QRS complex classification (normal, abnormal) using a norm of linear algebra or correlation coefficient

Aljaz Mur Erzen[1]

[1] *ae4664@student.uni-lj.si, 63160011*

Introduction

Automatic heartbeat classification plays a key role in analyzing streams of recorded ECG signals, finding abnormalities and predicting possible heart diseases. Due to significance of the implications, there is a lot of motivation to improve performance of existing classification algorithms.

This report describes how we implemented a simple classification algorithm based on beat comparison to a normal heartbeat using different algebraic norms and a correlation coefficient. The algorithm requires recorded ECG signal and detected R-peaks or QRS complex.

Baseline drift suppression

Due to baseline drift, which is caused by external factors during ECG recording, the signals of the complexes may contain different low frequency signals. This problem can be solved using different approaches:

High-pass filters could suppress the baseline drift, but may also introduce non-linear phase shift, which would distort the QRS complex, lowering probability it would be classified correctly. To remove this phase shift, we could apply the filter *in reverse direction*, but this solution can be done only on pre-recorded signals and not in real-time.

Isoelectric level offset If we assume that the baseline drift is constant over one heartbeat, we can align the complexes by simple subtraction. The offset, which is called isoelectric level of the QRS complex, can be estimated by averaging a few samples before the R-peak or using some other more sophisticated algorithm [1].

Our algorithm uses two variations of the second approach:

- isoelectric level estimation which uses sample average of points 72ms, 68ms, 64ms and 60ms before R-peak as the offset.
- normalization using the whole signal average and scaling to normalize standard divination.

Computing the normal heartbeat

To compare the heartbeats, we need a notion of a *normal* heartbeat. The simplest approach is to take first N detected heartbeats and compute their average. This approach may produce bad results if some or all of these beats turn out not to be classified as normal. But in most cases, when the first N QRS complexes are normal, the produced mean signal should be similar to our notion of a normal QRS complex.

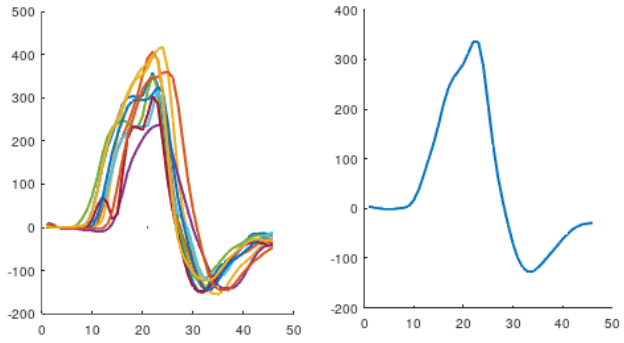

Figure 1. Overlaid and averaged QRS complexes 10 ECG signals of the QRS complexes were aligned to match their isoelectric levels. PR segments are therefore well aligned, but R and S peaks are not.

For our implementation, we used 500 first heartbeats, which is about 5 minutes of the signal.

ECG signal distance

Now we can compare each detected heartbeat to the *normal* heartbeat. This is done by computing a *distance* measure between the two and then applying a threshold.

We explored four different measures, namely ℓ_1 norm, Euclidean distance (or ℓ_2 norm), maximum norm (or ℓ_∞ norm) and intra-class correlation coefficient. To obtain best results, the measure and the threshold should be chosen carefully.

