Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

PhysioMerge: A tool for physiological data extraction and analysis

Taha Abbasi-Hashemi

abbasiht@mcmaster.ca

Department of Kinesiology, Faculty of Science, McMaster University https://orcid.org/0000-0002-6050-4492

Baraa K. Al-Khazraji

Department of Kinesiology, Faculty of Science, McMaster University https://orcid.org/0000-0003-1075-4183

Research Article

Keywords: waveform morphology, data analysis, waveform analysis, data validation

Posted Date: September 30th, 2025

DOI: https://doi.org/10.21203/rs.3.rs-7707776/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations:
The authors declare no competing interests.

I state that ethical approval was obtained for human participants from McMaster HiREB (The mcmaster ethics board). I state that consent was obtained from all human participants.

PhysioMerge: A tool for physiological data extraction and analysis

Taha Abbasi-Hashemi[1], Baraa K. Al-Khazraji[1]

September 2025

[1] Department of Kinesiology, Faculty of Science, McMaster University

Emails: abbasiht@mcmaster.ca (Taha Abbasi-Hashemi)*, alkhazrb@mcmaster.ca (Baraa K. Al-Khazraji)

* corresponding author

1 Abstract

Physiological research relies on accurate data extraction from recording systems such as LabChart. While these systems provide essential tools for data collection, processing large and complex datasets can remain time consuming and often requires manual validation of breaths or cardiac cycles. Manual validation introduces variability due to human judgment, fatigue, and the inability to reprocess large datasets efficiently. To streamline this process, we present PhysioMerge, an automated tool for robust physiological data extraction. PhysioMerge allows users to implement customizable checks to validate signals against predefined criteria, flag anomalies, and preserve valid segments, reducing the need for subjective manual assessment. The software also supports advanced waveform morphology analysis, extracting features such as peak prominence, curvature, and slopes that complement conventional analyses. Flexible data segmentation by time, intervention, or study phase, combined with batch processing capabilities, allows researchers to efficiently handle large datasets and perform consistent, high-precision analyses.

2 Introduction

Data extraction from physiological data acquisition/analysis systems or software (e.g. LabChart, ADInstruments) is a critical but time-consuming task in physiological research. Some software does not automatically assess whether breaths or cardiac cycles are physiologically valid (whether the signal is physiologically acceptable, and free of dropouts or artifacts). This means that data analysis that requires signal validity requires manual extraction. Manual analysis introduces variability due to human fatigue, subjective judgment, and the impracticality of reprocessing large datasets for minor adjustments (revisions, reevaluation following methodology changes). While software like LabChart provide built-in tools for data export, these often lack flexibility. For example, LabChart only supports extraction of a single continuous strip across all

channels within a time bin. Forcing researchers to discard valid segments of data when errors (e.g., artifact-contaminated blood pressure) occur in a single channel, even if other channels (e.g., blood velocity) remain usable. While LabChart does allow extraction of individual channels, this approach still creates challenges when working with multichannel physiological data. For instance, when an artifact contaminates the blood pressure signal but not blood velocity signal, extracting only the velocity channel removes the ability to align and analyze those signals in context. In practice, this forces researchers to either (i) discard valid segments from other channels, or (ii) perform time-intensive manual synchronization across channels after selective extraction. Both outcomes reduce efficiency. This "all-or-nothing" approach compromises both efficiency and precision. Automation of data extraction from these software is important as it requires minimal manual effort, removes human variability, and reduces computation time.

When analyzing physiological signals, determining signal validity can be challenging, particularly for less experienced evaluators. In many studies, signal validation is performed by experts or trained professionals who rely on experience and intuition [1], [2]. While this approach can be effective, it is also time-consuming, costly, and inherently inconsistent. Assessments may vary both between and within individuals, influenced by factors such as fatigue, mood, or familiarity with the data [3]. Such variability reduces the reliability and reproducibility of results.

Extracting data from software like LabChart can be sped up with built-in macros, but it still requires continuous blocks of clean data, and does not allow for batch processing. Without batch processing each file must be processed independently. Furthermore, many inexperienced staff members may not be aware of these capabilities or may struggle to use them effectively. Without these tools, extracting data for a large number of participants can take several hours or even days. With the right setup, however, this process can be reduced to just a few hours. The purpose of this tool is to enable parallelization and rapid processing of data, allowing for much more efficient and scalable analysis, especially when working with large datasets and batch processing files reducing hours to minutes.

