1 METHODS

This section details the methods used in this paper to investigate its hypothesis: does the use of the MSAX representation improve the performance of the HOT SAX anomaly detection algorithm applied to ECGs compared to the SAX representation? TODO fix this and make congruent with hypothesis. The following section will first cover the mathematical foundations of SAX, MSAX, and HOT SAX. Then, the statistical methods used to analyze the results will be presented, followed by a note on the implementation of the mathematical methods and statistical analysis, and the data used in this research. Lastly, the limitations of these methods will be discussed. TODO make sure this fits the actual structure

While MSAX and SAX both are time series representation methods, they can be applied to ECGs, as EGCs are discrete multivariate time series. Mathematically, a discrete time series is a series of T observations made at discrete points in time, with n values recorded at each moment in time. Following [anacleto2020],

$$\{\mathbf{v}[t]\}_{t\in\{1,\dots,T\}} \tag{1.1}$$

is a n-variate time series where, for each time point t,

$$\mathbf{v}[t] = (v_1[t], \dots, v_n[t])^T \tag{1.2}$$

represents the values of the time series. If the time series has $n = 1$ values at each time point, it is called univariate, if $n > 1$, it is called multivariate. For ECGs, the discrete points in time are dictated by the sampling frequency, which is the number of observations made in one second. The number of leads in an ECG is equivalent to the variable n in (1.2). As virtually all ECGs consist of more than one lead ($n > 1$), ECGs are multivariate time series.

1.1 Mathematical Foundations

The main method used in this research is the SAX representation. It will be compared to MSAX, a multivariate version of the representation. Both representations will be used with the HOT SAX algorithm to detect ECG anomalies. TODO make sure this is accurate

1.1.1 SAX

The Symbolic Aggregate Approximation, introduced in 2003 by Lin, Keogh, Lonardi, and Chiu, is a symbolic time series representation [lin2003]. Its main features are the symbolic representation and dimension reduction of time series data, and the lower bounding of the Euclidean Distance. A lower bound (or infimum) in set theory is a value that is the largest element in a set S that is smaller than all elements in a certain subset of S. For SAX, lower bounding the Euclidean Distance can be understood as stating that the SAX distance between two SAX representations is guaranteed to be smaller than or equal to the "true" or Euclidean Distance between the original time series. Accordingly, the distance between two SAX representations is guaranteed to be representative of the Euclidean Distance between the raw time series. The SAX representation only works for time

series $\mathbf{v}[t]$ for which $n = 1$, i.e. which are univariate. Thus (1.1) becomes $\mathbf{v}[t] = v_1[t]$. Using the SAX representation is a three-step process. Firstly, the raw time series is normalized. Secondly, the dimension of the normalized time series is reduced using PAA. Thirdly, the PAA-represented time series is discretized. Additionally, a distance measure between two SAX representations is defined.

Normalization. The normalization for the SAX representation is necessary because, to compare two time series, it is standard practice to normalize both of them because otherwise comparisons between them are not useful [lin2003]. SAX is normalized by applying standard Z-normalization, resulting in a time series with sample mean equal to 0 and sample standard deviation equal to 1. To do this, the mean and standard deviation of the univariate time series $\mathbf{v}[t]$ needs to be calculated. The sample mean of a list of values is

$$\bar{x} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{v}[t].$$

The sample standard deviation can be found with the formula

$$s = \sqrt{\frac{1}{T-1}\sum_{t=1}^{T} (\mathbf{v}[t] - \bar{x})^2}$$

(It should be noted that for applications to whole ECGs, the sample standard deviation and population standard deviation are very similar, as T is often > 100.000). Finally, the normalized time series values can be obtained by computing

$$\mathbf{v}[t] = \frac{\mathbf{v}[t] - \bar{x}}{s}, \qquad \forall t \in \{1, \dots, T\}.$$

The resulting time series will have the same shape as the raw time series, but it will have no unit and be normalized.

