CGM-Freq: A Python Library for Frequency Domain Analysis of Continuous Glucose Monitoring Data

Elizabeth Healey and Isaac Kohane

***Abstract*—In recent years, the number of patients using continuous glucose monitoring (CGM) has increased. In addition to helping patients manage their disease, CGM produces time series data that can be used for integration in control algorithms, predictive models, and for retrospective analyses. Through feature extraction, many digital biomarkers can be derived from CGM. In this work, we provide a tool to extract features derived from the frequency domain. We first introduce a novel open-source Python library, CGM-Freq, for the analysis of CGM data in the frequency domain. We then test the library on real data. This work provides an open-source tool to further investigate the frequency domain of CGM signals.**

***Index Terms*—signal processing, continuous glucose monitoring**

I. Introduction

Continuous glucose monitors (CGM) are wearable devices that sample interstitial glucose approximately every 5 minutes and assist individuals in managing their glucose levels. CGM use has increased over recent years in individuals with type 1 diabetes (T1D) and type 2 diabetes (T2D) [1]. Increasingly, there has been interest in analyzing retrospective CGM data to improve clinical outcomes through building predictive models and identifying associations between CGM features and clinical outcomes [2]–[4]. Much of this work has leveraged feature extraction from raw CGM data.

While there have been many recent advances in computational tools for analyzing CGM data [5], there has been limited work focusing specifically on frequency-domain analysis of CGM data. In this work, we present a new open-source tool in Python that can be used when analyzing CGM data in the frequency domain. We discuss the components of the library and test it on open-source data.

Our primary contributions are:

1) We provide an easy-to-use tool to visualize CGM signals in the frequency domain using advanced signal processing techniques.
2) We provide a library with functions for the generation of tabular digital biomarkers from the CGM data. The feature generation from time series data includes features derived from the frequency domain and features derived from the time domain. These features can be used when building predictive models or analyzing retrospective data.
3) We test the library on publicly available CGM data. Through this analysis, we confirm the potential of features from the frequency domain to discriminate different disease states.

Elizabeth Healey is with the Harvard-MIT Program in Health Sciences and Technology, Cambridge, MA, USA `ehealey@mit.edu`. Isaac Kohane is with the Department of Biomedical Informatics, Harvard Medical School, Boston, MA, USA. This work was supported by the National Science Foundation Graduate Research Fellowship program under grant 2141064.
The Python library is currently available online at https://github.com/lizhealey/CGM-Freq.

II. Related Work

A. Existing packages to analyze CGM data

Over the past decade, there have been many different computational tools developed for the analysis of CGM data [6]–[13]. These tools exist in multiple programming languages including R, Python, MATLAB, and through online graphical user interfaces (GUI), and they provide plots and feature derivation from raw CGM data. One of the limitations of the currently available tools is that they do not include a comprehensive frequency domain analysis.

B. Previous work supporting frequency domain analysis of CGM

There has been recent interest in analyzing the frequency spectrum of CGM signals. Fourier analysis of CGM has been explored in recent literature, and the results of which have suggested that there may be insights to be gained from analyzing the frequency domain of CGM signals [14]–[17]. In Fico et al., the authors analyzed the CGM signals from individuals and showed that frequency-domain features were associated with the different diagnoses of T2D, T1D, and individuals at risk for T2D. Features from the frequency domain have also been incorporated in predictive models [18], further supporting the rationale for developing more analysis tools.

Further, in the broader landscape of physiological signal processing, there has been demonstrated interest in characterizing the frequency spectrum of biological signals, and computational resources exist for this purpose [19], [20]. However, these tools are not specialized for CGM data.

III. Design and Implementation

A. Overview

CGM-Freq is a library for analyzing CGM features in the frequency domain. The toolbox takes in raw CGM input from a file with required headers for columns representing the time and the CGM values.

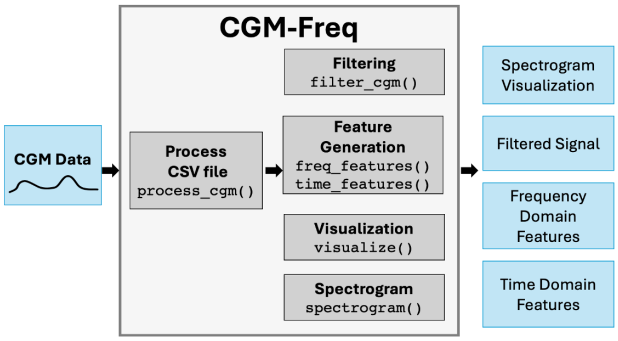

Fig. 1. Overview of the CGM-Freq library

.