Waveform morphology refers to the detailed shape of physiological signals, which can reveal information about vascular compliance, autonomic regulation, and other underlying physiological processes [4]–[6]. Many data extraction software do not currently support extensive morphological measurements. We seek to develop a tool that overcomes these limitations by providing direct access to the waveform properties, enabling researchers to extract and analyze subtle morphological variations that correlate with specific physiological states or pathologies [4], and providing additional information beyond current time-averaged analysis approaches. This capability supports advanced applications such as early disease detection [7], treatment monitoring [8], and deeper investigation of cardiovascular dynamics [6].

Herein we present an automated method for extracting physiological data, "PhysioMerge", which addresses these issues by incorporating built-in checks to ensure that the selected signals fall within the predefined bounds and meet the necessary *a priori*-defined quality standards. These checks help identify and flag any anomalies or outliers in the data, preventing faulty signals from being included in the analysis.

PhysioMerge also allows for extraction or waveform morphological data in addition to standard time-domain measures. PhysioMerge allows for consistent, rapid, and accurate extraction of data from LabChart and other physiological data recording tools. It allows single sessions to be split into time based groups, intervention groups, and studies. Unlike humans, PhysioMerge does not suffer from fatigue or subjectivity and data processing is consistent, reliable and fast allowing it to batch process long duration experimental recordings without any manual intervention. PhysioMerge is capable of verbose debugging allowing researchers to know where and what needs manual intervention when it is required. Once the signal and waveform metrics have been cleaned to remove non-valid data, they are processed and exported into one or more tables for evaluation in statistics, machine learning (ML) applications, or clinical trial analyses.

3 Methods

PhysioMerge is designed to operate through configuration files. The configuration files are written in Tom's Obvious, Minimal Language (TOML). PhysioMerge runs through a list of configuration files in the order they are provided. Using configuration file(s), makes sure the data extraction processs is flexible and can be customized based on the analytical needs of the study. The configuration file defines the parameters for how the data from the study should be extracted, providing detailed instructions on the extraction process. By using TOML, the configuration is both human-readable and easily editable, ensuring that researchers can quickly adjust settings to fit the requirements of different datasets or experimental setups.

PhysioMerge operates on a modular system, where each "command" or action is encapsulated within its own module. These modules each have their own configuration patterns and specific functionalities, allowing users to select, configure, and execute various commands based on the needs of the study. For example, one module might handle the extraction of signal data, while another could be responsible for applying specific filtering methods or performing waveform analysis. The modular design provides scalability and flexibility, enabling researchers to tailor PhysioMerge to a wide range of data extraction tasks.

Each module can be independently configured using parameters defined in the TOML file, providing a straightforward way to adapt PhysioMerge to different data sources, formats, or analysis methods. This system also allows for easy extension of PhysioMerge, as new modules can be added to support additional commands or functionality. Ultimately, this modular approach ensures that PhysioMerge remains adaptable, efficient, and able to accommodate a broad spectrum of research needs.

The output generated by each module can be stored in named variables, allowing results to be reused or referenced in subsequent operations. For example, if a dataset contains three distinct interventions, each one can be isolated and assigned to its own variable, such as *intervention1*, *intervention2*, and *intervention3*. This adds a layer of customization and control, enabling researchers to tailor their analysis workflows directly within the configuration file. In effect, the configuration file becomes a lightweight domain-specific language, empowering users to structure complex parsing logic in a concise and readable format.

3.1 Data input and output

3.1.1 Read Command

In many traditional physiological data processing workflows, researchers are required to process files individually, often one at a time. However, PhysioMerge significantly improves this process by allowing for batch processing, where multiple files can be read and processed simultaneously. The `read` command plays a pivotal role in this functionality, as it specifies the location of the files and initiates the reading process. Although it currently supports text-based file formats, it is designed with extensibility in mind and can be easily upgraded to accommodate additional file types in the future. This ensures that PhysioMerge can adapt to evolving data formats without requiring major reconfigurations. As a result, researchers can use the same pipeline for a wide range of data sources. Additionally, it is not limited to just reading LabChart files [9]. It currently supports reading duplex Doppler ultrasound readings [10], and RespirAct End tidal gas clamping output files [11]. The output for each of these devices can be combined together and processed together.

3.1.2 Save Command

The `save` function provides flexible data export options, allowing users to store processed physiological data in a variety of file formats (.csv, .tsv, .txt, etc.). Users can specify the delimiter (comma, tab, semicolon, etc.) to ensure compatibility with analysis tools. Additionally, selective variable export is supported, enabling researchers to save only the relevant parameters for their specific analysis, thereby optimizing storage efficiency and reducing file clutter.