TODO insert figure here

Dimension reduction with PAA. The dimension reduction of the SAX representation is due to the use of PAA. The PAA method takes a univariate time series $\mathbf{v}[t]$ of length T and an integer w and segments $\mathbf{v}[t]$ into w segments, taking the average of each. Following [lin2003], the resulting representation is denoted as $\bar{\mathbf{v}}[t]$ and now has length w. The PAA representation of $\mathbf{v}[t]$ can be calculated by using the following formula [lin2003]

$$\bar{\mathbf{v}}[t] = \frac{w}{T} \sum_{j=\frac{n}{w}(t-1)+1}^{\frac{n}{w}t} \mathbf{v}[t], \qquad \forall t \in \{1, \dots, w\}. \tag{1.3}$$

Table 1.1: Breakpoint values for numbers of breakpoints a from 3 to 6. The parameter a determines into how many equally-sized areas the normal curve $\mathcal{N}(0,1)$ is split. The breakpoints β_i delimit the areas. Table contents are quoted from [lin2003].

β_i \ a	3	4	5	6
β_1	-0.43	-0.67	-0.84	-0.97
β_2	0.43	0	-0.25	-0.43
β_3	—	0.67	0.25	0
β_4	—	—	0.84	0.43
β_5	—	—	—	0.97

Now $\mathbf{v}[t]$ has been converted to the PAA representation $\bar{\mathbf{v}}[t]$. This process reduces the length of the time series from T to w, with the dimension reduction ratio depending on the choice of w. TODO insert figure here

Discretization of PAA representation. This last step in the SAX representation process involves transforming the PAA representation $\bar{\mathbf{v}}[t]$ into a sequence of equiprobable symbols. Here it is assumed that a normalized time series has a Gaussian normal distribution ($\mathcal{N}(0,1)$). The number symbols used is denoted by a–the alphabet size. To create the equiprobable symbols, Lin, Keogh, Lonardi, and Chiu [1] use so-called "breakpoints". These breakpoints are a sorted list of numbers $B = \beta_1, \ldots, \beta_{a-1}$. The area under the normal curve $\mathcal{N}(0,1)$ (i.e. the probability) between two consecutive segments β_i and $\beta_{i+1} = 1/a$. This creates a segments ($a-1$ breakpoints) of $\mathcal{N}(0,1)$ that have the same area, i.e. the same probability. The values of the breakpoints in B can be found in a Z-table. For illustration, Table 1.1 shows the breakpoint values for $a = 3$ to $a = 6$.

Once the breakpoint values have been determined, the discretization process begins. The process assigns all PAA segments whose value is below β_1 the symbol "a". The PAA segments falling in the area $\beta_1 \leq$ and $< \beta_2$ are assigned "b". This mapping process is continued, until all PAA segments are symbolized. Now we have arrived at the SAX representation. The SAX representation of $\bar{\mathbf{v}}[t]$ is denoted $\hat{\mathbf{v}}[t]$ and has the same length as $\bar{\mathbf{v}}[t]$ (w). Mathematically, the discretization process is formulated in [lin2003] as

$$\hat{\mathbf{v}}[t] = \text{alpha}_j \quad \text{if } \beta_{j-1} \leq \hat{\mathbf{v}}[t] < \beta_j, \qquad \forall t \in \{1, \ldots, w\}.$$

Here alpha_j is the jth letter of the alphabet, i.e. alpha_1 = "a", alpha_2 = "b" … The resulting time series representation has an even more reduced dimension than PAA because instead of infinitely many possible values for the real-valued PAA values, now there are only a different, equiprobable symbols. Thus, the SAX representation $\hat{\mathbf{v}}[t]$ has been obtained. TODO insert graph here

SAX distance measure. A distance measure between two SAX representations of the same length is required to be able to compare them with each other. The SAX distance function is based on the

Table 1.2: A table for the dist function for $a = 5$. Each cell displays the distance between the symbols denoting its row and column. The formula for the cell values is (1.5).

