Electrocardiography Denoising via Sparse Dictionary Learning from Small Datasets

T. Steinbrinker and N. Spicher

***Abstract*—We propose and evaluate an algorithm for electrocardiography denoising via sparse dictionary learning, targeting two types of noise: baseline wander and muscle artifacts. Using MIT-BIH Noise Stress Database, for each type of noise a dictionary is built using K-singular value decomposition. This iterative method alternates between finding a sparse representation for every training signal and then updating every atom of the dictionary on its own. A spare representation is found using the using the orthogonal matching pursuit algorithm. The atoms are updated exploiting the properties of the singular value decomposition. Electrocardiography data stems from synthetic signals as well as the freely-available Brno University of Technology ECG Quality Database. During testing, we use the basis pursuit denoising algorithm. Our results regarding baseline wander demonstrate that the algorithm outperforms the American Heart Association-recommended bandpass filter w.r.t. signal-to-noise ratio on simulated signals. Moreover, a small number of training data is sufficient for satisfying results which indicates the suitability of the method for wearable hardware with low memory and power specifications.**

***Index Terms*—Electrocardiography, Filtering, Noise**

I. Introduction

Electrocardiography (ECG) is a standard method for diagnosis, monitoring, and risk assessment of cardiovascular diseases, such as heart failure. Recently, there is a trend towards wearable devices [1]. In comparison to standard 12-lead ECG, these devices are more unobtrusive but also more vulnerable to artifacts. Typical artifacts are baseline wander (BW) and muscle artifacts (MA). Removing these artifacts is important for downstream tasks and therefore it is important to remove only the artifacts and not the signal components [2]. Moreover, algorithms have to be adjusted when they shall be run on wearable devices as these have only limited energy, storage, and computing capacities.

In this work, we propose and evaluate an algorithm for ECG denoising via sparse dictionary learning with the requirement of small training data. We perform a quantitative analysis of its performance on BW and a qualitative analysis for MA.

II. Material and Methods

A. ECG data

1) We use the `ecg_simulate()` function of `Neurokit2` [3] to generate synthetic ECG data based on the ECGSYN model [4].

2) The Brno University of Technology ECG Quality Database (BUT ECG DB) comprises 18 long-term recordings of a single-lead ECG acquired from 15 subjects (9 female) aged between 21-83 years [5] The recordings are labelled based on the ECG signal quality, with annotation classes 1, 2, and 3. For training and testing purposes, 10s windows marked as class 1 are utilized.

3) MA and BW data is obtained from the MIT-BIH Noise Stress Database [6]. The database consist of real-life noise, for both types of noise there are two channels about half an our in length. 10s windows are added to the clean data to obtain the noisy data.

Regarding training and test split, we use four recordings ("100001", "100002", "103001", "103002") from BUT DB for training, and the 14 remaining for testing.

B. Algorithms

We build different dictionaries for both types of noise. To train these dictionaries, we use the K-SVD and the orthogonal matching pursuit (OMP) algorithm. During testing, to sparsely approximate a testsignal, given a dictionary, we use the basis pursuit denoising (BPDN) algorithm.

1) Orthogonal Matching Pursuit: Matching pursuit (MP) is a simple greedy algorithm [7] to find an approximate solution to the problem

$$\min_x \|y_i - \mathbf{\Psi} x_i\|_2^2 \qquad \text{subject to} \qquad \|x_i\|_0 \le T_0, \quad \forall i. \tag{1}$$

Here, $\|\cdot\|_0$ denotes the number of non-zero elements in a vector. Finding the global solution is NP hard, so different algorithms exist to find an approximation. MP aims at a approximation of the form $f \approx f_N := \sum_{n=1}^{N} a_n g_{\gamma_n}$, where g_{γ_n} is the γ_nth column of the matrix D and a_n is the scalar weighting factor. Assuming, that the columns of the dictionary are normalized, the "biggest contribution" is given by the scalar product. Therefore, we compute $g_{\gamma_n} \in \mathbf{\Psi}$ with maximum inner product $a_n = |\langle R_n, g_{\gamma_n}\rangle|$. Then we find the residual $R_{n+1} = R_n - a_n g_{\gamma_n}$ and repeat this step, until the sparsity constraint is reached. This algorithm happens to run into local minima very quickly.

The OMP algorithm is a variation of the MP algorithm. It allows to change the coefficients found in the previous step by computing the least square solution $\arg\min_x \|\boldsymbol{A}_{\Lambda_k} x - b\|_2^2$, where Λ_k is the set of atoms that we chose until that iteration. This reduces the risk to fall into local minima.