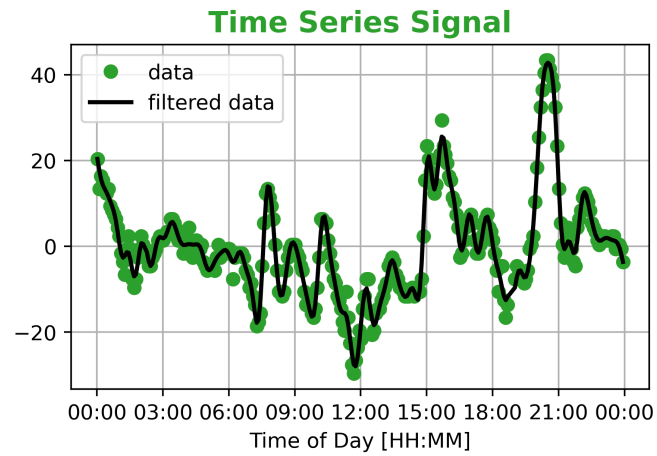

Fig. 2. Filtered CGM signal. This plot shows the output of the filter function with the filtered signal in black

.

The library has four primary modules:

1) **Filtering**. This module takes in the CGM signal and applies a low-pass filter based on a cutoff frequency that can be adjusted by the user. It returns and plots the filtered signal.
2) **Feature Generation**. This module has two parts. The first part computes frequency-domain features and the second part computes basic time-domain features.
3) **Visualization**. This module allows for the visualization of frequency domain of the signal and plots the time series signal.
4) **Spectrogram**. This module returns the output of the spectrogram and plots the spectrogram along with the time series signal.

Fig. 1 gives an overview of the system and the functions for each module.

B. Preprocessing

The CGM signal first must be preprocessed using a function called *process_cgm()*. The output of this function is two arrays: an array of the CGM values and an array of the time values. A warning is also printed making note of how many CGM readings are estimated to be missing based on the sampling frequency. In the current version of our library, we do not impute missing data.

C. Filtering

The CGM data is filtered using a low-pass filter. The input to the function includes the time and CGM values and the cutoff frequency, which can be adjusted. In this function, a filtered CGM signal is returned and plots are produced for visualization. Fig. 2 shows the visualization that is produced by this function. The raw CGM data is plotted with the filtered data.

D. Feature Generation

Features are generated using two different functions. Table I shows the features generated and their definitions. The feature generation functions are split by frequency domain features and time domain features.

The first function generates features from the frequency domain. These features are derived based on previous work [17] which extracts frequency-domain features from the signal using advanced signal processing techniques [21], [22]. In this function, the Discrete Fourier Transform is computed using the fast Fourier transform (FFT) algorithm. Additionally, the power spectral density (PSD) is generated using the Welch method [23]. Here, the window type and the number of samples per segment are input variables that can be modified to change the frequency resolution of the PSD.

The second function captures the time-domain features of the signal. We draw from metrics described in literature for CGM interpretation [24]. Since the purpose of this package is frequency analysis of CGM signals, we do not include a comprehensive list of time-domain features that have been previously reported. Both functions produce tabular data, enabling a quick transformation of the time series data for downstream tasks.

E. Visualization

The *visualize()* function takes in raw CGM data and returns a plot with three components. Fig. 3 shows the output of the function. The plots contain the FFT, the PSD, and the raw CGM data. The function also takes in the sampling frequency and optionally takes in additionally arguments for the PSD. The PSD is obtained using the *welch()* from SciPy [25]. While only 24 hours of data are shown, the function is capable of plotting longer time periods of time.

