparation of university theses.

In Prague, May 2018

....
Charlotte Panušková

Abstrakt

Stále širší využití technologií ve zdravotnictví bude v budoucnu nevyhnutelné. Jednou z již běžně používaných technologií je inteligentní zdravotní lůžko. Tato práce prezentuje údaje získané z inteligentního zdravotní lůžka a navrhuje jejich využití pro detekci polohy těla metodami klasifikace vzorů. Byla testována řada klasifikátorů jak v intrapersonálním, tak v interpersonálním scénáři. V obou scénářích bylo testováno několik klasifikátorů. Pro oba byl nejlepším klasifikátorem naivní Bayesovský klasifikátor s průměrnou chybou 8.1% pro intra-personální a 23.1 % pro inter-personální klasifikaci. Nejhorší klasifikační chyby dosáhl rozhodovací strom s průměrnou chybou 19.8% pro inter- a 33.4% pro intra-personální klasifikaci. Tato práce také diskutuje možné budoucí rozšíření. Konkrétní třídy poloh jsou porovnávány podle jejich rozlišitelnosti a výsledky jsou popsány a shrnuty.

Keywords: Chytré nemocniční lůžko, detekce poloh, klasifikátory.

Abstract

The usage of technologies in medical field in future is indisputable. One of already commonly used technology is smart medical bed. This thesis presents and analyses the data obtained from smart medical bed and proposes their usage for posture detection using pattern classification methods. Two scenarios were examined. Intra-personal classifier is trained on the same person on which it will be used. Inter-personal classifier can classify data from person it has never seen before. Several classifiers were tested in both scenarios. For both the best classifier was naive Bayes with average error 8.1% and 23.1% for intra-personal classification and inter-personal classification, respectively. The worst classifier turned out to be decision tree with average error 19.8% and 33.4%, respectively. The particular classes of postures are compared according to their discriminability and all results are described and summarized.

Keywords: Smart medical bed, posture detection, classifiers.

Acknowledgements

Foremost, I must express my sincere gratitude to my thesis supervisor, Ing Martin Macaš PhD, for his patient guidance, encouragement and advice he has provided throughout my time as his student.

Secondly, I would like to thank my friends Luisa Pánková, Alexandra Čechová and Tomáš Vobořil for their precious time spent on my experiments.

Last but not the least, I would like to thank my mother for supporting me throughout my life.

List of Tables

List of Figures

List of Acronyms

DT Decision trees. viii, ix, 14, 18, 21–23, 25, 30, 32

ECG Electro-cardiogram. 4

FCM Fuzzy C-Means. 4

FSR Force-Sensing Resistor. 3, 4

HO Hold-out. 15, 18

kNN k-Nearest Neighbour. viii, ix, 4, 13, 21–25, 30–33

LB Linear Bayes. viii, ix, 14, 18, 31, 33

NB Naive Bayes. viii, ix, 14, 18, 20–25, 27, 31, 33

nCV n-fold cross-validation. 15, 18

NN Neural Network. 4

PRTOOLS Pattern Recognition Toolbox. viii, ix, 13, 18, 30–33

PUP Pressure Ulcer Prevention. 3, 4

SMLT Statistics and Machine Learning Toolbox. viii, ix, 13, 18, 21, 22, 24, 25, 32, 34

SVM Support Vector Machine. ix, 3, 4, 14, 23, 32, 34

Contents

Chapter 1

Introduction

1.1 Motivation

In recent years, the demand for good healthcare services is growing. To increase the quality of provided care, healthcare institutions (e.g. hospitals, nursing homes, etc.) are expected to change with the help of rapidly evolving technologies. Another reason for the intervention of technology is that nursing is a tiring, physically demanding and not well paid job. Also, according to [1], there is a concern that as the population ages, a relatively high proportion of nurses and caring professionals will retire. Possible solution is to try to make nursing not so human dependent. For these and many other reasons, the future importance of support devices in hospitals is indisputable.

One of such devices supporting medical staff in everyday practice is the smart hospital bed. Recently, medical beds are more than a resting place for patients. They keep a record about patient's condition, and with such ability, they help healthcare workers to deliver a better care. Variety of smart medical beds provides sundry of conveniences. Examples are advanced bed positioning, temperature monitoring or pressure sensors, which are focused in this thesis.

1.2 Problem description

The goal of this thesis is to explore, interpret and utilize the data obtained from smart hospital bed LINET. There were many possible scenarios. After thorough evaluation of data, the body posture detection was chosen. It is an important application for many reasons that are detailed in chapter 2. The development of the posture detection algorithm was considered as a common supervised machine learning task. A set of labeled data (inputs assigned to a specific posture) was collected and used as training and testing data for classifiers. Using the training data, a training algorithm creates a function that assigns input data to particular posture. To obtain a better understanding about the detection problem it was desirable to try more classifiers and make a comparison among them. Due to a the lack of collected data, only simple classifiers were chosen and implemented in MATLAB.

1.3 Structure of Thesis

The thesis is divided into six chapters. In chapter 2 there is a brief survey of posture detection methods from other researchers and engineers. The exploration of useful data that can be acquired from bed is described in chapter 3. A description of algorithms and methods used in this work can be found in chapter 4. The experiments and their results are presented in chapter 5. Finally, in chapter 6, the thesis is summarized and concluded.

Chapter 2

Related work

It has been demonstrated by various researchers that the detection of postures can be useful and important in many applications. One example is Pressure Ulcer Prevention (PUP) for elderly and bedridden patients - a major and costly issue in care institutions. Detecting and keeping a record of the patient's posture on bed helps care givers reposition patient more efficiently and reduce the risk of developing a pressure ulcer [2]. Other examples are the vital signs monitoring, sleep quality assessment [3], sleep disorders analysis [4] or patient safety and caregiving efficacy [5].

In [6], the authors proposed a detection method using Bayesian classification based on kurtosis and skewness estimation. For data acquisition they used 16 long-narrow Force-Sensing Resistor (FSR) sensors. Assuming that the prior probability of postures is uniformly distributed, they adopted Gaussian distribution to model the bed postures. With three different postures - Supine, Right Lying and Left Lying - they achieved an average of 78.7% precision rate, that were critically influenced by the lying angle of the patient.

The use of kurtosis and skewness estimation was also presented in [7]. Furthermore, for sleeping posture classification they applied principal component analysis and SVMs. Their pressure sensing beds consisted of 16 or 56 FSR sensor pads. For patients with low mobility, 16-sensor pad was able to detect 3 positions with high accuracy (average 81.43%) which dropped when patients started to move and sleep in different positions.

In [2], an image based algorithm has been proposed. Authors used a flexible pad with sensors distributed in 32x64 rectangular grid, that can be considered as gray scale pressure image. Algorithm itself has three main steps: normalization, eigenvalue projection and kNN classifier. To evaluate their posture classification methodology they collected five different postures: right foetus, left foetus, right yearner, left yearner and supine. The most problematic posture was the right foetus as it was confused with the right yearner in 5.1%. They managed to obtain total accuracy of 94.3%.

In [8], the authors introduced Neural Network (NN) for posture classification with only four sensors, i.e. two piezoelectric and two pressure sensors. Postures of five classes were considered: out of bed, sitting, lying down, lying left and lying right. The total dataset consisted of 120 hours corresponding to 20 500 measurements (5 postures x 4100 measurements) from a single subject. The overall performance reached up to 98.3% of accuracy.

NN was also proposed in [5]. Moreover, the authors introduced Fuzzy C-Means (FCM) algorithm to transform the pressure contours and identify the regions of interest with high pressure for PUP. Their sensor pad contained 18x12 array of FSR. To obtain the data, six people were asked to perform six designed configurations of three lying postures and stay in each for one minute. They collected 900 pressure images in total. Their average posture recognition rate was up to 95.89%.

The authors in [4] developed a posture detector with usage of unconstrained measurements of Electro-cardiogram (ECG) signals using 12 capacitively coupled electrodes and a conductive textile sheet. They extracted features on the basis of the morphology of the QRS and applied them to five different classifiers. The highest performance was obtained for SVM with radial basis function kernels with an accuracy of 98.4% for estimation of four body postures - supine, right lateral, prone, and left lateral.

