Fetal Electrocardiogram analysis is a very important procedure for diagnosing the health of a child in the womb. This is done to monitor the condition of the fetus during pregnancy.

The fetal ECG signal is recorded using one electrode placed on the fetal head (Fetal Electrocardiogram) and 4 more signals from the mother's abdomen (Abdominal Electrocardiogram). However, when recording the signals in this way, the maternal and fetal ECG signals overlap and the fetal signal is distorted, so we need to clean it up to see the fetal heart rate and analyse it. To do this, we need to implement and compare different methods for cleaning the FECG from the maternal signal.

The signal database consists of five recordings from births between 38 and 41 weeks of gestation. The input bands, a direct electrocardiogram recorded from the fetal head, four signals from the mother's abdomen, and a constant electrode position during all recordings are recorded in .edf files. The signal sampling rate is 1 kHz, the bandwidth is from 1 to 150 Hz. Additional filtering was performed to eliminate power line interference (50 Hz) and baseline drift. The resolution is 16 bits.

Similar databases (with files of ECG signals and R-peak locations) were analysed in different ways to obtain pure FECG.
One of the articles [2] discussed the method of cleaning the FECG using a double Savitzky-Golay filter.

Very often, filtering removes the necessary data from the signal and smoothes its peaks and troughs too much, which is bad for further analysis, because the main and important features of the signal disappear. The Savitzky-Golay filter maintains the integrity of the original signal while preserving its features, which is useful for our task of cleaning the FECG.

The principle of this filter is to move a fixed-size window over the data and fit a polynomial to the points in this window. The polynomial value at the centre point of this window is taken as the smoothed value. This process is repeated for each point in the data set, resulting in a smoothed signal.

The basic idea of the method is to approximate the data points in a moving window with a polynomial of a certain degree. For example, for a set of points (x_i, y_i), where i varies from 1 to N, we want to fit a polynomial of degree p to these points.

y = a_0 + a_1 x + a_2 x^2 + ... + a_p x^p

For a given window, we need to choose the coefficients a_0, a_1... so that they are the best possible fit for the points that are in that window.

$$\text{Minimize} \sum_{i=-m}^{m} \left(y_{k+i} - \sum_{j=0}^{p} a_j x_{k+i}^{j} \right)^2$$

Standard cases:

- Small window size, low polynomial degree: the filter smoothes the data, but does not capture the overall trend well, especially for higher frequency components.
- Small window size, high polynomial: the filter does not capture the more complex trends in the signal well.
- Large window size, low polynomial power: The filter smooths everything consistently, but it can smooth out important signal characteristics.
- Large window size, high polynomial: the filter captures complex trends, which results in effective smoothing, but the window size may be too large relative to the frequency components of the signal.

For our signal, we'd rather choose a small window size and a low polynomial degree to smooth the data effectively.

In another article ([3]), we discussed the SVD method, Singular Value Decomposition, which is an algorithm for generalising a matrix decomposition to any matrix, without assuming that the matrix is square.

The SVD formula looks like this:

M = UΣV ^{t}, where:

- M is the original matrix we want to decompose.
- U is a left singular matrix (columns are left singular vectors). The columns of U contain the eigenvectors of the matrix MM ^{t}.
- Σ is a diagonal matrix containing singular eigenvalues
- V is a right singular matrix (columns are right singular vectors). V columns contain the eigenvectors of the matrix M ^{t} M.

This algorithm works as follows:

- Changes the basis from the standard basis to the V basis (using V^T)
- Applies the transformation described by the matrix Σ. This scales our vector in the new basis V
- Change the basis V to the basis U

The second method of FECG cleaning, which was described in the same article, [3] & [4] is the ICA method. Independent Component Analysis (ICA) is a technique that allows you to separate and identify underlying independent sources in a multivariate data set.