F. Spectrogram

This part of the toolbox produces both a visual output and numerical outputs of the spectrogram transformation from SciPy [25]. In Fig. 4, the visual output of the spectrogram function is shown. The function also returns the raw output of the spectrogram transformation, for analysis of the power spectrum as a function of time at different frequencies. The

TABLE I
FEATURE LIST

Frequency Domain Features	
Discrete Fourier Transform	
Feature	**Definition**
max_amplitude	The maximum magnitude of the FFT [17]
fft_dominant_frequency	The frequency where the maximum of the FFT occurs [17]
fft75_frequency	The frequency where 75% of the FFT signal is contained [17]
fft_peak2_frequency	The frequency where the second highest peak of FFT occurs [17]
fft_peak2_mag	The magnitude of the FFT where the second highest peak of the FFT occurs [17]
Power Spectral Density	
Feature	**Definition**
psd_max_amplitude	The maximum value of PSD [17]
psd_dominant_frequency	The frequency where the maximum value of PSD occurs [17]
bandwidth	The 3dB frequency bandwidth [17]
psd75_frequency	The frequency where 75% of the PSD is contained [17]
Time Domain Features	
Feature	**Definition**
Mean	Mean glucose in the segment
Minimum	Minimum glucose value in the segment
Maximum	Maximum glucose value in the segment
TIR	The percent of readings in the range 70mg/dL to 180mg/dL in the segment
TAR 1 (>180)	The percent of readings above 180 mg/dL in the segment
TAR 2 (>250)	The percent of readings above 250 mg/dL in the segment
TBR 1 (<70)	The percent of readings below 70 mg/dL in the segment
TBR 2 (<54)	The percent of readings below 54 mg/dL in the segment
Std	The standard deviation of glucose readings in the segment
CV	The coefficient of variation of glucose readings in the segment
	The features and definitions for the frequency domain features are drawn from definitions in [17] and the features and definitions for the time domain features are drawn from [24]

visual output can be used to identify temporal trends in the frequency domain of the signal.

G. Methods

We leverage several existing Python libraries including NumPy [26], SciPy [25], and Matplotlib [27]. Specifically, we leverage the *find_peaks()* function from SciPy. The full list of dependencies for the library are listed in the Github repository. The publicly available data used in this work was made available under a Creative Commons Attribution License [28].

IV. Demonstration on Real Data

A. Testing Dataset

We used a public dataset from Hall et al. [28] that has CGM data from individuals with T2D, with pre-diabetes, and with no diabetes. We used CGM-Freq to extract features from daily CGM data for each individual in the dataset. When analyzing the data, we included only CGM data where there were greater than 280 readings in the day.

B. Frequency domain features on real data

To analyze the features, we plotted the mean and standard deviation of the features by diagnosis group. Fig. 5 shows a visual representation of the separation of means for each of the features by diagnosis group. Many of the time-domain features and frequency-domain features had different mean values for each diagnosis group. As expected, the mean average glucose was higher in individual with diabetes than patients without diabetes.

In Table II, the mean and standard deviation of each feature is shown for each diagnosis group. The total number of individuals and CGM days is also shown in the table. A t-test was performed to assess significance in differences in the mean between the ”No diabetes” group and the ”pre-diabetes” group, and between the ”No diabetes” group and the ”diabetes” group. Many of the features had statistically significant different means. These results align with previous findings from Fico et al, [17] which found statistically significant differences in values of many frequency domain features, such as the maximum peak of the PSD, between groups of individuals with T2D and individuals at risk of diabetes.

We further analyzed the CGM features across the individuals to understand the heterogeneity. We took the mean value of each CGM feature for each individual and plotted the absolute value of the correlation coefficients of the features. Fig. 6 shows how the CGM features are correlated with one another. Both the fft75_frequency and psd75_frequency have correlation coefficients less than .7 with all time-domain features, but statistical significance in distinguishing groups

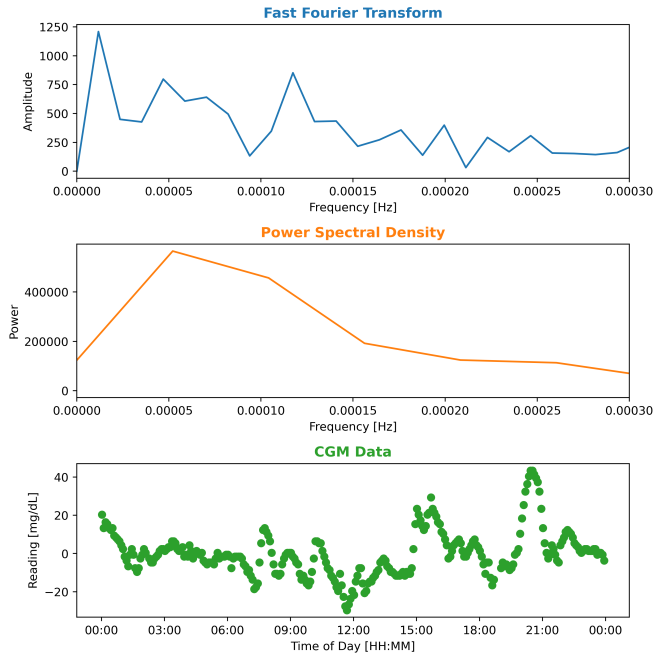

Fig. 3. Visualization output. These plots show the FFT (top), PSD (middle) and raw CGM data (bottom) for 24 hours of sample data.