The writers in [9] presented the use of a bed-based optical pressure sensor array to recognize sitting, lie-to-sit, and lying postures. They collected data using a pressure sensor array and video cameras and compared eight pressure signal features and three classification techniques. The measured subjects were divided into 4 distinct groups: young healthy, older healthy, older post-stroke, and older post-hip-fracture. The highest accuracy reached feature called weighted number of active sensors, exceeding almost 90% by all three tested classifiers - SVM, NN and kNN.

Alternative sensing techniques were proposed in [10], where authors used temperature sensors and acoustic sensors. Authors stated that "this method can be used standalone, can augment other sensors, or can validate data from pressure or visual-based systems."

To remove noises enabling acoustic posture detection, a least mean square algorithm was deployed. With the usage of thresholding algorithm, the postural changes events are then recognized from audio records. In a controlled environment, they were able to detect over 90% of postural changes.

Chapter 3

Data structure

In this chapter it is described and explained what kind of data can be obtained from the bed. There is also a description of demonstrative experiment that was made to evaluate reliability and overall behaviour.

The data is sent by a standard packet formatted in 256 bytes. Majority of information acquired from the packet is not interesting for posture detection and is not used in this work. The interesting ones are described here.

3.1 Data from four strain sensors

The main data that can be acquired from the bed are quantities from analog-to-digital converter of strain gages that are inserted into the body of the bed. There are four of them in total and are labeled from 142 to 145, which is their position in sent packet. Their approximate position is depicted in Fig. 3.1. The sensors are briefly described in the next sections.

3.1.1 Strain gage

Strain gage is a passive metal sensor, firmly connected with the body of the bed, that measures force applied on the bed. Metal sensor posses a 'strain sensitivity' - ratio of the relative change in electrical resistance of a conductor to the applied relative change in conductor length. Strain sensitivity is a dimensionless quantity. [11]

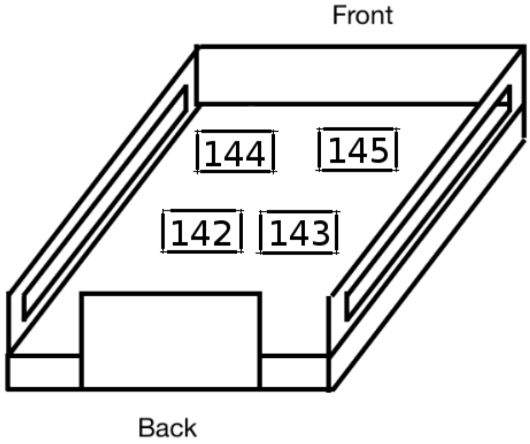


Figure 3.1: Approximate position of sensors

3.1.2 Repeatability

In order to understand the strain gage data, demonstration measurements were performed. Moreover, the experiment was used to test the repeatability of the strain gage data. The process is described in Table 3.1. A measurement was repeated three times, two in a row and last one five days after. Figure 3.2 shows the process of the experiment. As expected, the experiments demonstrate that sensors returns similar quantities when the same experiment is repeated.

Time	**Action**
0s - 60s	without any weight
60s - 90s	subject sits on the left side of bed
90s - 120s	subject lays down
120 - 150s	subject sits on the left side of bed
150s - 210s	without any weight

Table 3.1: Process of repeatability test

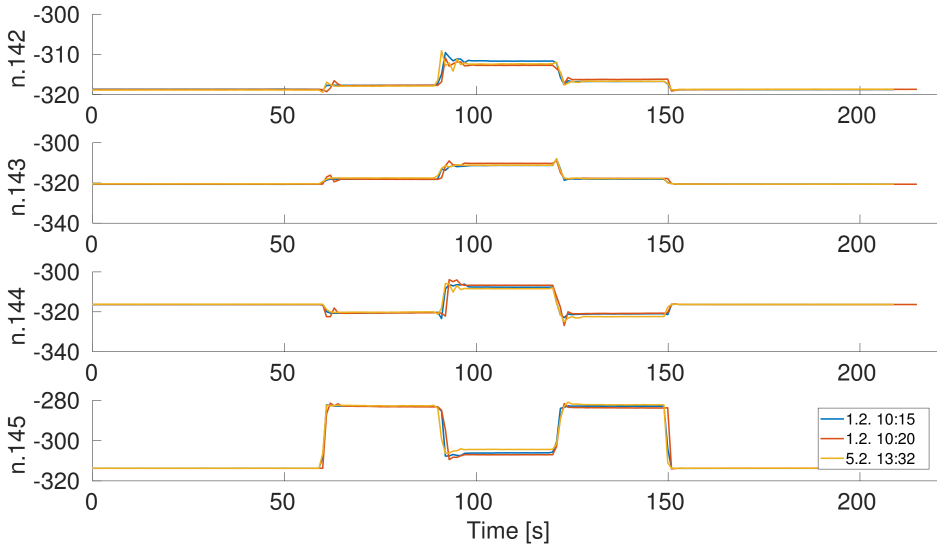


Figure 3.2: Plot of the signals obtained from the sensors

3.1.3 Dependence

Because of the bed structure, it is probable that some strain gages are correlated. In Table 3.2, there is a computed correlation coefficient for each strain gage from experiment described in section 3.1.2. There is a significant correlation between sensors indexed 142 and 143. It seems that this is due to the fact that the sensors are inserted close to each other in the upper part of the bed. There is also a negative correlation between sensors indexed 144 and 145. Presumably this factor is because the sensors lie far from each other on x - axis, so when one is loaded by down, the other one is unburdened. Conversely, sensors indexed 142 and 145 and sensors indexed 143 and 145 appear to be relatively independent.

	142	143	144	145
142	1	0.9759	0.6810	0.2062
143	0.9759	1	0.6033	0.2938
144	0.6810	0.6033	1	-0.5409
145	0.2062	0.2938	-0.5409	1

Table 3.2: Correlation of strain gages

3.2 Data with centre of mass position

Other data that can be acquired from the smart hospital bed is the centre of mass position. It is represented as two separated signals - x and y position of the centre of mass - that are computed from the quantities obtained from the strain gages. In Figure 3.3, it is outlined where the x and y position lies.

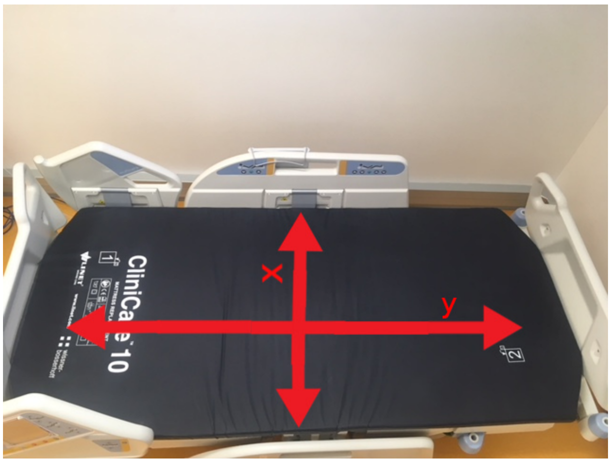


Figure 3.3: x and y position of the centre of mass

To obtain a better understanding of the obtained quantities and to check their validity, demonstration and validation measurements were performed and are described in the following section.

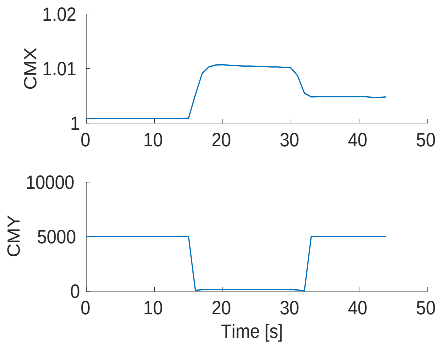
3.2.1 Static weight experiment

In this experiment, each corner and center of the bed were loaded down by a weight u and for a 15 seconds forced manually with extra weight v. The schedule of the measurement is described in Table 3.3.