	a	b	c	d	e
a	-0.43	-0.67	-0.84	-0.97	
b	0.43	0	-0.25	-0.43	
c	0	0.67	0.25	0	
d	0	0	0.84	0.43	
e	0	0	0	0.97	

Euclidean Distance between two time series $\mathbf{v}[t]$ and $\mathbf{u}[t]$ is [lin2003]

$$\mathrm{D}\left(\mathbf{u}[t], \mathbf{v}[t]\right) \equiv \sqrt{\sum_{t=1}^{T} \left(\mathbf{u}[t] - \mathbf{v}[t]\right)^2}.$$

Through the PAA distance as an intermediate step, the authors arrive at MINDIST in (1.4), the SAX distance function that returns the minimum distance between the two original time series. It is defined as [lin2003]

$$\mathrm{MINDIST}\left(\hat{\mathbf{u}}[t], \hat{\mathbf{v}}[t]\right) \equiv \sqrt{\frac{T}{w}} \sqrt{\sum_{t=1}^{w} \left(\mathrm{dist}(\hat{\mathbf{u}}[t], \hat{\mathbf{v}}[t])\right)^2}. \tag{1.4}$$

The function dist is based on a lookup table that contains the distances between two symbols. Table 1.2 shows the lookup table for $a = 5$. The values of each table cell are 0 for symbols letters or the absolute difference of the breakpoints otherwise. The formula

$$\mathrm{cell}_{r,c} = \begin{cases} 0, & \text{if } |r - c| \leq 1 \\ \beta_{\max(r,c)-1} - \beta_{\min(r,c)}, & \text{otherwise} \end{cases} \tag{1.5}$$

is used to calculate the values of each cell in Table 1.2 by r (row) and c (column) [lin2003]. **TODO** inset example of distance between a short segment of SAX stuff, maybe two ecg segments

1.1.2 MSAX

The Multivariate Symbolic Aggregate Approximation was introduced by Anacleto, Vinga, and Carvalho in 2020. It is an extension of SAX to multivariate time series [anacleto2020]. It shares the main features of SAX, but expands them to multivariate time series, such as ECGs–n can be any integer ≥ 1. A lower bound for the MSAX distance function also exists, i.e. distance between two MSAX representations is, just as in SAX, guaranteed to be representative of the Euclidean Distance between the raw time series. Using the MSAX representation has the same steps as SAX: normalization, PAA-based dimension reduction, and discretization. A variation of the MINDIST

function exists, too.

Normalization. The rationale for normalization in the MSAX representation is twofold. Firstly, the same considerations as for SAX apply with regards to comparing two time series. Secondly, MSAX utilizes multivariate normalization to take advantage of the covariance structure of multivariate time series data. To avoid confusion with the previous section, a multivariate time series shall be denoted as $\mathbf{V}[t]$. Multivariate normalization relies on a sample mean vector containing the sample mean for each of the time series $(V_1[t], \dots, V_n[t])^T$ in $\mathbf{V}[t]$. The sample standard deviation is replaced by a covariance matrix. The sample mean vector is equivalent to the vector of expected values $\vec{E}$, following [anacleto2020]:

$$E(\mathbf{V}[t]) = \vec{E} = \begin{bmatrix} \text{mean}(V_1[t]) \\ \vdots \\ \text{mean}(V_n[t]) \end{bmatrix} = \begin{bmatrix} \frac{1}{T}\sum_{t=1}^{T} V_1[t] \\ \vdots \\ \frac{1}{T}\sum_{t=1}^{T} V_n[t] \end{bmatrix}.$$

The covariance matrix, an $n \times n$ matrix, contains the variance of each part $(V_1[t], \dots, V_n[t])^T$ of $\mathbf{V}[t]$ on its main diagonal, and the covariance between ith and jth parts of $\mathbf{V}[t]$ in the (i, j) position. The general form of a covariance matrix is shown in (1.6) below. The covariance matrix is denoted as $\text{Var}(\mathbf{V}[t])$ or $\Sigma_{n\times n}$. It is calculated as:

$$\text{Var}(\mathbf{V}[t]) = \Sigma_{n\times n} = \begin{bmatrix} \text{cov}(V_1, V_1) & \dots & \text{cov}(V_1, V_n) \\ \vdots & \ddots & \vdots \\ \text{cov}(V_n, V_1) & \dots & \text{cov}(V_n, V_n) \end{bmatrix}. \tag{1.6}$$