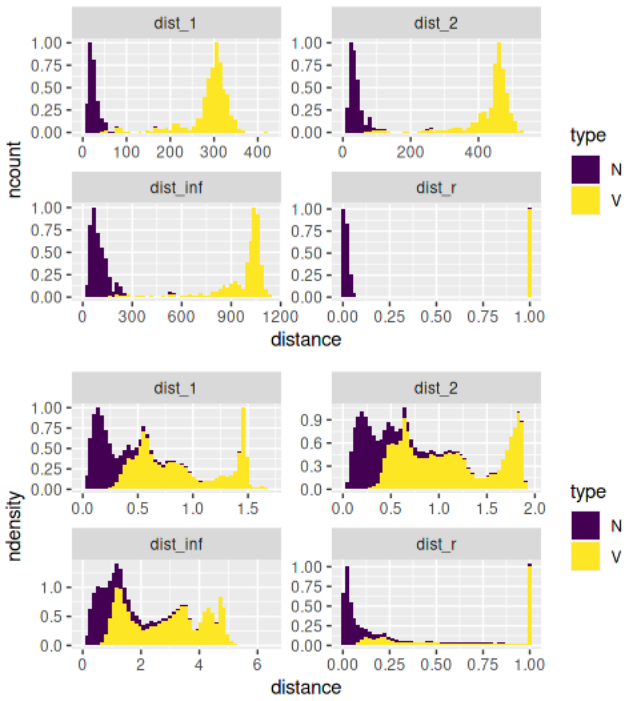

Figure 2. Comparison of different distance measures Plots show distributions of distances to mean heartbeat using different measures. Note that densities are scaled individually for normal (N) and PVC (V) complexes. Upper plot represents record s20501, lower represents the whole LTST database.

To choose the most appropriate threshold, we can plot ROC curve (Receiver Operating Characteristic), which shows different thresholds in FPR-TPR space.

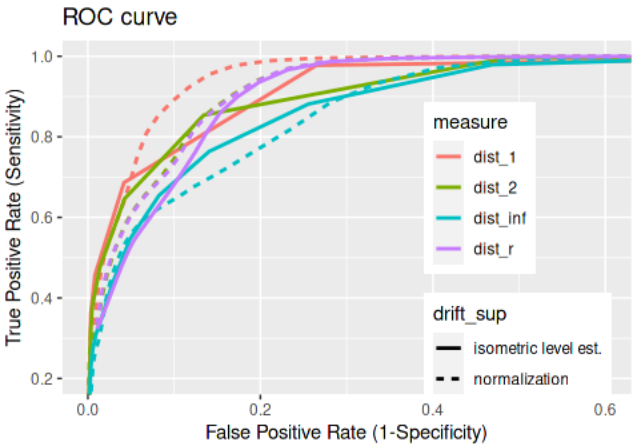

Figure 3. ROC curve Overall, ℓ_1 norm yields best results. Drift suppression should be chosen based on desired ratio between FPR and TPR.

Implementation and testing

Algorithm was implemented in Matlab programming language and used Bash scripts and Physionet's WFDB toolkit [2] for preparing, converting input data and evaluating classifier output. The algorithm was developed and tested on data from LTST database [3] containing 86 records each containing 2 or 3 ECG signals with mean duration 23.1h (std = 3.03h).

On a desktop computer with processor Processor AMD Ryzen 5 1600 Six-Core and Matlab version 9.9.0, mean running time of the classifier analyzing one of the records was 1.53s (std = 0.25s), but real running time (including Matlab startup and annotation reading) was 16.81s (std = 3.48s).

Results

With selected ℓ_1 norm, signal normalization for drift suppression and threshold of 0.43, the algorithm achieved following results: Negative prediction is associated with normal heartbeat (N notation in LTST) and positive prediction is associated with PVC (V notation in LTST).

Beats	8708579
FN	6786
FP	933422
Sensitivity (Se)	90.65%
Precision (P+)	6.58%
Specificity (Sp)	89.19%

Due to low ratio between positives and negatives, the precision (or positive predictively P+) is low, even though sensitivity and specificity are relativity high.

Considering that *Se* and *Sp* metrics can be chosen arbitrarily from the ROC curve by adjusting the threshold, we could adapt the algorithm to the use case, but combined metrics are still too low for the algorithm to be considered reliable.

One possible improvement is further alignment of the QRS signal; from the figures above we can see that the R and S-peaks are not well aligned, which may be leading cause for high FNR.

References

[1] Ales Smrdel and Franc Jager. An algorithm to estimate the ST segment level in 24-hour ambulatory ECG records. volume 35, pages 701 – 704, 10 2008.

[2] WFDB toolkit. https://archive.physionet.org/physiotools/wfdb.shtml.

[3] Franc Jager, Alessandro Taddei, George B. Moody, Michele Emdin, Gorazd Antolic, Roman Dorn, Ales Smrdel, Carlo Marchesi, and Roger G. Mark. The Long-Term ST Database (LTST DB). *Medical & Biological Engineering & Computing*, 412(2):172–183, 2003.