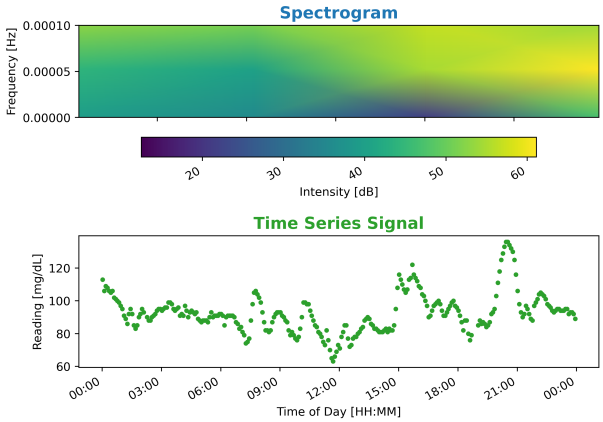

Fig. 4. Spectrogram of CGM data. These plots show a sample of 24 hours of CGM data with the spectrogram (top) and the raw time series data (bottom). The x axis is time of day for both plots.

based on diagnosis. This suggests that the frequency domain features may provide unique insight into glycemic dynamics.

V. Discussion

In this work, we introduce the first version of a Python library for generation of digital biomarkers from CGM signals in the frequency domain. In our library, we have four modules: visualization, filtering, feature generation, and spectrogram generation. This library serves as an open-source tool for analysis of CGM data in the frequency domain.

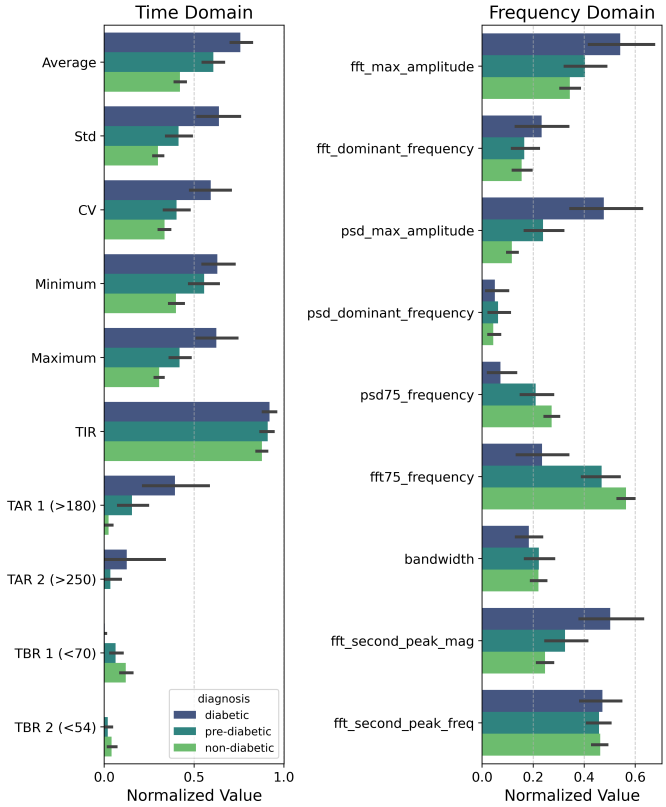

Fig. 5. CGM features by diagnosis. The values for each feature are normalized within each feature for demonstrative purposes. The mean is shown, with error bars indicating the standard deviation.