To enhance usability, the function includes options for header inclusion allowing for the embedding of variable names and units. Furthermore, the `save` command offers file handling control, permitting users to either overwrite existing files or append new data to them. This feature is particularly useful for longitudinal studies or iterative analyses, where incremental data updates are required without the need to manually merge datasets.

3.2 Analysis Of Data Commands

3.2.1 Check Command

The `check` command is used to validate the integrity or quality of data before further processing. This step is essential in automated pipelines, where ensuring that only clean, reliable data is passed downstream can prevent errors and improve the overall robustness of the analysis. A key strength of the `check` command lies in its flexibility. Researchers can define customized validation rules based on the specific requirements of their study. Since individual checks produce binary mask as a named variable output which can then be used as an input for other commands such as the `take`, `aggregate`, `save`, `merge` command. In the mask a

`1` represents a valid segment/cycle and `0` represents an invalid one. Furthermore, logical operations can be used to combine these results in the `merge` command.

A range of user-defined checks is supported, each implemented as a function that evaluates a data segment (a cardiac cycle, respiratory cycle, or time period) against a numerical threshold. These checks can be used to filter data based on both amplitude (Y-axis) and length or position (X-axis). For example, users can ensure that the maximum or minimum value of a segment remains within specified physiologically-relevant Y-axis bounds, or that the overall difference between the highest and lowest values does not exceed a certain threshold (e.g. systolic blood pressure between 80-150 mmHg). Similarly, checks can be applied to the segment's width. Specifically, one can ensure that data fall within an expected duration or sample count range (e.g. how long/short was the respiratory cycle or sit-stand protocol). Checks like this can ensure that all physiological measurements lie within a given range. For studies looking to incite changes into the person, these checks are time sensitive. For instance in an experiment inducing hypertension, the physiological checks can be set at different time periods of the intervention so that the baseline checks do not eliminate physiological relevant values during hypertension.

In addition to these, the `check` command allows researchers to define positional constraints. For instance, it can verify that the maximum or minimum value occurs within a specific portion of the segment, such as the first or last percentage of data points (e.g. first 10% or last 20%). Furthermore, checks can be performed on the first or last data points in a segment, allowing researchers to validate starting or ending conditions explicitly. This check can eliminate dropouts, or stalls during data collection ensuring only valid quality signals pass through the data pipeline.

3.2.2 Take Command

The `take` command allows researchers to extract a specific subset of data based on duration or count, which is especially useful when only a representative portion of the dataset is needed for analysis. In many cases, researchers may not require the entire dataset, but instead want to focus on the last few valid cardiac cycles, or a fixed time window of continuous, high-quality data, such as the final 10 seconds of an intervention.

An example where PhysioMerge can be used is if an intervention had a sudden signal dropout at 15 seconds into a 30 second bin. **LabChart's extraction process would not allow the noisy region to be omitted. Instead, the researcher would be limited to three options: (i) include the full 30 seconds and retain the noise, (ii) analyze only the first 15 seconds before the dropout, or (iii) analyze only the last 15 seconds after the dropout.** Assuming an intervention occurs **immediately before an intervention, taking only the first 15 seconds may exclude the influence of central command, while taking the last 15 seconds may no longer represent a true baseline. As a result, the extracted data could differ substantially depending on which portion of the 30-second bin is used**. In contrast, PhysioMerge allows the full bin to be retained while excluding only the brief dropout segment, achieved by combining the `check` command (to identify the dropout) with the `take`

command (to extract the cleaned signal).

This command supports two primary modes of selection: time-based and cycle-based. In time-based mode, PhysioMerge extracts a segment of data corresponding to a specified number of seconds. In cycle-based mode, PhysioMerge can extract a fixed number of discrete physiological cycles, such as heartbeats or breath cycles. This is particularly powerful when working with partitioned data, where each entry represents a single physiological event. Both methods have are time directional, meaning they can be extracted from the start or the end of the bin.

Additionally, the `take` command supports both continuous and non-continuous data selections. This means researchers can choose to extract segments that are contiguous in time, or alternatively, pull from scattered valid entries throughout the dataset. This means that the `text` command follows the guidelines setup in the check commands. If a researcher wants at least 5 seconds of (continuous or non-continuous) valid data within a 30 s bin, then the take commands will try to isolate for at least 5 seconds.

This flexibility allows for highly tailored data extraction strategies that align with the goals of the study while maintaining control over data quality and context. Finally, the `take` command supports extracting the longest (in either time or cycle) of continuous data as well.

3.2.3 Priority Command

In real-world data collection, ideal conditions are not always met. Researchers may aim to extract, for example, the last 30 seconds of continuous valid data, but if such a segment does not exist, they might be willing to settle for 20 seconds instead. Similarly, when working with physiological cycles, a researcher may prefer the last 10 valid cardiac cycles, but accept fewer if necessary.