2) K-SVD: The K-singular value decomposition (SVD) algorithm is used to train a dictionary [8]. Assume that a set of training signals $\{y_i\}_{i=1}^N$ is given and consider the matrix $\boldsymbol{Y} = \{y_i\}_{i=1}^N$ with the test signals as columns. We seek the dictionary $\mathbf{\Psi}$ and the representation X that solve the minimization problem

$$\min_{\mathbf{\Psi}, X} \left\{ \|\boldsymbol{Y} - \mathbf{\Psi} X\|_F^2 \right\} \qquad \text{subject to} \qquad \|x_i\|_0 \le T_0, \quad \forall i.$$

T. Steinbrinker and N. Spicher are with the Department of Medical Informatics at the University Medicine Göttingen, mail: tabea.steinbrinker@med.uni-goettingen.de

Again, this problem in NP-hard, so with the K-SVD algorithm we seek a suitable approximation. In every iteration, we first assume $\boldsymbol{\Psi}$ to be fixed and find a coefficient matrix X. This is done via the reformulation

$$\|\boldsymbol{Y} - \boldsymbol{\Psi}X\|_F^2 = \sum_{i=1}^N \|y_i - \boldsymbol{\Psi}X\|_2^2$$

and the OMP algorithm applied to every summand.

Next, we proceed to improve the dictionary. This is done, by keeping all columns of $\boldsymbol{\Psi}$ fixed except for the kth column ψ_k. We want to update this column together with the corresponding kth row of X, x_T^k, in order to minimize the mean squared error. For this, we rewrite the penalty term and obtain

$$\begin{aligned}\|\boldsymbol{Y} - \boldsymbol{\Psi}X\|_F^2 &= \|\boldsymbol{Y} - \textstyle\sum_{j=1}^K \psi_j x_T^j\|_F^2 \\ &= \|(\boldsymbol{Y} - \textstyle\sum_{j\neq k} \psi_j x_T^j) - \psi_k x_T^k\|_F^2 \\ &= \|\boldsymbol{E}_k - \psi_k x_T^k\|_F^2, \qquad k = 1, \ldots, K. \qquad (2)\end{aligned}$$

Alternating ψ_k and x_T^k in a maximized way consist of finding the best rank-1 approximation of the error matrix $\boldsymbol{E}_k$ and this can be done via SVD. But in general, this representation will not be sparse. Therefore, we define ω_k as the group of indices pointing to the set of $\{y_i\}$ that use the atom d_k, i.e. $\omega_k = \{i | 1 \leq i \leq K, x_T^k(i) \neq 0\}$. This set will allow us to only consider the atoms that are used in our already found sparse representation. If we only alternate these entries, then the new optimized solution will still be sparse. For this, we define the matrix $\boldsymbol{\Omega}_k \in \mathbb{R}^{N \times |\omega_k|}$,

$$(\boldsymbol{\Omega}_k)_{i,j} = \begin{cases} 1, & \text{if } j = \omega_k(i) \\ 0, & \text{otherwise.} \end{cases}$$

Multiplying with this matrix, shrinks the row vector x_T^k to its non-zero entries, $x_R^k = x_T^k \boldsymbol{\Omega}_k$. Similarly, we write $\boldsymbol{Y}_k^R = \boldsymbol{Y}\boldsymbol{\Omega}_k$ to create a matrix of size $n \times |\omega_k|$ that includes the subset of training signals which currently use the ψ_k atom. Now, we can return to the penalty term as in eq. (2) and find an equivalent formulation

$$\left\|\boldsymbol{E}_k\boldsymbol{\Omega}_k - \psi_k x_T^k \boldsymbol{\Omega}_k\right\|_F^2 = \left\|\boldsymbol{E}_k^R - \psi_k x_R^k\right\|_F^2. \qquad (3)$$

Now, we can solve this via the SVD and ensure that the level of sparsity either stays the same or grows due to possible zeros occurring in the new solution. If we set $\boldsymbol{E}_k^R = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^T$ to be the singular value decomposition, the new atom in the dictionary $\tilde{\psi}_k$ is chosen as the first column of $\boldsymbol{U}$. The corresponding coefficient vector $\tilde{x}_R^k$ is chosen as the first column of $\boldsymbol{V}\boldsymbol{\Lambda}$. This minimizes eq. (3).

