ECG Feature Detection Using SAS®
Suhas R. Sanjee, Merck & Co., Inc., Upper Gwynedd, PA
Sridhar Vijendra, MaxisIT Inc., Metuchen, NJ

ABSTRACT
ECG analysis continues to be a vital part of drug and device discovery especially in the cardiovascular franchise. Various parameters derived from the ECG waveform such as the P-R, Q-T, R-R, Q-R-S and S-T intervals, and heart rate variability serve as safety and efficacy endpoints for many clinical trials. The detection of the R-peak and the QRS complex in the ECG is an important step in deriving these endpoints. A number of algorithms based on time and frequency domains are available for detecting various aspects of the ECG waveform. ECG features are routinely identified in core labs using algorithms implemented in C, MATLAB®, and other languages. This paper explores the implementation of ECG R-peak detection using SAS®. The ECG data that is used for analysis and reporting in clinical trials is usually preprocessed. Implementing an ECG feature detection algorithm in-house using SAS® provides the flexibility to perform exploratory analysis of parameters derived from the ECG. Further, it allows the programmer or researcher to analyze the same data from exploratory perspective. Also when ECG based parameters are used as primary endpoints, it enhances the transparency of pharmaceutical and medical device companies while submitting SAS® code and data to regulatory agencies.

INTRODUCTION
Electrocardiogram (ECG) is the most commonly recorded physiological signal. It is a crucial component of all diagnostic tests involving cardiac health. The ECG signal contains information about electrical activity of the heart. A number of ECG parameters are also used as primary and secondary endpoints in drug and device trials. The ECG signal has a characteristic pattern (as shown in Figure 1) that repeats with every beat of the heart. Of the many morphological markers of the ECG, the QRS complex and the R-peak are the most significant – with the contribution of the R-peak to the R to R interval being a driving factor. The number of R peaks in a specific time interval translates to the heart rate (in beats per minute). Some studies [1] have shown that instantaneous change in R-R interval (also called Heart Rate Variability or HRV) is of clinical significance. R-wave amplitude along with the R-R interval have been used as biomarkers in some sleep apnea trials [2].

Automated detection of the R-peak in the ECG has received widespread attention due to its use in ECG Holter recorders, which are required to deliver robust performance under poor signal conditions. A number of time and frequency-domain methods have been proposed for the detection of the R-peak in the ECG [3, 4]. While MATLAB®, R®, C, and other languages have been routinely used to implement signal analysis in research environments, SAS® is not extensively used for signal analysis. In this paper, an algorithm developed by Pan & Tompkins [6] is being implemented in SAS®.

SAS® provides Interactive Matrix Language (IML) procedure which allows users to use matrix-based data elements, operators and define functions that work on matrices. This affords the usage of any matrix oriented language within the IML procedure.

Figure 1: Electrocardiogram (ECG) Waveform for one normal heartbeat showing P, QRS and T waves
DATA OVERVIEW

The ECG data used for this exercise was downloaded from MIT-BIH arrhythmia database hosted on the Physionet website (www.physionet.org). The MIT-BIH Arrhythmia Database contains ECG signals acquired from Lead II and Lead V5 from subjects studied by the BIH Arrhythmia Laboratory. The signals were acquired at 360 samples per second per channel with 11-bit resolution over a 10 mV range. In addition to the ECG signal files, the database also contains signal annotations. Two or more cardiologists independently annotated each signal. Disagreements among the cardiologists were resolved to obtain the computer-readable reference annotations for each beat. These annotations are taken as the reference to evaluate the implementation of R peak detection algorithm.

The signals and annotations can be downloaded either in ASCII or binary format. ASCII files are used here for convenience. Lead II ECG signals are used for development of the algorithm. Ten lead II ECG clips of one minute duration are used for development and evaluation of the program. Figure 2 below shows an excerpt of ECG signal from one of the clips.

![Sample ECG data from MIT-BIH Arrhythmia Database](image)

ALGORITHM FOR R PEAK DETECTION

Pan and Tompkins [6] have developed an algorithm for ECG R-peak detection. This section discusses some of the important details of the algorithm. SAS® implementation of this algorithm will be discussed in the next section. Figure 3 depicts the important parts of the algorithm.

