Gated recurrent units for activity recognition

Deep neural networks application on self collected dataset

Andrea Mascari - Davide Pivato

Padua March 7th 2023

TABLE OF CONTENTS

The Topic: Activity Recognition

ECG Signals

Heart rate monitor to record electrical activity of the heart in ECG signals

VS

Rest vs Walk

Binary classification task between two main activities

The Dataset

Hand collected data

- Detection time of about 1m
- Seven subjects
- Age and sex independent

The Goal of Generalization

QRS Complexes

01

We were interested on catching peculiarities of the heartbeat's shape

Train and Test sets

02

Five subjects in train

Two subjects in test

Extend Knowledge

03

Activity recognition ability should hold when predicting unseen subjects

Data pre-processing

Segmentation

Single-beat sequence extraction from the signal, using QRS detection

Normalization

Standard scaling to reduce the range and the variance of values.

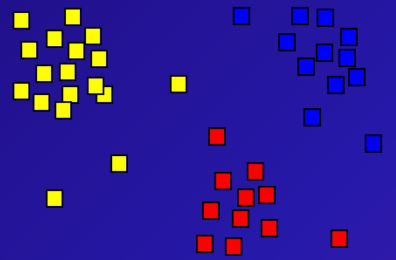
Data augmentation

SMOTE technique to increase the samples of the minority class

Key Features

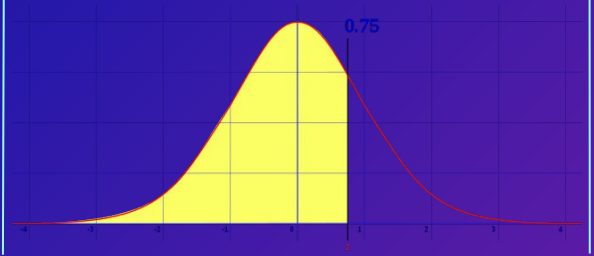
SMOTE

Synthetic generated data from minority class



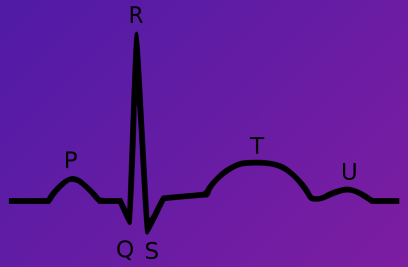
STANDARD SCALING

Performed by subtracting the mean and dividing by standard deviation



QRS DETECTION

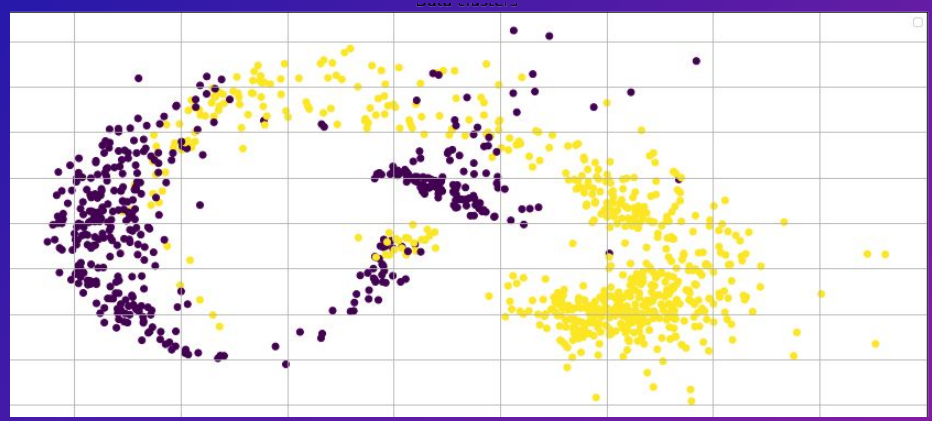
Detection of the QRS complex in the signals