3) Sparse approximation: We now assume, that we have a given dictionary $\Psi \in \mathbb{R}^{n \times d}$ and a testsignal $y \in \mathbb{R}^d$ for which we want to find a sparse representation $y = \sum_\gamma \alpha_\gamma \psi_\gamma$. We solved this problem via the OMP but now want to consider a more complex algorithm. This algorithm, called the basis pursuit algorithm (BP) is computationally more expensive, but yields better approximations of the solution [9]. Relaxing the ℓ^0 "norm" from problem (1) into an ℓ^1 norm convexifies this problem and gives us the tools of linear and quadratic programming to solve them. Formally, we obtain the problem

$$\min_\alpha \|\alpha\|_1 \qquad \text{subject to} \qquad \boldsymbol{\Psi}\alpha = y. \qquad (4)$$

which we can reformulate this into a Linear Problem in standard form by replacing α by its positive and negative part, $\alpha = u - v$, where now $u, v \geq 0$ and adjusting the other parameters as needed. There are a variety of algorithms to solve linear problems in standard form.

If we want to consider noise into this problem, we need to relax the formulation $\boldsymbol{\Psi}\alpha = y$ and allow for some slack. The basis pursuit denoising algorithm (BPDN) is allows for this slack and solves the problem

$$\alpha^{(\lambda)} = \arg\min \|y - \boldsymbol{\Psi}\alpha\|_2^2 + \lambda\|\alpha\|_1, \qquad (5)$$

where now $\alpha^{(\lambda)}$ is a function of the parameter $\lambda \in \mathbb{R}$. We find that eq. (5) is equivalent to the following problem:

$$\min_{x,p} c^T x + \tfrac{1}{2}\|p\|_2^2 \quad \text{subject to} \quad \boldsymbol{A}x + p = y, \quad x \geq 0, \quad (6)$$

if we use

$$x = \begin{bmatrix} u \\ v \end{bmatrix}, \quad c = \lambda \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \boldsymbol{A} = [\boldsymbol{\Psi}, -\boldsymbol{\Psi}], \qquad \alpha = u - v.$$

This is a quadratic program with linear equality constraints and again, we have a variety of algorithms that solve this problem.

4) Dictionary building and Denoising: To train a dictionary for denoising BW, $\boldsymbol{\Psi}_{\mathrm{BW}}$, we approximate the BW in the signal [10]. Elementary sinusoidal waves with random frequencies in the range from 0–0.8 Hz are used as the first 250 training signals. The remaining training data is BW from real data, obtained from the Brno University of Technology ECG Quality Database (BUT QB) database [5]. The dictionary is trained using the K-SVD algorithm described above. The test signals are chosen to have a length of 10 s at a sampling rate of 250 Hz, i.e. $\boldsymbol{\Psi}_{\mathrm{BW}} \in \mathbb{R}^{2500 \times 100}$. To remove the noise from the signal, we use the trained dictionaries and the BPDN method to find a sparse representation of BW (α_{BW}) from the noisy ECG X_{nECG}, i.e. $X_{\mathrm{BW}} = \boldsymbol{\Psi}_{\mathrm{BW}}\alpha_{\mathrm{BW}}$. Then, we can find a clean ECG by subtracting the noise from the noisy signal:

$$X_{\mathrm{cECG}} = X_{\mathrm{nECG}} - X_{\mathrm{BW}}.$$

The other dictionaries that are trained approximate the ECG signal directly. We split up the signal into frames of 132 samples. These frames are divided into two groups, based on whether there exists a QRS complex in the frame or not. A frame is considered to have a QRS complex, if the maximum deviation from the mean value of the frame

$$d_{\max} = \max(|X'_{\mathrm{nECG}}|) - \tilde{X}_{\mathrm{nECG}}, \qquad (7)$$

where X'_{nECG} is the considered frame and $\tilde{X}_{\mathrm{nECG}}$ is the mean of this frame, is greater than a threshold. Based on the training data, the threshold was set to 0.6 mV. For the two groups obtained, different dictionaries are trained, Ψ_{nQRS} and Ψ_{QRS}. The frames that have a QRS complex in them are again sorted by the position of the R peak. For this, the frame is split up into 12 spans and a subdictionary is trained for every span. These subdictionaries are then concatenated together

$$\boldsymbol{\Psi}_{\mathrm{QRS}} = \begin{bmatrix} \boldsymbol{\Psi}_{\mathrm{QRS}}^1 & \boldsymbol{\Psi}_{\mathrm{QRS}}^2 & \cdots & \boldsymbol{\Psi}_{\mathrm{QRS}}^{12} \end{bmatrix} \in \mathbb{R}^{132 \times 1200},$$

where $\boldsymbol{\Psi}_{\mathrm{QRS}}^i \in \mathbb{R}^{132 \times 100}$ for $i = 1, \ldots, 12$. For denoising, the signal is split up into frames with a length of 132 samples. For every frame the value $d_{\max}$ is computed.