The covariance of two time series parts $V_i[t]$ and $V_j[t]$ is defined as the mean of product of the difference between the values of $V_i[t]$ and its expected value. The following equation illustrates this process:

$$\text{cov}(V_i[t], V_j[t]) = E\Big(\big[V_i[t] - E(V_i[t])\big] \cdot \big[V_j[t] - E(V_j[t])\big]\Big)$$

(Note that $\mathbf{V}[t]$ can be conceptualized as a matrix, with its row representing the different sub-series and the columns representing specific values of t). Once $\vec{E}$ and $\Sigma_{n\times n}$ have been found, the time series $\mathbf{V}[t]$ can be normalized by the following formula [anacleto2020]:

$$\mathbf{V}[t] = (\Sigma_{n\times n})^{-1/2}\left(\mathbf{V}[t] - \vec{E}\right).$$

The result will have a mean of zero and uncorrelated variables [anacleto2020].

TODO insert figure here, use both leads in both graphs; may turn out a little large

Dimension reduction with PAA. Dimension reduction using PAA for MSAX is performed in exactly the same way as for SAX. The procedure outlined in the previous section is applied to each of the elements $(V_1[t], \dots, V_n[t])^T$ of the time series $\mathbf{V}[t]$–equation (1.3) is applied to each part. This results in a PAA representation of the original time series $\overline{\mathbf{V}}[t] = \left(\overline{V_1}[t], \dots, \overline{V_n}[t]\right)^T$.

This process also reduces the length of the time series from T to w for each sub-series, with the dimension reduction ratio depending on the choice of w [anacleto2020] TODO insert figure here

Discretization of PAA representation. The discretization of the PAA representation for MSAX also works like it does for SAX. Like in the previous paragraph, the process used in the SAX representation is applied to each of the sub-time series in $\overline{\mathbf{V}}[t]$ to obtain $\widehat{\mathbf{V}}[t]$. The alphabet size a is the same for each $V_n[t]$ and the symbols are found in the same way as in the SAX representation. The breakpoint values are calculated the same and Table 1.1 is as valid for MSAX as it is for SAX. The assigning of symbols is generally performed in the same way, too. For bivariate time series ($n = 2$), $V_1[t]$ could be assigned lowercase symbols ("a", "b" …) while $V_2[t]$ could be assigned uppercase symbols ("A", "B" …). This has no impact on the method, it is simply a visual aid for the viewer to distinguish the values. The final MSAX representation $\widehat{\mathbf{V}}[t]$ will consist of one long list of symbols because for each moment t all generated symbols are combined into a list for this time that represent all sub-time series at that time. TODO add a graph and example here for illustration

MSAX distance measure. The MSAX distance measure expands MINDIST to multivariate time series. This is done by adding an additional summarization step to the MINDIST function. The MSAX distance MINDIST_MSAX operates on two MSAX representations $\widehat{\mathbf{U}}[t], \widehat{\mathbf{V}}[t]$. Both representations must have the same length w and same number n. MINDIST_MSAX sums the distances between the individual elements $U_i[t], V_i[t]$ for $i = \overline{1, \dots, n}$. The following equations expresses MINDIST_MSAX [anacleto2020].

$$\text{MINDIST\_MSAX}\left(\widehat{\mathbf{U}}[t], \widehat{\mathbf{V}}[t]\right) \equiv \sqrt{\frac{T}{w}} \sqrt{\sum_{t=1}^{w} \left(\sum_{i=1}^{n} \left(\text{dist}(\widehat{U}_i[t], \widehat{V}_i[t])\right)^2 \right)}. \tag{1.7}$$