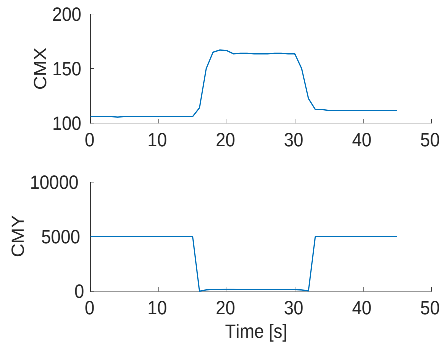
Time	**Weight**
0s - 15s	only weight u
15s - 30s	extra force $u + v$
30s - 45s	only weight u

Table 3.3: Process of experiment with weight

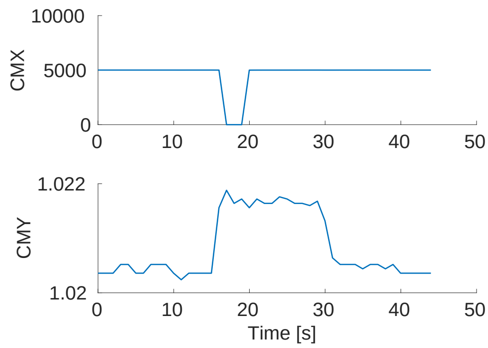
Following graphs show the change of the centre of mass during the experiment. First thing that should be observed is that it may seem that the values have a large range. This can be seen at the plot from the upper left corner Figure 3.4a. The centre of mass on the y-axis acquired value around 10 000 when the bed was weighted only with weight u and overflowed to values around 100 when weighted with extra weight w. Comparing the plots from the upper left and upper right corner from Figure 3.4, centre of mass on x-axis acquire values around 10 000 on left side and around 100 on right side.



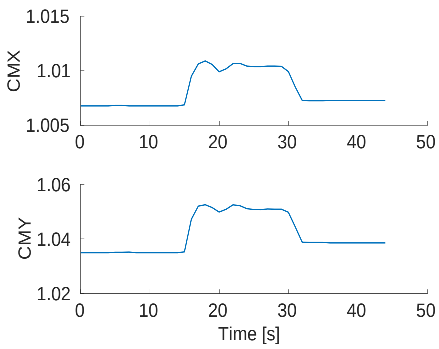
(a) Upper left corner



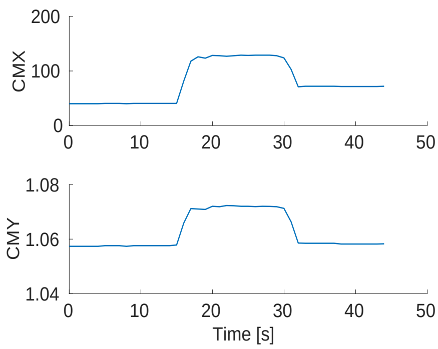
(b) Upper right corner



(c) Centre



(d) Bottom left corner



(e) Bottom right corner

Figure 3.4: Centre of mass position signals

3.2.2 Moving weight experiment

Another experiment that was created to observe the centre of mass was with moving weight. The weight was moved continuously horizontally (see Figure 3.5) and vertically (see Figure 3.7) for several seconds. Moving horizontally, only on x-axis was observed a changing signal (Figure 3.6). As can be seen, first 20 seconds the values descended, then overflowed and started to ascend from 0. These tests revealed that the negative values are encrypted reversely and as numbers higher than 10 000. The following pseudocode fixes the problem, where $centreOfMass$ denotes the position of centre of mass on x-axis or y-axis and $threshold$ is the value that centre of mass never exceeds as positive number. In this application $threshold$ is set to 9000. As illustrated in Figure 3.8, centre of mass courses as could be expected.

if $CMX \geq threshold$ **then**

  $CMX \leftarrow CMX - 10000$

end if

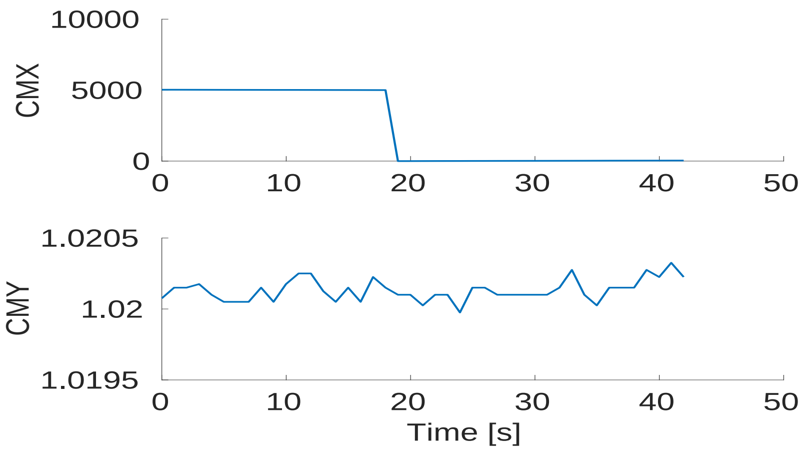


Figure 3.5: Both axes

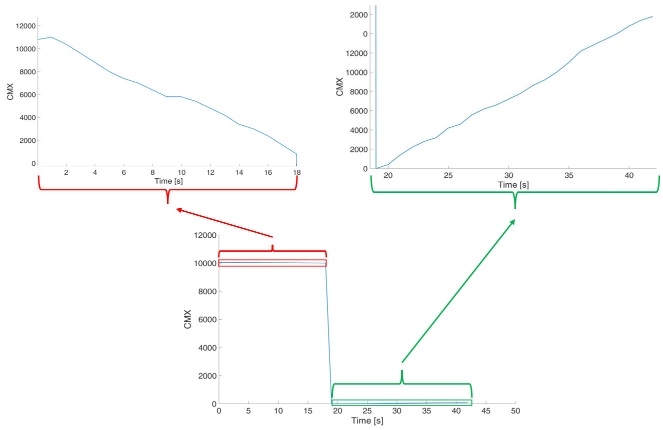


Figure 3.6: Only X-axis

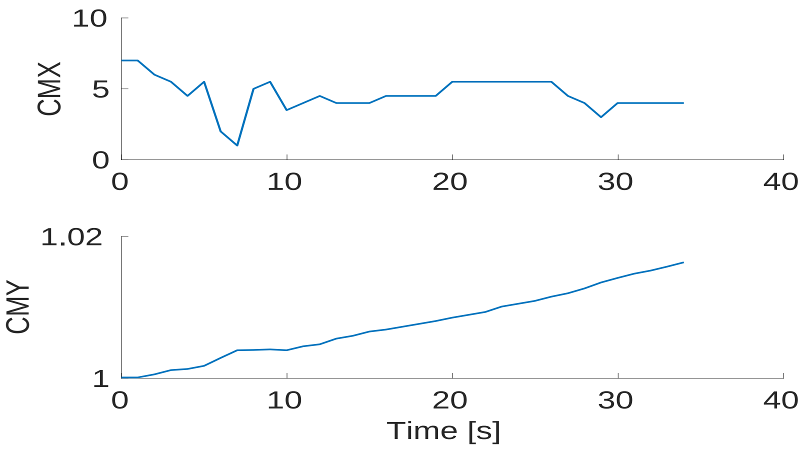
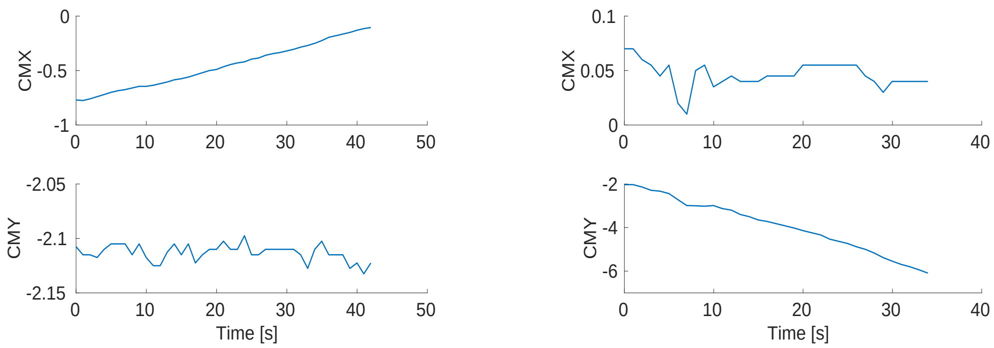


Figure 3.7: Both axes



(a) Fixed horizontal move

(b) Fixed vertical move

Figure 3.8: Plot of fixed move

Chapter 4

Main components

This section describes the main components of this work including classifiers (section 4.1) and error estimation (section 4.2) methods.