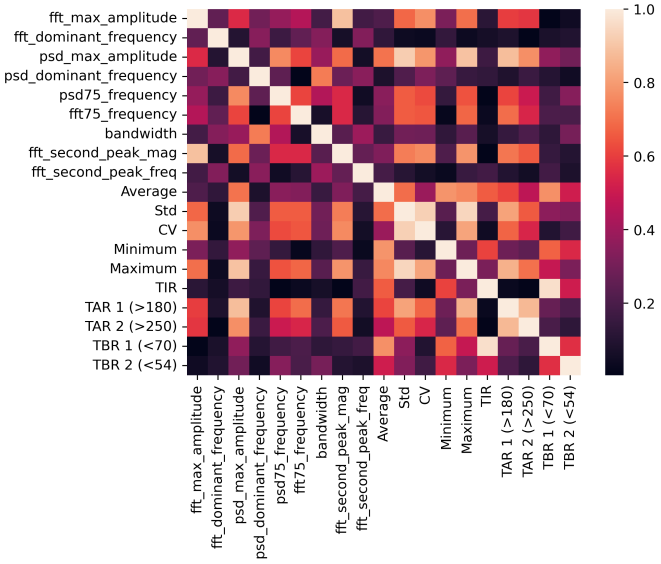

Fig. 6. Heatmap showing the absolute value of the correlation coefficients of the features across all individuals. For individuals with data from multiple days, we take the mean feature value.

The library was tested on a public dataset of patients with diabetes, pre-diabetes, and no diabetes. The results of the testing were presented in the paper, confirming the discrimi-

TABLE II
FEATURE VALUES AND SIGNIFICANCE

	No diabetes	Pre-diabetes	Diabetes
Number of individuals in category	25	10	2
Number of days included	132	46	12
Time Domain Features			
Average	96.30 (10.31)	107.96 (13.08)*	117.35 (7.15)*
CV	0.18 (0.05)	0.19 (0.06)*	0.25 (0.06)*
Maximum	154.98 (27.12)	179.31 (41.01)*	222.92 (43.73)*
Minimum	61.77 (10.94)	69.93 (14.95)*	73.75 (8.44)*
Std	17.10 (5.29)	21.11 (8.09)*	28.95 (7.52)*
TAR 1 (>180)	0.00 (0.01)	0.02 (0.03)*	0.05 (0.04)*
TAR 2 (>250)	0.00 (0.00)	0.00 (0.01)*	0.01 (0.01)*
TBR 1 (<70)	0.06 (0.09)	0.03 (0.06)*	0.00 (0.00)*
TBR 2 (<54)	0.00 (0.01)	0.00 (0.01)	0.00 (0.00)
TIR	0.93 (0.11)	0.94 (0.07)	0.95 (0.04)
PSD Features			
bandwidth	3.8e-05 (2.7e-05)	3.9e-05 (3.2e-05)	3.2e-05 (1.8e-05)
psd75_frequency	1.4e-04 (3.8e-05)	1.2e-04 (5.4e-05)*	8.7e-05 (2.3e-05)*
psd_dominant_frequency	4.2e-05 (2.0e-05)	4.6e-05 (2.4e-05)	4.3e-05 (1.6e-05)
psd_max_amplitude	2.0e+06 (1.8e+06)	4.0e+06 (4.0e+06)*	7.8e+06 (4.2e+06)*
FFT Features			
fft75_frequency	3.7e-04 (7.3e-05)	3.3e-04 (1.0e-04)*	2.3e-04 (7.2e-05)*
fft_dominant_frequency	1.7e-05 (1.9e-05)	1.8e-05 (1.6e-05)	2.4e-05 (1.7e-05)
fft_max_amplitute	3.3e+03 (1.8e+03)	3.8e+03 (2.3e+03)	5.0e+03 (2.0e+03)*
fft_second_peak_freq	1.4e-05 (4.0e-05)	1.3e-05 (4.0e-05)	1.7e-05 (3.8e-05)
fft_second_peak_mag	2.5e+03 (1.4e+03)	3.2e+03 (2.3e+03)*	4.6e+03 (1.9e+03)*
** indicates p<.05 in t-test with "no diabetes" group*			

native power of multiple frequency-domain features, which is consistent with previous literature [17].

The data this was tested on contained only a small number of individuals with limited heterogeneity in disease phenotype. We expect that the discriminative power of the features may be different in a larger cohort. Our library provides a tool to include frequency-domain features in future analyses of CGM data. Below we highlight some potential use cases of our library:

1) Predictive models: Predictive models from time series data are often built using extracted features in tabular form. This library provides an easy-to-use feature extraction function to quickly transform raw CGM data into features that can be used to build predictive models.
2) Anomaly detection: By studying how frequency components change over time, it may be possible to identify anomalies in the data.
3) Unsupervised clustering: Digital biomarkers that characterize CGM signals in the frequency domain can be included in clustering approaches to identify novel phenotypes.