If $d_{max} < Th$, the frame does not contain a QRS complex and we estimate a sparse representation α_{nQRS} using $\boldsymbol{\Psi}_{nQRS}$ through the BPDN method. Then we determine the cleaned ECG by $\bar{X}_{nQRS} = \boldsymbol{\Psi}_{nQRS}\alpha_{nQRS}$.

If $d_{max} \geq Th$, the frame does contain a QRS complex and find the index of the maximum value and calculate the section i in which it lies. Then we find a sparse approximation α_{nQRS} using $\boldsymbol{\Psi}_{nQRS}^i$ through the BPDN method. We determine the cleaned ECG by $\bar{X}_{\mathrm{QRS}} = \boldsymbol{\Psi}_{\mathrm{QRS}}^i\alpha_{\mathrm{QRS}}$. For obtaining the cleaned signal, we concatenate the signal frames back together.

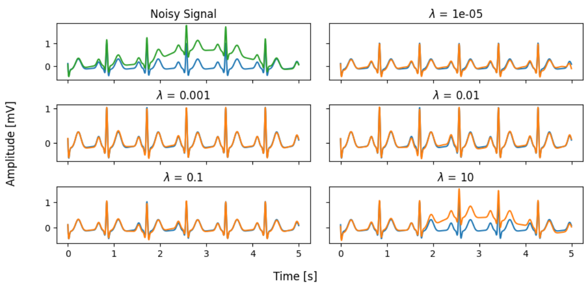

Fig. 1: Denoising of BW noise for different choices of λ.

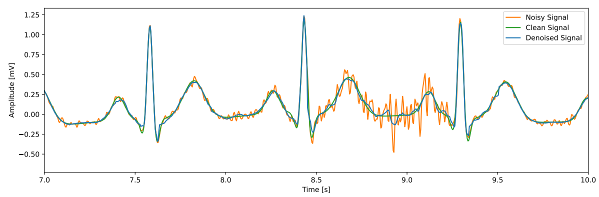

Fig. 2: Reduction of MA noise on synthetic ECG data.

C. Evaluation

The results are evaluated on noisy signals of different signal-to-noise ratios (SNR). The quality of the denoising method is quantified using the improvement of the SNR ($\Delta\,\mathrm{SNR}$) and the mean-square error (MSE):

$$\mathrm{SNR} = 10\log_{10}\frac{V[X_{\mathrm{oECG}}]}{V[X_{\mathrm{oECG}} - X_{\mathrm{nECG}}]}$$

$$\Delta\,\mathrm{SNR} = 10\log_{10}\frac{\sum_{n=1}^{N}(X_{\mathrm{nECG}}[n] - X_{\mathrm{oECG}}[n])^2}{\sum_{n=1}^{N}(X_{\mathrm{dECG}}[n] - X_{\mathrm{oECG}}[n])^2}$$

$$\mathrm{MSE} = \frac{1}{N}\sum_{n=1}^{N}(X_{\mathrm{oECG}}[n] - X_{\mathrm{dECG}}[n])^2,$$

where X_{oECG}, X_{nECG}, and X_{dECG} are the original ECG signal, noisy signal and denoised signal, respectively. N is the ECG length in samples and $V[X]$ is the variance operator.

III. Results

BPDN has an open parameter λ, regulating the compromise between approximating the signal without overfitting the noise. First, we evaluate the choice of this parameter. Second, we analyze the performance w.r.t. SNR improvement and MSE using the determined best choice. To denoise BW we use the dictionary $\boldsymbol{\Psi}_{\mathrm{BW}}$ as described above. To illustrate the role of the parameter $\lambda \in \mathbb{R}$, we generate a noisy signal by combining a synthetic ECG signal and BW noise and denoise the signal with different choices of λ as depicted in Fig. 1. The blue curve represents the clean signal, green the noisy signal and orange the reconstruction. Our findings indicate that reconstruction is effective when λ is chosen within $[0.001, 0.1]$.

As BW consists of low frequencies, a substantial amount can be reduced by usage of a high-pass filter. Hence, we compare the proposed method with a second-order Butterworth filter and a cutoff frequency of 0.4 Hz, as per the recommendation of the American Heart Association (AHA) [11]. Table I shows that throughout all SNR values, the dictionary method yields higher $\Delta\,\mathrm{SNR}$ and lower MSE values than this method.