4.1 Classifiers

Classification is a task of classifying the data into previously known classes. The algorithms that implement classification are called classifiers. Classifiers play crucial part in this work. The particular implementation of used classifiers can be found in the documentation for the PRTOOLS toolbox and Statistics and Machine Learning Toolbox that were used. Two different toolboxes were used in parallel to avoid invalidity of the results caused by an incorrect implementation.

4.1.1 *k*-Nearest Neighbour Classifier

k-Nearest Neighbour is a simple classifier that always finds the k nearest (usually Euclidean distance is used) training data instances which are the closest to the testing instance and assigns the instance into the most frequent class amongst the k nearest neighbours. When $k = 1$, the classifier simply finds nearest neighbour and assigns the testing data to the found class.

4.1.2 Bayesian classifiers

Bayesian classifiers are based on Bayesian theorem - a rule that expresses a posteriori probability for each class from the likelihood and class prior probability.

$$p(l|\mathbf{x}) = \frac{p(\mathbf{x}|l)p(l)}{p(\mathbf{x})},$$

where l denotes the class and $\mathbf{x}$ denotes the vector of input features. Further, it assigns the feature vector into the class, which maximizes the a posterior probability. In this work, uniform prior probabilities are considered.

Naive Bayes

Naive Bayes is called naive because it assumes that all features are conditionally independent given class, which is an unrealistic assumption. However, the classifier is often surprisingly effective [12].

Linear Bayes

Linear Bayes differs from NB in assumption of independent features. LB supposes that the joint probability distribution of the features follows a multivariate normal distribution [13]. It considers the same covariance matrix for all likelihoods, which leads to the linear decision boundary.

4.1.3 Decision trees

Decision trees are one of the simplest, but popular classifiers. Their main advantages are simplicity and illustrative nature. They are easily interpretable for people, even for those who have no knowledge about machine learning. The creation of DT is described in Algorithm 1.

Algorithm 1 Decision tree algorithm

repeat
1. Test the most important attribute and place it on top
2. Split the data set into subsets using the attribute values.
3. Build an independent tree for each subset.

until all leaf nodes of all branches are created

4.1.4 SVM

Support Vector Machine is a discriminative classifier which finds an optimal hyperplane that separates the training data. The idea of this classifier is to find the elements with smallest distance between them from different classes called support vectors. Then find the ideal hyperplane with the greatest possible margin between support vectors. This guarantee the optimality of classifier.

4.2 Error estimation

Error estimation in an important part of pattern classification. The main tool is to split data into training and testing part, train the classifier on the training set and test it on the testing set. Different error estimation methods are suitable in different cases. Two error estimation methods that fit for our task are described in the following subsections.

4.2.1 Hold-out validation

Hold-out (HO) is a simplest method of validation. It splits data set D into two disjunctive subsets - training one T and testing one $D - \{T\}$. Classifier fits a function to the training part only. Then tries to predict labels of testing part, which labels it does not know. The empirical error is equal to a ratio of wrongly estimated samples to total number of samples. In this thesis, HO estimate is used for evaluation of intra-personal classifiers that can classify only postures from one particular person.

4.2.2 k-fold cross-validation

A little more sophisticated method is n-fold cross-validation (nCV) method. It divides data set D into n disjunctive subsets $\{D_i\}_{i=1\ldots N}$ of similar size. At each of N steps, a classifier is trained on the training part $D \setminus D_i$ and than tested on the remaining part D_i. The empirical error is equal to the sum of wrongly estimated samples to total number of samples divided by N. The nCV provides almost an unbiased estimation of true error.

In this thesis, a "subject-based" nCV estimate is used for evaluation of inter-personal classifiers, that can classify postures of any subject (also subject that it has never seen before). The n was set to the number of subjects and each fold contained all data from one subject. This enabled to estimate error of classifier trained on $n - 1$ subjects and tested on data from subject that has never been used for training.

Chapter 5

Posture detection

The detection of posture was selected as the main task. In this section the whole process is explained.

5.1 Data acquisition

In order to evaluate the posture classification methodology, a set of five different postures was measured on five different volunteers whose weights are summarized in Table 5.1. Each subject stayed for 30 seconds in particular posture, then changed the posture. The set of five postures was repeated 10 times for each subject. In order to obtain only posture data without samples that correspond to transitions between postures, just inner 23 seconds from each posture were taken. In total, 125 minutes of data were taken. The experiments were acquired in the smart home lab in CIIRC, so they are not from real hospital environment. The postures were - Out of bed (Figure 5.1a), Supine (Figure 5.1b), Left log (Figure 5.1c), Right log (Figure 5.1d) and Sitting (Figure 5.1e).

Posture	**Time**
Out of bed	30s
Supine	30s
Left Log	30s
Right Log	30s
Sitting	30s

Table 5.1: Set of postures

(a) Out of bed

(b) Supine

(c) Left log

(d) Right log

(e) Sitting

Figure 5.1: Postures

Intentionally, subjects of different weights, body constitutions and sex were chosen. It is intended for the classifier to be as universal as it can be, which can be supported with high variance in the training dataset. The weight of each subject is described in Table 5.2.

Subject	Weight
S1	50 kg
S2	59 kg
S3	66 kg
S4	75 kg
S5	81 kg

Table 5.2: Weight of subjects

5.2 Posture classification algorithm

The Chapter 2 reported that variety of algorithms can be used for posture detection. In this work, 5 classification methods were compared (Decision trees, 1-Nearest Neighbor, 3-Nearest Neighbor, Naive Bayes and Linear Bayes). These algorithms were selected because of their simplicity and illustrative nature. It was not desirable to use complicated algorithms due to the limited amount of data. Moreover, the use of two different implementations - PRTOOLS and SMLT toolboxes in Matlab - helped to check the validity of results. The features that were used as inputs for the classifiers were data from sensor n. 142, data from sensor n. 143, data from sensor n. 144, data from sensor n. 145, x-position of the centre of mass (CMX) and y-position of the centre of mass (CMY).

In the following section only results for Decision trees, 1-Nearest Neighbor, 3-Nearest Neighbors and Naive Bayes from SMLT toolbox are chosen to be described, since they are the most demonstrative ones. Remaining results are summarized in Appendix B.

5.3 Results

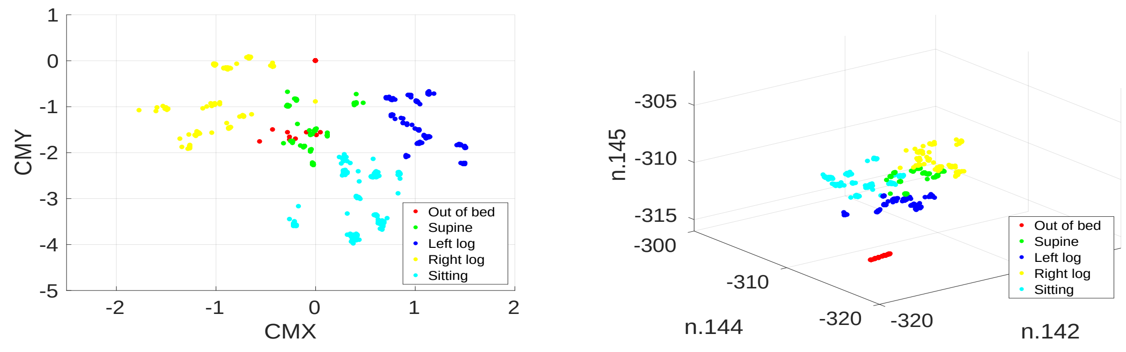
Following sections describes the results of both inter- and intra- personal classification. For intra-personal classification, i.e. testing and training on one subject, Hold-out error estimation method described in 4.2.1 is used. The results can be found further in section 5.3.2. The error estimation method divides dataset D of one subject into two disjunctive subsets of the same size, trains classifier on one subset and tests it on the other one. For inter-personal classification, i.e. testing and training on different subjects, n-fold cross-validation error estimation method described in section 4.2.2 is used. The results are described in section 5.3.3. In our case with five subjects, it divides dataset D into five parts $D_{i=1..5}$, where each part contains solely data from one subject. For each ith subject the classifier is trained on $D \setminus D_i$ and then tested on D_i.