![Schematic of the R-peak detection algorithm](image)

The first step of signal analysis is normalization which is described in the programming overview section below. The normalized ECG signal is passed through a low-pass filter to reduce muscle noise and 60-Hz power line interference. The filtering is accomplished by using the difference equation described in Equation 1 below

\[ y(n) = 2y(n - 1) - y(n - 2) + x(n) - 2x(n - 6) + x(n - 12) \ldots \]  

Equation 1
Where
y(n) represents the current output sample
x(n) represents the current input sample
y(n – p) represents the output variable with a lag of p, where p is any integer

The next stage is a high pass filter to reduce baseline wander in the ECG. Baseline wander is a low frequency variation caused by among other reasons, patient movement during recording. The high pass filter is implemented using the difference equation given in Equation 2 below.

\[ y(n) = x(n - 16) - \frac{1}{32} [ y(n - 1) + x(n) - x(n - 32)] \] \text{ ... Equation 2}

Where
y(n) represents the current output sample of the high-pass filter
x(n) represents the current input sample to the high-pass filter
y(n – p) represents the output variable with a lag of p, where p is any integer

The next stage is the derivative filter. A derivative filter helps in identifying a change in direction in the slope of the signal which is indicative of a peak in the signal and is implemented using the difference equation given in Equation 3 below.

\[ y(n) = \frac{1}{8} [2x(n) + x(n - 1) - x(n - 3) - 2x(n - 4)] \] \text{ ... Equation 3}

Where
y(n) represents the current output sample of this differentiator stage
x(n) represents the current input sample to the differentiator stage
x(n – p) represents the input variable with a lag of p, where p is any integer

The next stage, a simple squaring function, helps not only in making all the signal values positive but also amplifies the output of the previous stage in a nonlinear manner thus emphasizing the R peaks in the signal.

The final stage is a moving window summation of the previous N samples of the output of the previous stage. N is decided based on the sampling rate of the signal being analyzed. This moving window integral is implemented using the difference equation shown in Equation 4 below.

\[ y(n) = \frac{1}{N} [ x(n - (N - 1)) + x(n - (N - 2T)) + \ldots + x(n)] \] \text{ ... Equation 4}

Where
y(n) represents the current output sample of the moving integral stage
x(n) represents the current input sample to the moving integral stage
x(n – p) represents the input signal with a lag of p, where p is any integer
N represents the window length which was chosen as 32

The output of this stage is passed through a thresholding stage which identifies peaks in the ECG signal by setting a threshold on the output of the last stage. This part of the algorithm is described in greater detail in next section.

SAS PROGRAMMING OVERVIEW

The difference equations, moving window integral and thresholding described in the previous section, are implemented using the IML procedure. Graphs are generated by using the GPLOT procedure.

STEP 1: NORMALIZATION

The raw data is normalized by subtracting each data point by its mean and dividing by the range.
STEP 2: DIFFERENCE EQUATIONS

Normalized ECG data is passed through band-pass filter and then through a derivative filter. The difference equations for these filters are shown in equations 1, 2, 3, and 4 in previous sections. These difference equations are implemented in SAS® using the IML procedure. Implementation of low pass filter is shown in Figure 4.

In figure 4, SAS® code has been provided to illustrate the implementation of difference equation using a one dimensional matrix and a loop. Difference equations for high pass and derivative filters are implemented in a similar manner. The output from the band pass filter is subjected to squaring function.

Note: Signal values at negative time are assumed to be zero.

```
PROC IML;
    USE ecg;
    *Read input ECG data from SAS dataset into matrix;
    READ all var {ecg_d} into ecg;
    *Define the matrix and initialize all
    the elements to zero;
    filt=j(nrow(ecg),1,0);

    *Implement difference equation for a low pass filter;
    filt[1,1]=ecg[1,1];
    filt[2,1]=2*filt[1,1]+ecg[2,1];
    DO i=3 to nrow(ecg);
        IF i<7 then
            filt[i,1]=2*filt[i-1,1]-filt[i-2,1]+ecg[i,1];
        IF i>=7 & i<13 then
            filt[i,1]=2*filt[i-1,1]-filt[i-2,1]+ecg[i,1]-2*ecg[i-6,1];
        IF i>=13 then
            filt[i,1]=2*filt[i-1,1]-filt[i-2,1]+ecg[i,1]-2*ecg[i-6,1]
                    +ecg[i-12,1];
    END;

    *Define variable names for output dataset;
    varnames = {"lowpass"};

    *Create SAS dataset from matrix;
    CREATE filtered FROM filt [colname=varnames];
    APPEND from filt;
    CLOSE filtered;
QUIT;
```

