EMF Sensor Data Processing

Here, we outline our strategy for signal processing to analyze trends in the data obtained from EMF sensors. Our approach focuses on processing the raw sensor data efficiently to extract meaningful insights.

Processing Steps:

1. Load Data: Begin by loading the .tdms file and converting it into a pandas DataFrame object.
2. Time Correction: For the second project, we corrected the time data from 10:00 to 12:00 to align with actual observation times.
3. Convert Time: We convert the time data to seconds from midnight to standardize and simplify subsequent calculations.
4. Resample Data: Resample the sensor data to 1-second intervals using pandas' resample function, applying the mean to aggregate the data. The resampling function reduces the data's granularity to a manageable level while preserving its temporal structure. This step involves averaging the data points within each 1-second interval, effectively smoothing some of the short-term fluctuations.
5. Decompose Data: Utilize the seasonal\_decompose function from the statsmodels library, configured with an additive model and a period of 3600 (seconds), to decompose the sensor data into trend and seasonality components.
6. Cross-Corelation Techniqe: Calculatie the cross-correlation between the two time series of the glucose and the sensor trend extracted from step 5, using numpy correlate function. We first normalizes the both time series, and then identify the lag time that results in the strongest negative correlation. This analysis is based on the assumption that the EMF trend is expected to mirror the opposite pattern of glucose levels (hyperglycemia or hypoglycemia) and precede these changes.

Detailed Seasonal\_decompose Function Explanations:

Resample Function in Pandas: This function transforms the time series data into a consistent time frame for analysis. By specifying a frequency (e.g., '60S' for 60 seconds), the resample function groups the data into these intervals and applies a specified aggregation method, in this case, the mean. This technique not only simplifies the data but also helps in identifying overarching patterns by averaging out the data points within each interval.

Seasonal Decompose Function from Statsmodels: The seasonal\_decompose function decomposes a time series into its constituent components: trend, seasonal, and residual. This decomposition allows for a clearer understanding of different influences on the data:

Trend: Shows the long-term progression, highlighting how the data evolves.

Seasonal: Captures regular, periodic fluctuations influenced by seasonal factors.

Residual: Represents the noise or random variations left after accounting for the trend and seasonal effects.

Additive Model

In the additive model, the observed time series data, is assumed to be the sum of three components:

Trend component: This captures the underlying trend of the dataset. It is generally smoothed out using methods like moving averages.

Seasonal component: This represents the repeating short-term cycle in the data. It's typically constant through series and computed by averaging the data across the same period (like day of week, month, etc.).

Residual component: These are the irregularities that are not explained by the trend or seasonal components. It's what remains after the trend and seasonal components have been subtracted from the original data.