The function dist is the same as the SAX function, being based on equation (1.5) and lookup tables like Table 1.2. Like MINDIST, MINDIST_MSAX also lower bounds the Euclidean Distance and derives all the same benefits from that. TODO inset example of distance between a short segment of SAX stuff, maybe two ecg segments

1.1.3 MSAX

- idea
- normalization
- dimensionality reduction
- discretization
- distance measure
- TODO all with graphs and formulas
- TODO points out differences to SAX

1.1.4 HOTSAX

- what is hotsax
- its theoretical foundations
- advantages, disadvantages
- how does it work

TODO the idea is to use the ECG as it would be recorded or digitized by anyone, without filtering. see zhang2019 if they did filtering

TODO make a final applications section that explains how HOT SAX will be used with each of the time series

TODO why can filtering be ignored? put this as further research to investigate the influence of filtering on this process

TODO give good reasons why I chose the methods and data bases

TODO goals:
- reader can assess believability of results
- all information necessary to replicate the research
- describe all materials, procedure, ect
- all the formulae
- state all the limitations of the methods and the ones I impose myself
- analytical methods and languages

TODO answer questions:
- can someone else accurately replicate the study
- can the data be obtained again
- are all parts / instruments described with enough accuracy
- is the data freely available
- can the statistical analysis be repeated
- can the algorithms be replicated?

TODO Sections:
- general overview -> flowchart
- then explain each element of the flowchart one by one
- use formulae etc
- nice amount of tikz graphs
- section on implementation with details and the more important elements
use another flow chart?
- use graphs to illustrate all important elements
- make a data description section that describes my process of data handling; which database
- explain the parameters that the methods have and what they mean
- describe how a got all the data

This section explains the methods used in this research. TODO create flow charts for all this shit to make it simpler. First methods section for the analytical methods in a mathematical way.

1.2 Statistical Analysis of Results

- explain true positive, true negative, and so on
- explain recall, accuracy, precision, f1
- explain why recall was chosen and if that is fair
- introduce the correlations that we would expect to find if my hypothesis is true and also the ones that would disprove it
- which types of correlation, significance testing, and modeling will be used and why; what are the justifications

1.3 Implementation

How I implemented the above stuff. Languages, approaches, hurdles, all the details needed to reproduce this research. Also mention the simplifications I chose to make and why: no sliding window, only even divisors, only divisors within sampling frequency and cutting ECG to even multiple of sampling frequency.

1.3.1 ECG acquisition

flow chart for process

- where to download
- what exactly are the ECGs
- where do they come from
- technical parameters of them
- the physionet suite
- annotations, what they mean, how I can get them, etc

1.3.2 preprocessing

flow chart for process

TODO the codes and constants given for each thing

- how were they preprocessed
- physionet suite
- my script and what it does and why
- problems and limitations of this
- libraries used

1.3.3 SAX

TODO how was the whole data thing handled, how is the data created

flow chart for process

- how was sax implemented
- how does HOTSAX work here

- libraries used

1.3.4 MSAX

flow chart for process

- how was sax implemented
- how does HOTSAX work here
- TODO point out differences to SAX
- libraries used

1.4 Statistical Evaluation

- reading the data into R
- summarizing the data
- the summarized data files
- libraries used

REFERENCES

[1] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, "A symbolic representation of time series, with implications for streaming algorithms," en, in *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery - DMKD '03*, San Diego, California: ACM Press, 2003, pp. 2–11. DOI: 10.1145/882082.882086.

[2] M. Anacleto, S. Vinga, and A. M. Carvalho, "MSAX: Multivariate Symbolic Aggregate Approximation for Time Series Classification," en, in *Computational Intelligence Methods for Bioinformatics and Biostatistics*, P. Cazzaniga, D. Besozzi, I. Merelli, and L. Manzoni, Eds., ser. Lecture Notes in Computer Science, Cham: Springer International Publishing, 2020, pp. 90–97, ISBN: 978-3-030-63061-4. DOI: 10.1007/978-3-030-63061-4_9.