MA are denoised using the dictionaries Ψ_{QRS} and Ψ_{nQRS}, as described above. An example of denoising a synthetic signal can be seen in Figure 2. Initial tests on synthetic data lead to selecting λ values from the interval $[0.002, 0.02]$. Table II depicts $\Delta\,\mathrm{SNR}$ and MSE for some choices of λ. It can be observed that the best choice depends heavily on SNR.

IV. Discussion

In this work, we proposed a sparse dictionary learning approach with a focus on minimizing the size of the training dataset. In contrast to more complex machine learning models, such as deep neural networks, this method is rather computational effective and shows a potential for efficient denoising using only a small number of ECGs for training.

		$\Delta\,\mathrm{SNR}$	MSE
SNR = -5	D	21.1111 ± 2.9127	0.0108 ± 0.0091
	F	17.9799 ± 3.4367	0.0248 ± 0.0273
SNR = 0	D	17.0893 ± 2.6209	0.0025 ± 0.0017
	F	13.3553 ± 4.5640	0.0112 ± 0.0224
SNR = 5	D	10.0186 ± 3.1075	0.0014 ± 0.0012
	F	5.2005 ± 5.5899	0.0101 ± 0.0229
SNR = 10	D	1.1091 ± 3.6312	0.0012 ± 0.0012
	F	-4.4098 ± 5.8072	0.0099 ± 0.0230

TABLE I: Comparison between the dictionary learning-based approach (D) and the AHA-recommended highpass filter (F).

	λ	$\Delta\,\mathrm{SNR}$	MSE
SNR = -5	0.002	3.2646 ± 0.6485	0.3861 ± 0.1613
	0.008	3.5320 ± 0.6831	0.3649 ± 0.1548
	0.02	3.8641 ± 0.7409	0.3399 ± 0.1468
SNR = 0	0.002	4.3569 ± 1.1136	0.0317 ± 0.0159
	0.008	4.8808 ± 1.2331	0.0285 ± 0.0151
	0.02	5.4331 ± 1.3564	0.0254 ± 0.0142
SNR = 5	0.002	5.1214 ± 1.1665	0.0026 ± 0.0013
	0.008	5.4623 ± 1.1991	0.0024 ± 0.0012
	0.02	5.4982 ± 1.1835	0.0024 ± 0.0011
SNR = 10	0.002	1.7465 ± 1.7063	0.0006 ± 0.0003
	0.008	0.6367 ± 1.7059	0.0007 ± 0.0003
	0.02	-0.6572 ± 1.6475	0.0009 ± 0.0004

TABLE II: Comparision between λ values in MA denoising.

References

[1] A. Singhal, M. R. Cowie. *The Role of Wearables in Heart Failure.* Curr Heart Fail Rep 17: 125–132, 2020.
[2] S. Chatterjee, R. S. Thakur, R. N. Yadav, L. Gupta, and D. K. Raghuvanshi. *Review of noise removal techniques in ecg signals.* IET Signal Processing 14(9): 569-590, 2020
[3] D. Makowski, et al. *NeuroKit2: A python toolbox for neurophysiological signal processing.* Behavior Research Methods, 53(4):1689–1696, 2021.
[4] P. E. McSharry, et al. *A dynamical model for generating synthetic electrocardiogram signals.* IEEE transactions on biomedical engineering, 50(3):289–294, 2003.
[5] A. Nemcova, et al. *Brno university of technology ecg quality database (but qdb).* 2020.
[6] G. B. Moody, W. Muldrow, and R. G. Mark. *A noise stress test for arrhythmia detectors.* Computers in cardiology, 11(3):381–384, 1984
[7] S. Mallat and Z. Zhang. *Matching pursuits with time-frequency dictionaries.* IEEE Transactions on Signal Processing, 41(12):3397–3415, 1993
[8] M. Aharon, M. Elad, and A. Bruckstein. *rmk-svd: An algorithm for designing overcomplete dictionaries for sparse representation.* IEEE Transactions on Signal Processing, 54(11):4311–4322, 2006
[9] S. S. Chen, D. L. Donoho, and M. A. Saunders. *Atomic decomposition by basis pursuit.* SIAM Journal on Scientific Computing, 20(1):33–61, 1998.
[10] M. Rakshit and S. Das. Hybrid approach for ecg signal enhancement using dictionary learning-based sparse representation. IET Science, Measurement & Technology, 13(3):381–391, 2019.
[11] P. Kligfield, et al. *Recommendations for the standardization and interpretation of the electrocardiogram: part I* Circulation vol. 115,10 (2007): 1306-24.