5.3.1 Scatter plots

Since it is not possible to plot all features from the dataset at once, the data can be visualized using scatter plots - point graphs of dependence between features. The scatter plot of the position of centre of mass and 3D scatter plot of sensor n. 142, sensor n. 144 and sensor n. 145 were created. These three sensors were chosen as they have smallest correlation. Only scatter plots for two subjects were made to obtain better insight. The two subjects were chosen because of their illustrative and different nature.

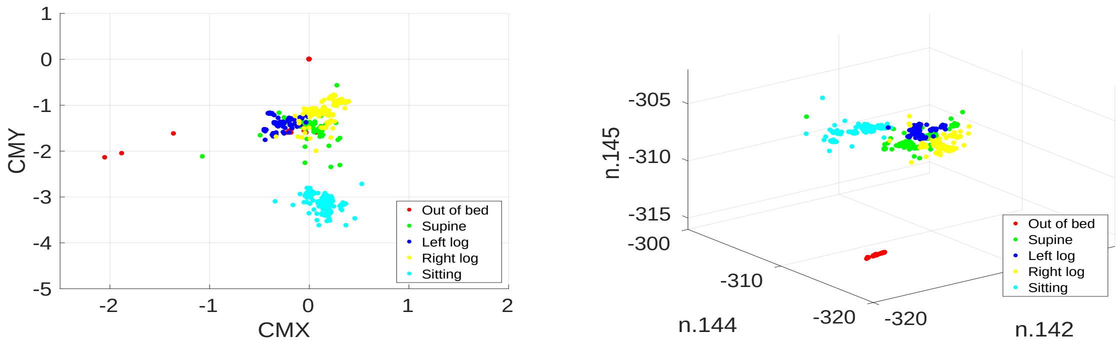
Subject **S1** has gracile body type (the smallest weight and physique among all subjects). This causes that its centre of mass has wider range of movement and data from its postures (classes) have consequently higher spread than postures of the other subjects. This phenomenon can be seen in Figure 5.2. On the other hand, centre of mass of subject **S4** does not vary so much. Although the subject **S4** is not the heaviest among all subjects, it has a rather robust body type and the movement of its center of mass is much more limited, which can be seen in Figure 5.3.

The same scatter plots for all subjects in one are depicted in Figure 5.4. the "Sitting" posture is quite separated from the other postures. Because the majority of subjects has their centre of mass in the middle, the "Out of bed" posture can be separated from tensometric data easily as can be seen in Figure 5.4b. Conversely, remaining postures seem to be less separable.



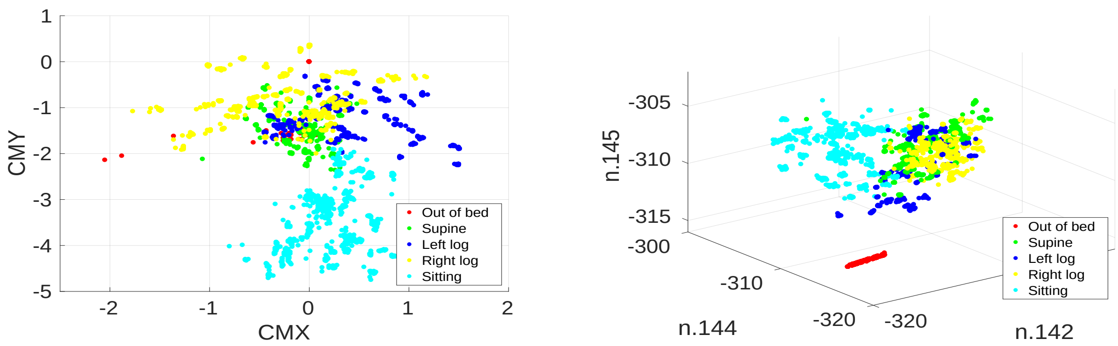
(a) Scatter plot of centre of mass (b) 3D scatter plot of three sensors

Figure 5.2: Scatter plots for Subject 1



(a) Scatter plot of centre of mass (b) 3D scatter plot of three sensors

Figure 5.3: Scatter plots for Subject 4



(a) Scatter plot of centre of mass (b) 3D scatter plot of three sensors

Figure 5.4: Scatter plots for multiple subjects

5.3.2 Intra-personal classifier

This section describes experiments with classifiers trained and tested on the same subject. It is expected, that this setting will reach higher accuracy than the inter-personal setting. Its disadvantage is that they are not universal and can be used only for the subject on which the classifier had been trained. It means that such system must be trained before its application to a new patient.

The results are summarized in Tables 5.3- 5.6, where one can find the marginal errors for particular subjects and classes. The last column describes averages computed over all classes (i.e. average errors on each subject) and the last row contains averages computed over all subjects (i.e. average errors on each class). Right bottom element of the tables represents the overall error for particular classifier.

The first interesting point is the performance of the NB classifier summarized in Table Table 5.3. It has been mentioned above that a violation of the assumption of feature independence can be a problem for NB. Moreover, it has been discussed in section 3.1.3 that some features are strongly dependent. Despite those facts, the best result was obtained for NB classifier with an average error 8.1%. This may be caused by relatively high independence of x and y position of centre of mass.

Slightly higher average errors were obtained for both kNN classifiers (see Table 5.4 and Table 5.5). The biggest average error was made by DT as depicted in Table Table 5.6. This is not surprising since DT generally tends to overfit, based on its greedy characteristic. The high risk of overfitting is implied by lack of acquired data (for each subject, only 10 samples from each class were obtained).

As could be expected, classifiers did not have a problem with classifying "Out of bed" and "Sitting" posture. As can be seen in the confusion matrix for NB - Table 5.7, "Out of bed" posture was wrong in only 1%. NB classified correctly all samples of the "Sitting" posture, whereas DT classifier 78.7% on average.

For all classifiers, the most confused posture was the "Right log". Neither kNN was not able to detect any "Right log" posture of **S1**. That is in contrast to **S3**, where both kNNs and NB were able to classify "Right log" posture with 0.0% error. This might be caused by previous knowledge of **S3** about the sensors in bed.The "Right log" posture was mostly confused with the "Supine" posture as can be seen in Table 5.7.

	Out of bed	**Supine**	**Left log**	**Right log**	**Sitting**	**Average error**
S1	0.0	0.0	20.0	81.7	0.0	20.3
S2	2.5	3.3	0.0	30.0	0.0	7.2
S3	0.0	0.0	0.0	0.0	0.0	0.0
S4	0.8	0.8	20.0	20.8	0.0	8.5
S5	1.7	20.0	0.0	0.0	0.0	4.3
Average error	1.0	4.8	8.0	26.5	0.0	**8.1**

Table 5.3: Table of hold-out estimate of relative errors in percents on each class and subject for NB classifier implemented in SMLT

	Out of bed	Supine	Left log	Right log	Sitting	Average error
S1	0.0	0.0	34.2	100.0	0.0	26.8
S2	0.0	39.2	0.0	19.2	0.0	11.7
S3	0.0	10.0	0.0	0.0	0.0	2.0
S4	0.0	4.2	4.2	0.0	0.0	1.7
S5	0.0	10.8	0.0	0.0	8.3	3.8
Average error	0.0	12.8	7.7	23.8	1.7	**9.2**

Table 5.4: Table of hold-out estimate of relative errors in percents on each class and subject for kNN ($k = 1$) classifier implemented in SMLT

	Out of bed	Supine	Left log	Right log	Sitting	Average error
S1	0.0	0.0	32.5	100.0	0.0	26.5
S2	0.0	23.3	0.0	11.7	0.0	7.0
S3	0.0	5.0	0.0	0.0	0.0	1.0
S4	0.0	19.2	4.2	0.0	0.0	4.7
S5	0.0	10.8	0.0	0.0	42.5	10.7
Average error	0.0	11.7	7.3	22.3	8.5	**10.0**

Table 5.5: Table of hold-out estimate of relative errors in percents on each class and subject for kNN ($k = 3$) classifier implemented in SMLT