This analysis technique is important because it allows you to understand the hidden structure of a data set and can be used in a variety of applications, from signal processing to finance. This is an unsupervised learning algorithm in the field of artificial intelligence, the essence of which is to divide a multidimensional signal into additional subcomponents to filter out unnecessary noise and signals that distort the signal of interest.

This algorithm works according to the following formula:

$$\vec{s} = A^{-1} * \vec{x}$$

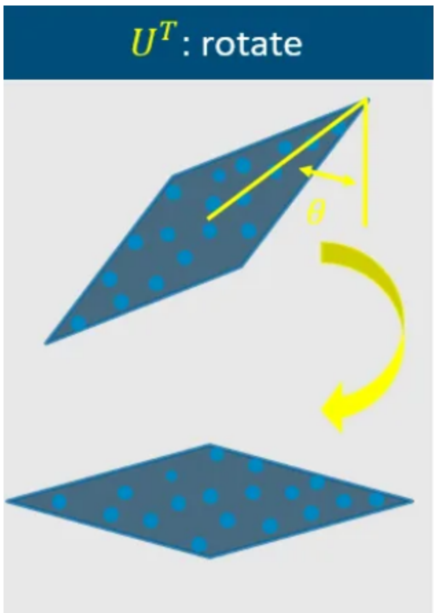
, where

- S - independent signals we want to receive
- A is the mother signal mixing matrix
- x - measured signals

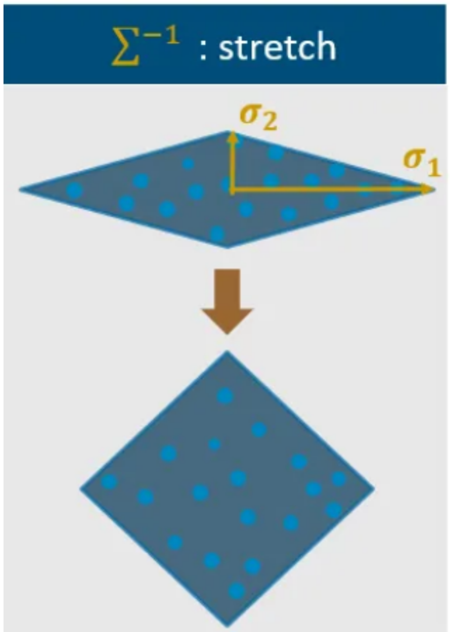
However, to obtain the matrix A, the mother signal mixing matrix, additional steps are required:

$$A^{-1} = (U^T \Sigma^{-1} V)$$

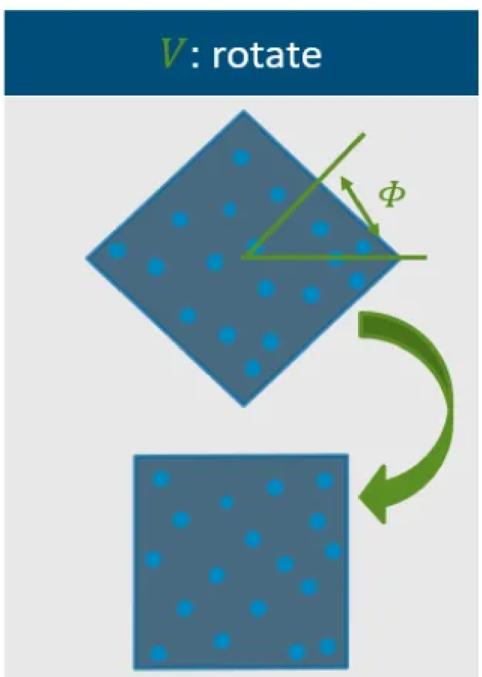
1) Finding the angle with the maximum variance for a rotation (U^T)
   The first theta rotation angle is found using principal component analysis (PCA).



2) Finding the scaling of the principal components (Σ)
   The purpose of this step is to stretch our matrix using the outliers from the sigma 1 and sigma 2 data.