CGM-Freq is a library for researchers to quickly transform CGM data into time and frequency domain features in Python. Our library serves as an additional computational resource that investigators can use when analyzing CGM data. The toolbox will continue to be updated and expanded to include more features as research in this space progresses.

REFERENCES

[1] N. Reddy, N. Verma, and K. Dungan, "Monitoring technologies- continuous glucose monitoring, mobile technology, biomarkers of glycemic control," in *Endotext*, K. R. Feingold, B. Anawalt, M. R. Blackman, A. Boyce, G. Chrousos, E. Corpas, W. W. de Herder, K. Dhatariya, K. Dungan, J. Hofland, S. Kalra, G. Kaltsas, N. Kapoor, C. Koch, P. Kopp, M. Korbonits, C. S. Kovacs, W. Kuohung, B. Laferrère, M. Levy, E. A. McGee, R. McLachlan, M. New, J. Purnell, R. Sahay, A. S. Shah, F. Singer, M. A. Sperling, C. A. Stratakis, D. L. Trence, and D. P. Wilson, Eds. South Dartmouth (MA): MDText.com, Inc., Jul. 2023.

[2] R. Holzer, W. Bloch, and C. Brinkmann, "Continuous glucose monitoring in healthy Adults—Possible applications in health care, wellness, and sports," *Sensors*, vol. 22, no. 5, p. 2030, Mar. 2022.

[3] X. Tang, S. Li, Y. Wang, M. Wang, Q. Yin, P. Mu, S. Lin, X. Qian, X. Ye, and Y. Chen, "Glycemic variability evaluated by continuous glucose monitoring system is associated with the 10-y cardiovascular risk of diabetic patients with well-controlled HbA1c," *Clin. Chim. Acta*, vol. 461, pp. 146–150, Oct. 2016.

[4] E. Longato, G. Acciaroli, A. Facchinetti, L. Hakaste, T. Tuomi, A. Maran, and G. Sparacino, "Glycaemic variability-based classification of impaired glucose tolerance vs. type 2 diabetes using continuous glucose monitoring data," *Comput. Biol. Med.*, vol. 96, pp. 141–146, May 2018.

[5] M. T. Olsen, C. K. Klarskov, A. M. Dungu, K. B. Hansen, U. Pedersen-Bjergaard, and P. L. Kristensen, "Statistical packages and algorithms for the analysis of continuous glucose monitoring data: A systematic review," *J. Diabetes Sci. Technol.*, p. 19322968231221803, Jan. 2024.

[6] I. Attaye, E. W. J. van der Vossen, D. N. Mendes Bastos, M. Nieuwdorp, and E. Levin, "Introducing the continuous glucose data analysis (CGDA) R package: An intuitive package to analyze continuous glucose monitoring data," *J. Diabetes Sci. Technol.*, vol. 16, no. 3, pp. 783–785, May 2022.

[7] E. Chun, N. Fernandes, and I. Gaynanova, "An update on the iglu software for interpreting continuous glucose monitoring data," *Diabetes Technol. Ther.*, Jun. 2024.

[8] E. Olawsky, Y. Zhang, L. E. Eberly, E. S. Helgeson, and L. S. Chow, "A new analysis tool for continuous glucose monitor data," *J. Diabetes Sci. Technol.*, vol. 16, no. 6, pp. 1496–1504, Nov. 2022.

[9] T. Vigers, C. L. Chan, J. Snell-Bergeon, P. Bjornstad, P. S. Zeitler, G. Forlenza, and L. Pyle, "cgmanalysis: An R package for descriptive analysis of continuous glucose monitor data," *PLoS One*, vol. 14, no. 10, p. e0216851, Oct. 2019.

[10] G. Cappon, G. Sparacino, and A. Facchinetti, "AGATA: A toolbox for automated glucose data analysis," *J. Diabetes Sci. Technol.*, p. 19322968221147570, Jan. 2023.

[11] J. Shao, Z. Liu, S. Li, B. Wu, Z. Nie, Y. Li, and K. Zhou, "Continuous glucose monitoring time series data analysis: A time series analysis package for continuous glucose monitoring data," *J. Comput. Biol.*, vol. 30, no. 1, pp. 112–116, Jan. 2023.