STEP 3: MOVING WINDOW INTEGRAL

Moving window integral is implemented by using nested DO loops as shown in Figure 5. Note: Signal values at negative time are assumed to be zero.
STEP 4: PEAK DETECTION AND THRESHOLDING

The peak of the moving window integral waveform is found by determining the change in slope. The waveform is searched for decreasing slope, and the time instant at which the slope becomes negative and stays negative for nine consecutive data points is considered as the peak. Slope is calculated using the following formula:

\[ \text{Slope} = x(n) - x(n-1) \]

The time instant corresponding to half the amplitude of the peak corresponds to QRS complex. This instant is found by searching a window of 40 samples before the peak.

RESULTS AND DISCUSSION

Results from each block of Figure 2 are presented below.

The ECG signal after normalization is presented in Figure 6. Note that normalization limits the range of the signal to (-1 to +1). This reduces the inter-patient variability and helps in standardizing the thresholds used in the algorithm. Note the presence of high frequency noise which is typically due to power line interference and muscle noise.
Figures 7 and 8 show the ECG signal after low pass and high pass filtering. The high frequency noise is no longer present in the filtered signal.
Figure 8: ECG data after High-pass filtering

Figure 9 shows band-pass filtered ECG after subjecting to derivative filtering and squaring function. Note that P and T waves are attenuated and peak-to-peak signal corresponding to the QRS complex is amplified.

Figure 9: Band-pass filtered ECG after subjecting to derivative filtering and squaring function

Figure 10 shows the output of moving window integral stage. This signal is noise free and has waveform similar to rectangular wave which peaks when the QRS complex occur.
CONCLUSION

An R-peak detection algorithm was developed in SAS® by implementing difference equations in matrix-based data manipulation routines and running them in SAS® using the IML procedure. The R-peaks were successfully detected by the algorithm when run on sample data from the MIT-BIH Arrhythmia database.
The focus of this paper is to demonstrate that analysis of ECG signals and other physiological signals can be successfully performed using SAS® with relative ease. This approach can be extended to develop automated and semi-automated algorithms to process images used for tumor and plaque quantification in oncology and cardiovascular trials. Using SAS®/IML for signal analysis and detection can be very useful in scenarios where statistical analysis needs to be performed on a multitude of parameters derived from the signal of interest. Using only SAS® can help researchers focus on analysis and avoid data movement between multiple tools i.e. one for detection of features (E.g., MATLAB®) and another for statistical analysis (SAS®). Using only SAS® also helps to constrain costs.

Hopefully this paper inspires further quantitative analysis to help compare performance and efficiency issues of SAS® and MATLAB® or procedural languages such as C in scenarios involving signal analysis.

In clinical trials, ECG data is already preprocessed. Maybe with SAS® and raw data, it allows for enhanced exploration and analysis. The macro can calculate R-R interval, heart rate variability, and instantaneous heart rate from ECG data.

ACKNOWLEDGEMENTS

We would like to thank Mary Varughese, Associate Director, Merck & Co., Inc. for her encouragement and support. We also would like to thank Gopal Rajagopal of Merck & Co., Inc. for his feedback. We would also like to thank Dr. Khosrow Behbehani of UT Arlington for his guidance and valuable inputs on ECG signal processing. We would also like to thank all the reviewers in respective companies for valuable inputs.

REFERENCES


CONTACT INFORMATION

Author Name: Suhas R. Sanjee Author Name: Sridhar Vijendra
Title : Scientific Programming Analyst Company : MaxisIT Inc.
Company : Merck & Co., Inc. Address : 351 Sumneytown Pike Address : 203 Main St
City : North Wales State: PA ZIP: 19454 City : Metuchen State: NJ ZIP: 08840
Email : suhas_sanjee@merck.com Email : svijendra@maxisit.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. © indicates USA registration.
Other brand and product names are trademarks of their respective companies.