3) Final rotation of the matrix to restore the signal
   Rotate the matrix around the Phi angle to restore the input signal dimensions.



Also, the method of replacing the mother signal in the FECG with the value of the isoline was discussed in [8].

Method for determining the location of the QRS complex in the ECG signal. Its essence lies in the detection of QRS complex areas with the subsequent imposition of an isoline on these areas. This allows you to get a cleaned signal for further analysis.

The implementation of all these methods is available in the .ipynb file, along with all the graphs for each .edf file. Everything was implemented in Python using the tools of the sklearn library (FastICA, TruncatedSVD) and scipy (savgol_filter, butter, filtfilt).
The effectiveness was evaluated by the SNR value.

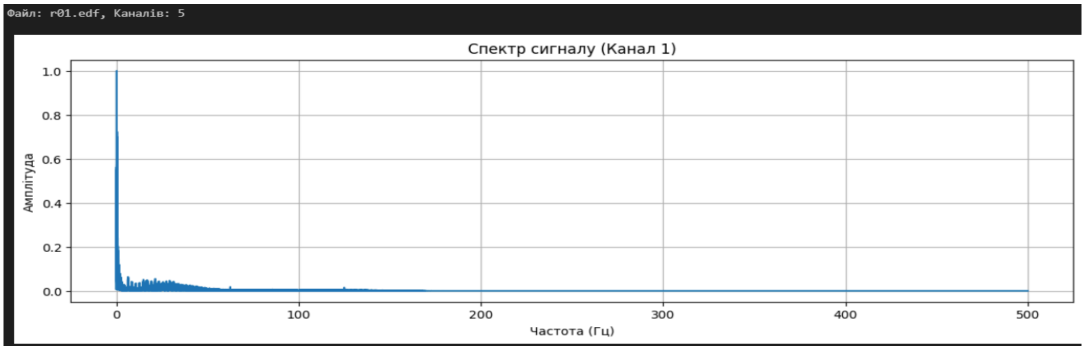
Results:
Our .edf files contain 5 signals - 1 baby signal and 4 mother signals.
The efficiency of the method was evaluated by graphs and SNR value.

1) Signal spectra.

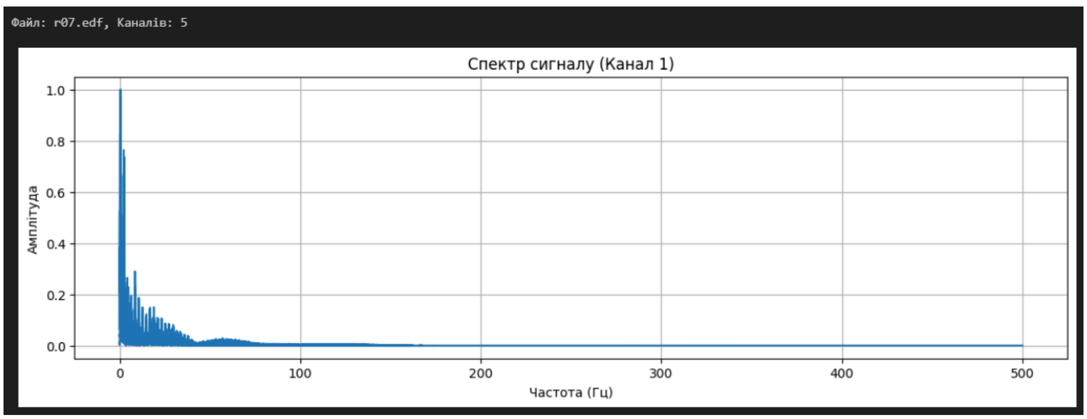
From the graphs of the signal spectra, we can conclude that there are a large number of low-frequency components in the signals, and some signal channels show a more even distribution of energy across frequencies. In general, the high values on the graph are present at low frequencies, and there are almost no spikes at high frequencies.