	Out of bed	Supine	Left log	Right log	Sitting	Average error
S1	0.0	40.8	65.0	80.0	20.0	41.2
S2	0.0	24.2	0.8	30.0	0.0	11.0
S3	0.0	0.0	1.7	80.8	25.8	21.7
S4	0.0	0.8	40.0	0.8	0.0	8.3
S5	1.7	20.0	0.0	0.8	60.8	16.7
Average error	0.3	17.2	21.5	38.5	21.3	**19.8**

Table 5.6: Table of hold-out estimate of relative errors in percents on each class and subject for DT classifier implemented in SMLT

	ESTIMATED LABELS				
TRUE LABELS	**Out of bed**	**Supine**	**Left log**	**Right log**	**Sitting**
Out of bed	99.0	0.5	0.0	0.3	0.2
Supine	0.0	95.2	2.2	0.2	2.5
Left log	0.0	0.0	92.0	4.0	4.0
Right log	0.0	17.8	8.7	73.5	0.0
Sitting	0.0	0.0	0.0	0.0	100.0

Table 5.7: Confusion matrix estimate for NB classifier implemented in SMLT

5.3.3 Inter-personal classifiers

As stated in Chapter 1, the main goal was to create a classifiers that would be able to detect postures of previously unseen subject. This section describes the subject-based cross-validation errors that estimate the errors in such scenario. In tables 5.8 to 5.11 is accuracy for each classifier. As expected, these classifiers reached lower accuracy than those described above in section 5.3.2. Their advantage is that they are more robust and general.

Again, the best overall performance was achieved by the NB classifier with an average error of 23.1% as depicted in Table 5.8. In correspondance to section 5.3.2, the lowest accuracy was reached by the DT classifier with the average error of 33.4% (see Table 5.11). This may be also caused by overfitting, as it is a significant practical difficulty of DT. In contrast to the intra-personal scenario, the kNN classifiers performance is much closer to DT classifier and are worse relatively to the NB classifier. This can be caused by much higher overlap of the class distributions. (see tables 5.9 and 5.10). Both kNN classifiers reached similar average error of approximately 30%.

As can be seen, the classification of "Out of bed" posture was again easy. The only error made on this class was made by NB on subject **S1** and **S4** which maybe caused by a noise in data. The second most recognizable posture was "Sitting". This is not surprising since the "Sitting" posture vary a lot from the other ones.

The most confused posture was the "Left log". That differs from intra-personal classification scenario, where the most confused posture was the "Right log". (Posture "Left log" was a biggest complication for SVM classifier, reaching up to 63.3% error (see appendix B). Only for the NB classifier which made only 33.8%, the "Left log" was not that difficult to categorize, but simultaneously the NB is the only classifier that has a bigger problem to classify the "Supine" posture. This is because the "Supine" posture data are very close to "Left log" and "Right log" data.

Another interesting fact observed from tables 5.8 to 5.11 is that the majority of classifiers categorized subject **S1** with smallest error. The biggest error for **S1** was made on class "Supine", which does not follow the trend described in previous paragraph. Also, only for S1, the majority of classifiers had difficulties to classify the "Sitting" posture.

The biggest average error was made by classifiers on subject **S4**. Not even one classifier was able to detect its "Left log" posture. The main cause of this phenomenon may be a particular way of laying on the left side of **S4** since for only one subject described in 5.3.2, classifiers had not problems with classifying the "Left log" posture. Another explanation for this might be subject's robust body type. It can be observed from scatter plot of subject 4 and scatter plot of all data, that subject S4 data for "Right log" and "Left log" postures differ from all the others. This corresponds to 100% error on class right log in Tables 5.8-5.11. This is an example of inter-personal variability which causes the inter-personal classification more difficult.

In Table 5.12 can be seen the confusion matrix for NB classifier. For NB, the most wrongly estimated position was the "Supine" and was confused with "Right log". Postures "Left log" and "Right log" were estimated by classifier with a similar error, though the "Left log" was mostly confused with the "Supine" posture and "Right log" with "Left log". "Sitting" and "Out of bed" posture was detected without any error almost every time.

Although less than 60% accuracy on some classes was obtained, it must be pointed out that in a real-world application, the classifier will not be used directly, but will be post-processed by a subsequent procedure that will confirm the posture change. For example, the post-processing rule can be defined as: "If last m samples were all classified as class C, change the current system output to class C". This majority rule can help to fluctuations of the system output caused by occasional and short-missclassifications and noise.

	Out of bed	**Supine**	**Left log**	**Right log**	**Sitting**	**Average error**
S1	0.4	30.0	10.0	0.0	0.4	8.2
S2	0.0	30.0	11.7	40.8	0.0	16.5
S3	0.0	50.0	20.0	0.0	0.0	14.0
S4	6.2	27.1	100.0	100.0	0.0	46.7
S5	0.0	80.0	27.5	31.2	11.2	30.0
Average error	1.3	43.4	33.8	34.4	2.3	**23.1**

Table 5.8: Table of cross-validation estimate of relative errors in percents on each class and subject for NB classifier implemented in SMLT

	Out of bed	**Supine**	**Left log**	**Right log**	**Sitting**	**Average error**
S1	0.0	30.4	7.9	0.4	18.8	11.5
S2	0.0	39.6	12.5	60.0	0.0	22.4
S3	0.0	80.0	49.2	36.3	0.0	33.1
S4	0.0	31.2	100.0	95.4	0.0	45.3
S5	0.0	57.9	100.0	16.7	10.4	37.0
Average error	0.0	47.8	53.9	41.8	5.8	**29.9**

Table 5.9: Table of cross-validation estimate of relative errors in percents on each class and subject for kNN ($k = 1$) classifier implemented in SMLT

	Out of bed	Supine	Left log	Right log	Sitting	Average error
S1	0.0	30.4	0.0	0.4	15.4	9.2
S2	0.0	39.6	12.5	59.6	0.0	22.3
S3	0.0	80.0	50.4	52.1	0.0	36.5
S4	0.0	32.1	100.0	99.2	0.4	46.3
S5	0.0	63.7	100.0	19.6	10.0	38.7
Average error	0.0	49.2	52.6	46.2	5.2	**30.6**

Table 5.10: Table of cross-validation estimate of relative errors in percents on each class and subject for kNN ($k = 3$) classifier implemented in SMLT

	Out of bed	Supine	Left log	Right log	Sitting	Average error
S1	0.0	69.6	52.9	49.6	7.9	36.0
S2	0.0	23.7	10.8	90.0	4.6	25.8
S3	0.0	70.0	45.4	0.0	0.4	23.2
S4	0.0	46.7	100.0	99.6	5.0	50.3
S5	0.0	47.1	100.0	0.4	12.1	31.9
Average error	0.0	51.4	61.8	47.9	6.0	**33.4**

Table 5.11: Table of cross-validation estimate of relative errors in percents on each class and subject for DT classifier implemented in SMLT

	ESTIMATED LABELS				
TRUE LABELS	**Out of bed**	**Supine**	**Left log**	**Right log**	**Sitting**
Out of bed	98.7	0.0	0.0	0.0	1.3
Supine	0.0	56.6	17.2	24.2	2.1
Left log	0.0	24.1	66.2	7.8	2.0
Right log	0.0	14.1	20.3	65.6	0.0
Sitting	0.0	0.0	2.3	0.0	97.7

Table 5.12: Confusion matrix for NB classifier implemented in SMLT

Chapter 6

Summary and conclusion

Firstly, this thesis familiarizes reader with the data obtained from the smart medical bed. In Chapter 3, a description of data is provided. Only the following relevant data were picked, tested and described - 4 signals obtained directly from four strain gages inserted into bed and x and y position of centre of mass. Next, in Chapter 4, algorithms and methods of machine learning used in this work are detailed. The main components are the classifiers and error estimation methods. Finally, in Chapter 5, the complete description of posture detection experiment is provided. Firstly, the acquisition of data is described. Then, the results for both one subject and multiple subjects are presented and probable cause of the outcomes are explained.

The most important outcomes are:

- Intra-personal classifiers can classify the data with average error 8-19%, but must be trained on each patient separately.
- Inter-personal classifiers can classify the data with average error 20-30%.
- There is an important impact of inter-personal variability if the inter-personal classification is considered. Any subject whose postures differ from the others, increases significantly the error.
- Although it is extremely easy to classify "Sitting" and "Out of bed" postures, the classification of the other postures is much more difficult.
- The real system must be accompanied by a post-processing procedure that will probably increase the final classification performance. This will be the part of a

future work.

