

# Human Activity Recognition With Smartphones

Jeremy Sapienza, Simone Fiorellino, Stefano D'Arrigo

December 27, 2020

## Abstract

## 1 Introduction

Nowadays everyone has a smartphone in his pocket and carries it everywhere he goes during the day journey. Smartphones collect each time a huge amount of data through their embedded sensors. High tech industries are involved into the development of accurate wearable devices to improve every day life, monitoring the activities of their customers, from walking in a park to laying on the bed. Hence, we reckoned this topic valuable and in our project we focused on the classification task of human activities, making use of accelerometer and gyroscope records. Our attention was particularly concentrated on the training of two classifiers, GDA and SVM, and on the research of a convenient data representation. In this report we summarize the methods we applied and the choices we made. Finally, in the last section we present our findings and give some further analyses.

## 2 Related work

For this work we were inspired by the paper [?]. The research focuses on developing a classifier that operates on accelerometer and gyroscope data from mobile phones. As explained in [?], they recorded the stream of data of 30 subjects during different types of activity: walking, walking up stairs, walking down stairs, sitting, standing and laying.

The dataset have two types of data:

- “Raw data”: raw gyroscope and accelerometer readings.
- “Preprocessed data”: Vectors of 561 features that represent 2.56s of time. These features represent different functions, some examples are: triaxial average, maximum and minimum acceleration, angular velocity (over the given interval) or Fourier transform.

They used for this part of work the preprocessed data.

To approach this problem they used three difference models:

- Naives Bayes
- GDA

- GDA + Hidden Markov Model

The GDA accuracy is much higher than the Naive Bayes model because this last model makes the assumption that the features are independent of each other. But, in this context, having the features more correlated together, the accuracy with the Naive Bayes model is pretty low.

Another important aspect of this paper is the dimensionality reduction. They used the PCA in order to decrease the computational complexity of the model and improve the accuracy.

## 3 Methods

### 3.1 Data preprocessing

We retained 60% of the data for the models' training; the remaining part was equally splitted into evaluation and test sets.

### 3.2 Models

As a first step, we implemented the Gaussian Discriminant Analysis classifier and trained it on the data; the choice of this model was mainly conditioned by [?]. For the implementation, we faced out with the problem of multiclass classification, as in [?]. With the purpose of improving the performance, we took into account a step of feature selection and dimensionality reduction: indeed, the high number of derived features forced us to focus on the relationship between the features of the original dataset, the variance of them and the predicted classes. So, we considered two different strategies:

- feature selection with the Analysis of Variance (ANOVA) model;
- dimensionality reduction with the Singular Value Decomposition (SVD) method.

The second step was training the Support Vector Machine classifier on the data, following the same approach just described. After reviewing [?, ?, ?] and having tried to implement it as in [?, ?], we decided to use the implementation given by the library "Scikit Learn" [?, ?]. In this case, it was essential finding a good configuration of feature space and model hyper-parameters. Since this research would have taken too much to be conducted on each possible combination of hyper-parameters, we chose a set of them which we reckoned reasonable.

The results of the training and fine tuning step will be presented and commented on in the next section.

## 4 Experimental results

## 5 Conclusion and future work

## References

- [1] T. D. Matt Brown and L. O'Connor, "Activity classification with smartphone data," 2013.

- [2] J. Platt, “Sequential minimal optimization: A fast algorithm for training support vector machines,” 1998.
- [3] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [4] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, “API design for machine learning software: experiences from the scikit-learn project,” in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pp. 108–122, 2013.