In these plots, the X-axis is the frequency axis in Hz and the Y-axis is the amplitude axis

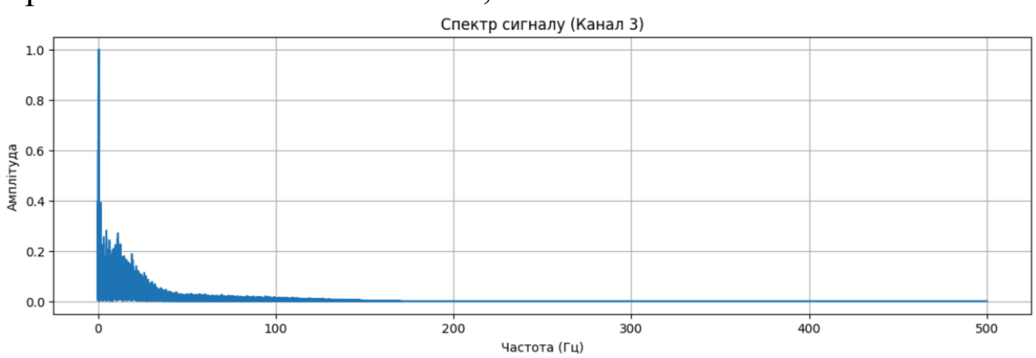
Signal Spectra 1 Channel: File r01.edf, Channels: 5



Signal Spectra 1 Channel: File r07.edf, Channels: 5



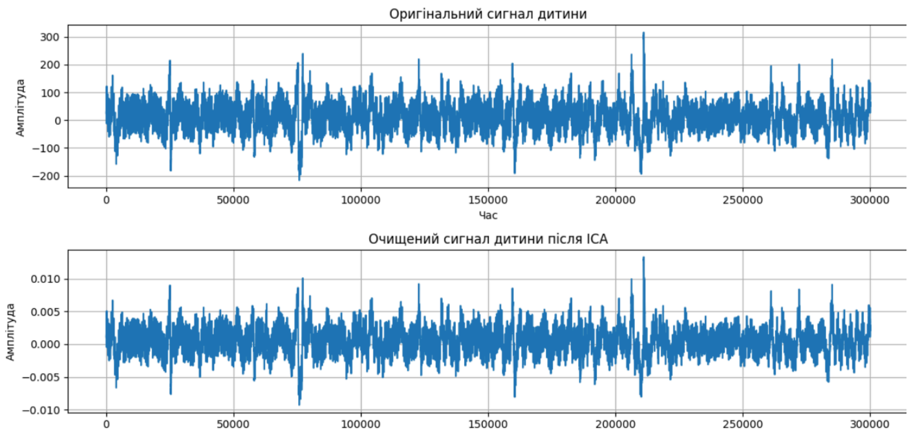
Signal Spectra 3 Channel: File: r10.edf, Channels: 5

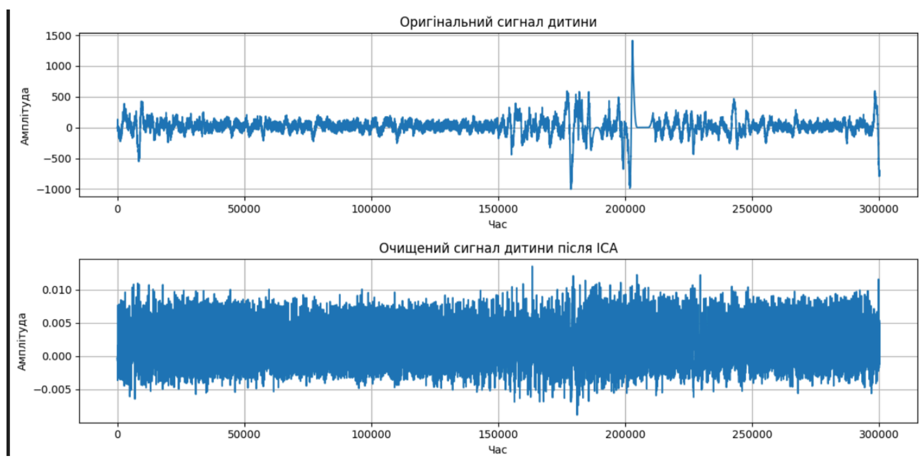


ICA:

As we can see for the first file, there is almost no cleaning (first image of the graphs), and for the last file, there is no cleaning (second image of the graphs). In terms of SNR (table after the results of all methods), we can conclude that this method did not perform well.