Exploratory Data Analysis

Normalized ECG values Histogram

PCA Visualization



Applications Timeline

Convolutional neural network

To extract local dependencies

01

02

GRU units

Injecting memory in the process

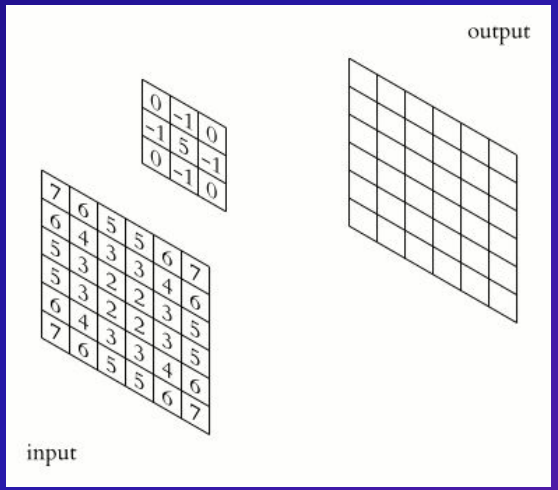
Autoencoders

As weights initialization for the network

03

Convolutional neural network

Suitable architecture to process images

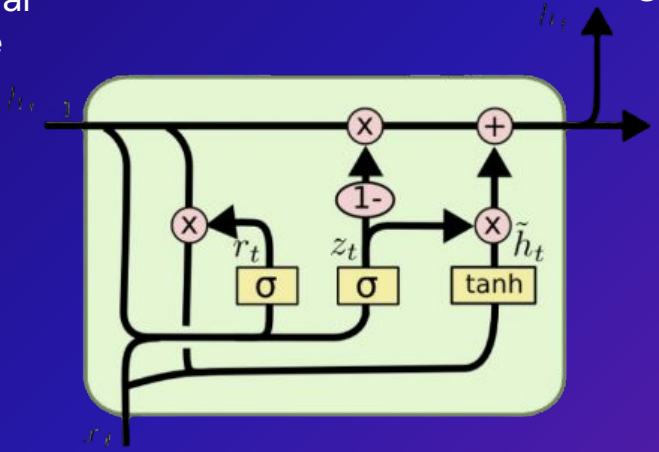


Proposed a 1d-convolutional network to process the signal inputs

Implemented to compare the performance with the recurrent classifier

GRU cells

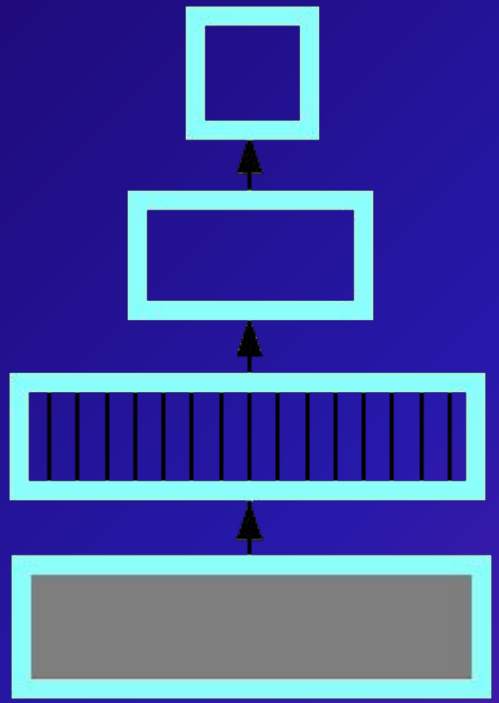
Type of recurrent neural network architecture



Capture long time in sequential data

have fewer parameters than LSTM and are therefore faster to train and require less memory to store

Recurrent network classifier



Output Dense Layer (1) - sigmoid

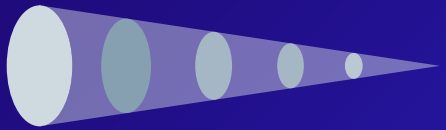
Gru Layer (8) - tanh

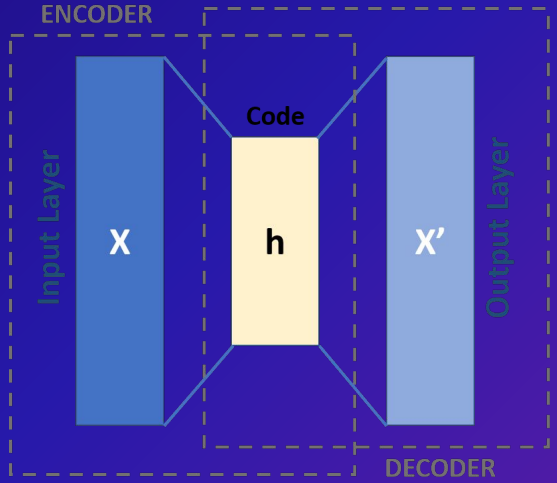
Starting Dense Layer (16) - relu

Input (100,)

Autoencoder

Compressing input representation in a more dense one





Idea of injecting preliminary representation in the classifier

Initialize the dense layer weights with the weights learned by the undercomplete autencoder

RESULTS

	Accuracy	Std	Precision	Recall	F1-score
Dense + GRU	90.0%	2.1%	97.2%	83.0%	89.5
AE + Classifier	87.4%	4.2%	97.2%	84.3%	90.3
DeepCNN	74.2%	0.7%	82.1%	77.0%	79.5

THANK YOU FOR YOUR ATTENTION