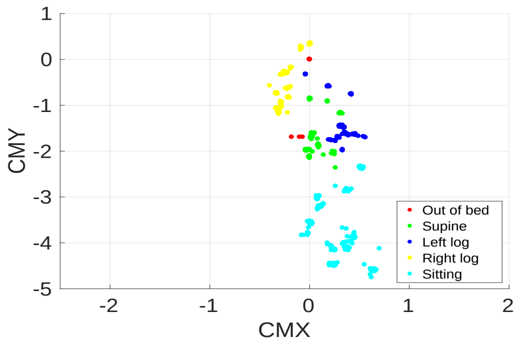
With respect to other state-of-the-art techniques, the results for multiple subjects are not as achieving as some of those described in Chapter 2. This can be due to our limited number of sensors, since every reported work had at least 16 sensors. Nevertheless, the results presented in this thesis are comparable with [6], where authors used Bayesian classification as well and achieved 78.7% average precision rate. Naive Bayes classifier presented in this thesis reached 76.9% precision rate.

Finally, it should be noted that all the goals of the thesis were fulfilled. First, the structure of data obtained from smart hospital bed LINET was explored and some demonstration experiments were presented in Chapter 3. Next, the posture detection was proposed as a possible way of utilization of bed data in a real environment. The posture detection system was implemented and evaluated. To accomplish all those objectives, many different data sets were measured in smart home laboratory of CIIRC in colaboration with several volunteers.

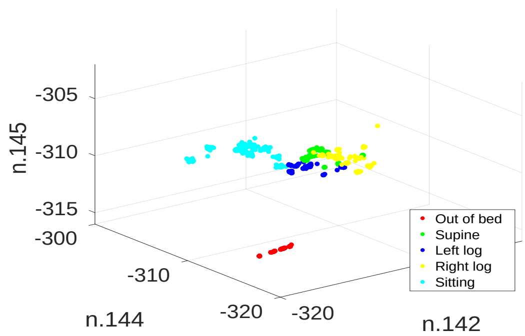
The work has many possible future extensions. The examples are real-time implementation of the posture detection, classifier output post-processing system deciding about the final state indication, extraction of different high level features for classification or a classifier ensemble system combining different types of differently performing classifiers.

Appendix A

Scatter plots

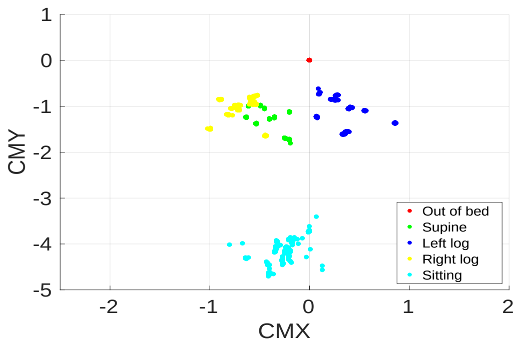


(a) Plot of centre of mass

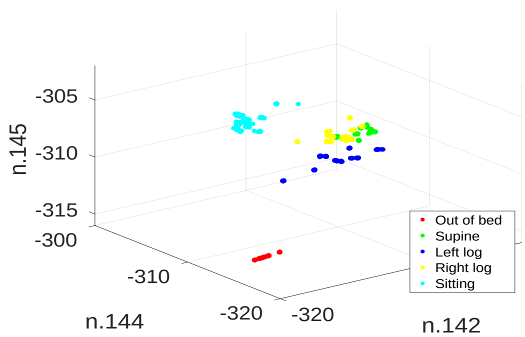


(b) 3D plot of three sensors

Figure A.1: Plots for Subject 2

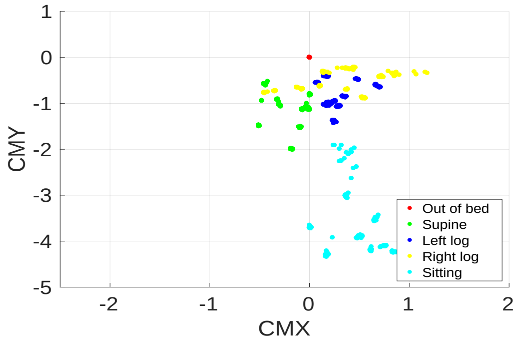


(a) Plot of centre of mass

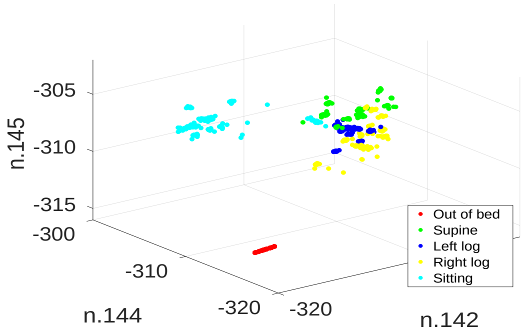


(b) 3D plot of three sensors

Figure A.2: Plots for Subject 3



(a) Plot of centre of mass



(b) 3D plot of three sensors

Figure A.3: Plots for Subject 5

Appendix B

Tables of results

B.1 Intra-personal classification results

	Out of bed	Supine	Left log	Right log	Sitting	Avg. error
S1	0.0	0.0	60.0	80.0	0.0	28.0
S2	4.2	25.0	21.7	89.2	29.2	33.9
S3	0.0	20.0	0.0	25.8	60.0	21.2
S4	0.8	23.3	55.8	0.0	7.5	17.5
S5	0.0	0.8	0.0	4.2	40.8	9.2
Avg. error	1.0	13.8	27.5	39.8	27.5	**22.0**

Table B.1: Table of hold-out estimate of relative errors in percents on each class and subject for DT classifier implemented in PRTOOLS

	Out of bed	Supine	Left log	Right log	Sitting	Avg. error
S1	0.0	0.0	40.8	100.0	0.0	28.2
S2	0.0	25.0	0.0	18.3	0.0	8.7
S3	0.0	23.3	0.0	14.2	0.0	7.5
S4	0.0	0.0	2.5	0.8	0.0	0.7
S5	0.0	1.7	0.0	0.0	20.8	4.5
Avg. error	0.0	10.0	8.7	26.7	4.2	**9.9**

Table B.2: Table of hold-out estimate of relative errors in percents on each class and subject for kNN ($k = 1$) classifier implemented in PRTOOLS

	Out of bed	**Supine**	**Left log**	**Right log**	**Sitting**	**Avg. error**
S1	0.0	0.0	39.2	100.0	0.0	27.8
S2	0.0	23.3	0.0	20.8	0.8	9.0
S3	0.0	20.0	0.0	19.2	0.0	7.8
S4	0.0	20.0	2.5	0.8	0.0	4.7
S5	0.0	0.0	0.0	0.0	24.2	4.8
Avg. error	0.0	12.7	8.3	28.2	5.0	**10.8**

Table B.3: Table of hold-out estimate of relative errors in percents on each class and subject for kNN ($k = 3$) classifier implemented in PRTOOLS

	Out of bed	**Supine**	**Left log**	**Right log**	**Sitting**	**Avg. error**
S1	0.0%	0.0	30.0	80.0	0.0	22.0
S2	0.0	22.5	0.0	25.8	4.2	10.5
S3	0.0	5.8	0.0	25.8	0.8	6.5
S4	0.0	20.8	20.0	0.8	18.3	12.0
S5	0.0	20.0	0.0	0.0	1.7	4.3
Avg. error	0.0	13.8	10.0	26.5	5.0	**11.1**

Table B.4: Table of hold-out estimate of relative errors in percents on each class and subject for NB classifier implemented in PRTOOLS

	Out of bed	**Supine**	**Left log**	**Right log**	**Sitting**	**Avg. error**
S1	0.0	0.0	20.0	80.8	0.0	20.2
S2	0.0	23.3	0.0	24.2	0.0	9.5
S3	0.0	0.0	0.0	0.0	0.0	0.0
S4	0.0	0.0	22.5	38.3	0.0	12.2
S5	0.0	0.0	0.0	0.0	53.3	10.7
Avg. error	0.0	4.7	8.5	28.7	10.7	**10.5**