In these graphs, the first graph is the child's original signal, and the second is the cleaned signal after ICA. The axes have the same meaning as in the upper graphs - X is the frequency in Hertz, Y is the amplitude.



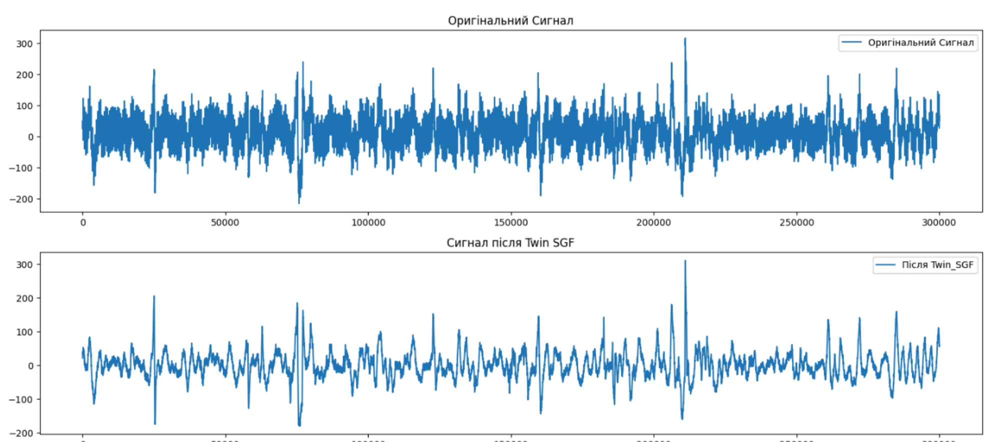


SVD:
Based on the graphs of this method and its SNR values, we can also conclude that it did not perform very well.

In these graphs, the first graph is the child's original signal, and the second is the cleaned signal after SVD. The axes have the same meaning as in the upper graphs - X is the frequency in Hertz, Y is the amplitude.

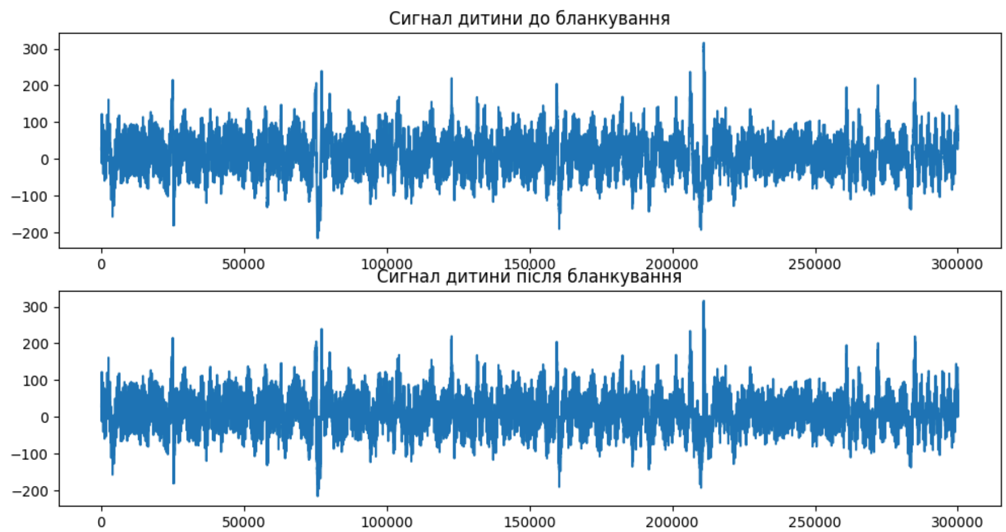
Double Savitzky-Golay filtering:
This method works better, the graph shows that the signal is more cleaned than the original and the SNR values are positive, which makes us understand that this method performs the task.