[12] J. Chrzanowski, S. Grabia, A. Michalak, A. Wielgus, J. Wykrota, B. Mianowska, A. Szadkowska, and W. Fendler, "GlyCulator 3.0: A fast, Easy-to-Use analytical tool for CGM data analysis, aggregation, center benchmarking, and data sharing," *Diabetes Care*, vol. 46, no. 1, pp. e3–e5, Jan. 2023.

[13] A. Piersanti, F. Giurato, C. Göbl, L. Burattini, A. Tura, and M. Morettini, "Software packages and tools for the analysis of continuous glucose monitoring data," *Diabetes Technol. Ther.*, vol. 25, no. 1, pp. 69–85, Jan. 2023.

[14] M. D. Breton, D. P. Shields, and B. P. Kovatchev, "Optimum subcutaneous glucose sampling and fourier analysis of continuous glucose monitors," *J. Diabetes Sci. Technol.*, vol. 2, no. 3, pp. 495–500, May 2008.

[15] D. A. Gough, K. Kreutz-Delgado, and T. M. Bremer, "Frequency characterization of blood glucose dynamics," *Ann. Biomed. Eng.*, vol. 31, no. 1, pp. 91–97, Jan. 2003.

[16] M. Miller and P. Strange, "Use of fourier models for analysis and interpretation of continuous glucose monitoring glucose profiles," *J. Diabetes Sci. Technol.*, vol. 1, no. 5, pp. 630–638, Sep. 2007.

[17] G. Fico, L. Hernández, J. Cancela, M. M. Isabel, A. Facchinetti, C. Fabris, R. Gabriel, C. Cobelli, and M. T. Arredondo Waldmeyer, "Exploring the frequency domain of continuous glucose monitoring signals to improve characterization of glucose variability and of diabetic profiles," *J. Diabetes Sci. Technol.*, vol. 11, no. 4, pp. 773–779, Jul. 2017.

[18] A. Guemes, G. Cappon, B. Hernandez, M. Reddy, N. Oliver, P. Georgiou, and P. Herrero, "Predicting quality of overnight glycaemic control in type 1 diabetes using binary classifiers," *IEEE J. Biomed. Health Inform.*, vol. 24, no. 5, pp. 1439–1446, May 2020.

[19] O. O. Nacitarhan and B. Semiz, "PySio: A new python toolbox for physiological signal visualization and feature analysis," *Conf. Proc. IEEE Eng. Med. Biol. Soc.*, vol. 2022, pp. 3686–3689, Jul. 2022.

[20] P. Bota, R. Silva, C. Carreiras, A. Fred, and H. P. da Silva, "BioSPPy: A python toolbox for physiological signal processing," *SoftwareX*, vol. 26, p. 101712, May 2024.

[21] U. Lee, "Spectral analysis of signals," pp. 11–38, Sep. 2009.

[22] B. Champagne and F. Labeau, "Discrete time signal processing," 2004.

[23] P. Welch, "The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Trans. Audio Electroacoust.*, vol. 15, no. 2, pp. 70–73, Jun. 1967.

[24] T. Battelino, T. Danne, R. M. Bergenstal, S. A. Amiel, R. Beck, T. Biester, E. Bosi, B. A. Buckingham, W. T. Cefalu, K. L. Close, C. Cobelli, E. Dassau, J. H. DeVries, K. C. Donaghue, K. Dovc, F. J. Doyle, 3rd, S. Garg, G. Grunberger, S. Heller, L. Heinemann, I. B. Hirsch, R. Hovorka, W. Jia, O. Kordonouri, B. Kovatchev, A. Kowalski, L. Laffel, B. Levine, A. Mayorov, C. Mathieu, H. R. Murphy, R. Nimri, K. Nørgaard, C. G. Parkin, E. Renard, D. Rodbard, B. Saboo, D. Schatz, K. Stoner, T. Urakami, S. A. Weinzimer, and M. Phillip, "Clinical targets for continuous glucose monitoring data interpretation: Recommendations from the international consensus on time in range," *Diabetes Care*, vol. 42, no. 8, pp. 1593–1603, Aug. 2019.

[25] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[26] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[27] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020. [Online]. Available: https://doi.org/10.1038/s41586-020-2649-2

[28] H. Hall, D. Perelman, A. Breschi, P. Limcaoco, R. Kellogg, T. McLaughlin, and M. Snyder, "Glucotypes reveal new patterns of glucose dysregulation," *PLoS Biol.*, vol. 16, no. 7, p. e2005143, Jul. 2018.