Table B.5: Table of hold-out estimate of relative errors in percents on each class and subject for LB classifier implemented in PRTOOLS

	Out of bed	Supine	Left log	Right log	Sitting	Avg. error
S1	0.0	2.5	26.7	89.2	0.0	23.7
S2	0.0	25.8	0.0	24.2	1.7	10.3
S3	0.0	0.0	0.0	0.0	0.0	0.0
S4	0.0	0.0	20.0	5.0	0.0	5.0
S5	0.0	0.8	0.0	0.0	40.8	8.3
Avg. error	0.0	5.8	9.3	23.7	8.5	**9.5**

Table B.6: Table of hold-out estimate of relative errors in percents on each class and subject for SVM classifier implemented in SMLT

B.2 Inter-personal classification results

	Out of bed	Supine	Left log	Right log	Sitting	Avg. error
S1	0.0	30.4	75.0	49.2	19.6	34.8
S2	0.0	51.2	37.9	60.4	0.4	30.0
S3	0.0	100.0	50.0	10.0	0.0	32.0
S4	2.5	52.9	100.0	85.0	0.0	48.1
S5	0.0	40.4	99.2	19.2	10.0	33.8
Avg. error	0.5	55.0	72.4	44.8	6.0	**35.7**

Table B.7: Table of cross-validation estimate of relative errors in percents on each class and subject for DT classifier implemented in PRTOOLS

	Out of bed	Supine	Left log	Right log	Sitting	Avg. error
S1	0.0	30.4	10.0	0.0	19.6	12.0
S2	0.0	65.4	21.3	59.2	0.0	29.2
S3	0.0	70.0	56.2	45.8	0.0	34.4
S4	0.0	34.6	100.0	89.6	0.8	45.0
S5	0.0	41.2	99.6	20.0	8.8	33.9
Avg. error	0.0	48.3	57.4	42.9	5.8	**30.9**

Table B.8: Table of cross-validation estimate of relative errors in percents on each class and subject for kNN ($k = 1$) classifier implemented in PRTOOLS

	Out of bed	Supine	Left log	Right log	Sitting	Avg. error
S1	0.0	30.4	10.0	0.0	19.6	12.0
S2	0.0	64.6	21.3	51.2	0.0	27.4
S3	0.0	70.0	59.2	32.1	0.0	32.3
S4	0.0	35.0	100.0	98.3	0.8	46.8
S5	0.0	67.5	93.8	20.0	8.8	38.0
Avg. error	0.0	53.5	56.9	40.3	5.8	**31.3**

Table B.9: Table of cross-validation estimate of relative errors in percents on each class and subject for kNN ($k = 3$) classifier implemented in PRTOOLS

	Out of bed	Supine	Left log	Right log	Sitting	Avg. error
S1	0.0	40.4	62.1	10.4	5.4	23.7
S2	0.0	29.6	11.7	60.4	0.0	20.3
S3	0.0	50.0	51.2	0.0	0.0	20.2
S4	0.0	33.8	100.0	99.2	0.8	46.8
S5	0.0	70.4	76.7	51.2	10.8	41.8
Avg. error	0.0	44.8	60.3	44.2	3.4	**30.6**

Table B.10: Table of cross-validation estimate of relative errors in percents on each class and subject for NB classifier implemented in PRTOOLS

	Out of bed	Supine	Left log	Right log	Sitting	Avg. error
S1	0.0	50.4	0.0	0.4	40.4	18.2
S2	1.2	30.0	20.0	15.8	10.0	15.4
S3	0.0	0.0	60.4	32.5	0.0	18.6
S4	6.2	57.1	100.0	90.0	0.0	50.7
S5	0.0	50.0	20.0	40.8	10.4	24.2
Avg. error	1.5	37.5	40.1	35.9	12.2	**25.4**

Table B.11: Table of cross-validation estimate of relative errors in percents on each class and subject for LB classifier implemented in PRTOOLS

	Out of bed	**Supine**	**Left log**	**Right log**	**Sitting**	**Avg. error**
S1	0.0	40.4	14.6	9.6	20.4	17.0
S2	0.0	28.7	55.8	22.9	0.0	21.5
S3	0.0	30.0	60.4	30.4	0.0	24.2
S4	0.0	25.0	100.0	95.0	0.0	44.0
S5	0.0	90.4	87.5	0.4	10.0	37.7
Avg. error	0.0	42.9	63.7	31.7	6.1	**28.9**

Table B.12: Table of cross-validation estimate of relative errors in percents on each class and subject for SVM classifier implemented in SMLT

Bibliography

[1] [Online]. Available: https://ec.europa.eu/health/workforce/overview_en.

[2] R Yousefi, S Ostadabbas, M Faezipour, M Farshbaf, M Nourani, L Tamil, and M Pompeo, "Bed posture classification for pressure ulcer prevention", in *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*, IEEE, 2011, pp. 7175–7178.

[3] H. M. Van Der Loos, N. Ullrich, and H. Kobayashi, "Development of sensate and robotic bed technologies for vital signs monitoring and sleep quality improvement", *Autonomous Robots*, vol. 15, no. 1, pp. 67–79, 2003.

[4] H. J. Lee, S. H. Hwang, S. M. Lee, Y. G. Lim, and K. S. Park, "Estimation of body postures on bed using unconstrained ecg measurements", *IEEE journal of biomedical and health informatics*, vol. 17, no. 6, pp. 985–993, 2013.

[5] Y.-W. Hung, Y.-H. Chiu, Y.-C. Jou, W.-H. Chen, and K.-S. Cheng, "Bed posture classification based on artificial neural network using fuzzy c-means and latent semantic analysis", *Journal of the Chinese Institute of Engineers*, vol. 38, no. 4, pp. 415–425, 2015. DOI: 10.1080/02533839.2014.981212. eprint: https://doi.org/10.1080/02533839.2014.981212. [Online]. Available: https://doi.org/10.1080/02533839.2014.98121y.

[6] C.-C. Hsia, Y.-W. Hung, Y.-H. Chiu, and C.-H. Kang, "Bayesian classification for bed posture detection based on kurtosis and skewness estimation", in *E-health Networking, Applications and Services, 2008. HealthCom 2008. 10th International Conference on*, IEEE, 2008, pp. 165–168.

[7] C. C. Hsia, K. J. Liou, A. P. W. Aung, V. Foo, W. Huang, and J. Biswas, "Analysis and comparison of sleeping posture classification methods using pressure sensitive bed system", in *2009 Annual International Conference of the IEEE Engineering in*

Medicine and Biology Society, 2009, pp. 6131–6134. DOI: `10.1109/IEMBS.2009.5334694`.

[8] W. Viriyavit, V. Sornlertlamvanich, W. Kongprawechnon, P. Pongpaibool, and T. Isshiki, "Neural network based bed posture classification enhanced by bayesian approach", in *2017 8th International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES)*, 2017, pp. 1–5. DOI: `10.1109/ICTEmSys.2017.7958769`.

[9] N. Foubert, A. M. McKee, R. A. Goubran, and F. Knoefel, "Lying and sitting posture recognition and transition detection using a pressure sensor array", in *2012 IEEE International Symposium on Medical Measurements and Applications Proceedings*, 2012, pp. 1–6. DOI: `10.1109/MeMeA.2012.6226630`.

[10] L. Russell, R. Goubran, and F. Kwamena, "Posture detection using sounds and temperature: Lms-based approach to enable sensory substitution", *IEEE Transactions on Instrumentation and Measurement*, pp. 1–12, 2018, ISSN: 0018-9456. DOI: `10.1109/TIM.2018.2795158`.

[11] R. Hannah, S. Reed, and S. for Experimental Mechanics (U.S.), *Strain gage users' handbook*. Elsevier Applied Science, 1992, ISBN: 9780912053363. [Online]. Available: `https://books.google.cz/books?id=ig64QgAACAAJ`.

[12] I. Rish, "An empirical study of the naïve bayes classifier", vol. 3, Jan. 2001.

[13] J. Gama, "A linear-bayes classifier", in *Advances in Artificial Intelligence*, M. C. Monard and J. S. Sichman, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 269–279, ISBN: 978-3-540-44399-5.