In these graphs, the first graph is the child's original signal, and the second is the cleaned signal after Double Savitzky-Golay filtering. The axes have the same meaning as in the upper graphs - X is the frequency in Hertz, Y is the amplitude.



Isoline:
From the graph and the SNR value, we can conclude that this method also cleans up our signal quite well.

In these graphs, the first graph is the child's original signal, and the second is the cleaned signal after filtering. The axes have the same meaning as in the upper graphs - X is the frequency in Hertz, Y is the amplitude.



The SNR value:

Method\File	r01.edf	r04.edf	r07.edf	r08.edf	r10.edf
ICA	-87.52 дБ	-93.24 дБ	-86.20 дБ	-91.64 дБ	-97.80 дБ
SVD	-136.00 дБ	-138.85 дБ	-135.16 дБ	-139.15 дБ	-151.21 дБ
Double Savitzky-Golay filtering	11.00 дБ	9.10 дБ	15.34 дБ	10.52 дБ	8.53 дБ
Isoline	12.06 дБ	12.59 дБ	12.37 дБ	12.16 дБ	12.63 дБ

As we can see from the values in the table, the Isoline method has the best SNR values, but at the same time, the Savitsky-Golay method has very good values. The ICA and SVD methods have negative SNR values, which is bad, but the reason for this may be an error in the implementation of these methods.

From these data, it can be concluded that the Isoline and Savitzky-Golay methods are best suited for cleaning FECGs from noise and the influence of mother signals.

Further research will include improving the implementation of the SVD and ICA methods, which will likely lead to better performance of these methods for a particular task.

References:

[1] PhysioNet. (n.d.). Abdominal and Direct Fetal ECG Database, Version 1.0.0. Retrieved from https://physionet.org/content/adfecgdb/1.0.0/

[2] S. R. Breesha, S. S. Vinsley (2023) Automated Extraction of Fetal ECG Signal Features Using Twinned Filter and Integrated Methodologies https://link.springer.com/article/10.1007/s00034-023-02494-0

[3] M. Kotas, J. Jezewski, K. Horoba, A. Matonia (2011) Application of spatio-temporal filtering to fetal electrocardiogram enhancement https://www.sciencedirect.com/science/article/abs/pii/S0169260710001781

[4] M. Kotas, J. Jezewski, A. Matonia, T. Kupka (2010)
Towards noise immune detection of fetal QRS complexes
https://www.sciencedirect.com/science/article/abs/pii/S0169260709002727

[5] Jonas Dieckmann (2023) Introduction to ICA: Independent Component Analysis
https://towardsdatascience.com/introduction-to-ica-independent-component-analysis-b2c3c4720cd9

[6] Risto Hinno (2024) Singular Value Decomposition (SVD) Algorithm Explained https://builtin.com/articles/svd-algorithm

[7] Thomas Konstantinovsky (2024) Introduction to the Savitzky-Golay Filter: A Comprehensive Guide (Using Python)

https://medium.com/pythoneers/introduction-to-the-savitzky-golay-filter-a-comprehensive-guide-using-python-b2dd07a8e2ce

[8] Adam Matonia, Janusz Jezewski, Tomasz Kupka, Krzysztof Horoba, Janusz Wrobel, Adam Gacek (2006) The influence of coincidence of fetal and maternal QRS complexes on fetal heart rate reliability https://link.springer.com/article/10.1007/s11517-006-0054-0#citeas