The `priority` command is designed to accommodate these scenarios by allowing researchers to define a ranked list of acceptable data extraction targets. This command enables PhysioMerge to attempt data selection based on a prioritized sequence of criteria, falling back to secondary options if the preferred conditions cannot be satisfied.

This mechanism provides greater flexibility and resilience in data extraction workflows. Rather than failing outright when ideal segments are unavailable, PhysioMerge selects the best possible option according to the user's preferences. In doing so, it balances strict data quality requirements with the practical realities of experimental variability and noise. The `priority` command thus ensures that researchers retain meaningful control over their extraction logic while maximizing the usability of their datasets.

3.2.4 Merge Command

The `merge` command enables boolean logical operations to be performed on the binary mask computed by the `check` command allowing for more complex and nuanced evaluations. This functionality is particularly valuable when multiple conditions must be considered simultaneously to determine the validity or relevance of a data segment. This allows researchers to construct composite validity criteria, such as requiring a

segment to pass multiple checks simultaneously (logical AND), or to be valid if it satisfies at least one of several checks (logical OR). As an example, identifying a participant with a severe reaction to a stimulus could be achieved using logical OR merges: if the possible reactions are hypoxemia, hyperventilation, or tachycardia, the evaluation would take the form of (hypoxemia OR hyperventilation) OR tachycardia.

This approach provides a mechanism for refining data selection and validation logic, supporting more comprehensive filtering based on multiple dimensions of quality or physiological meaning. In essence, the `merge` command acts as a logical toolkit, enabling researchers to tailor their data processing pipelines to meet greater complexity.

3.3 Modification of Data

3.3.1 Filter Command

The `filter` command supports several techniques designed to enhance signal quality and reduce noise. These include frequency band filters, moving window filters, and adaptive filters, each with its own strengths depending on the type of signal and noise. Each respective filter is fully adjustable. Frequency Band Filters: These filters, such as the Butterworth filter, are used to isolate specific frequency ranges within a signal. For example, they can help remove high-frequency noise or focus on particular physiological signal components, like heart rate in ECG signals.

Moving Window Filters: Filters like the Savitzky-Golay (Savgol) filter smooth the signal by fitting polynomials to the data within a moving window [12]. These filters are useful for reducing noise and fluctuations in time-series data while keeping the overall shape of the signal intact. Another filter like the sliding standardization filter, allows for peak detection in a signal [9].

Adaptive Filters: Adaptive filters adjust their parameters based on the characteristics and properties of the signal. Sometimes the noise profile across multiple participants is not constant, and over-filtering might mask out noise but also likely meaningful data/features or can even blunt absolute values (lower peaks). Whereas, less aggressive filters might not properly eliminate noise. Adaptive filters adjust to the characteristics of the signal itself, and can adapt to changes such as the introduction of a stimulus which changes the look of the signal. The Kalman filter combines predictions with noisy measurements to provide an optimal estimate based on probabilistic models, while being able to adapt to new data [13].

3.3.2 Cut Command

There are two primary methods for segmenting, or "cutting", data: one based on time and the other based on predefined comments. Comments function as markers embedded in the dataset to denote the occurrence of specific events. These events may include the initiation of an intervention, the beginning of baseline measurements, or the conclusion of an experimental condition. These markers are invaluable during both the data analysis and extraction phases, as they help researchers accurately identify and isolate meaningful

periods within the dataset.

In practice, data collection often includes buffer time at both the beginning and end of a session. This additional time allows researchers to confirm the stability and reliability of the recording before initiating baseline readings or beginning the intervention. However, using absolute time (e.g., the first 30 seconds of the entire recording) for analysis can be misleading or inappropriate if those initial moments contain noise, setup artifacts, or unrelated activity. Instead, researchers can define cuts relative to the comments, such as extracting the first 30 seconds between the "baseline start" and "intervention start" comments, ensuring that the data segment corresponds precisely to the intended experimental phase. Additionally, this allows for support of scenarios where multiple interventions are recorded concurrently. With instances where only 1 comment is present it can extract to that comment or after that comment.

This comment-based approach also enhances flexibility when working with complex datasets containing multiple interventions. Rather than treating the entire file as a single continuous recording, researchers can use comments to split the dataset into distinct segments corresponding to each intervention. Further commands can then be run on any selected segment independently, allowing for modular analysis and streamlined processing across varied experimental designs.

3.3.3 Data Partition Command

Data stored in variables can be further partitioned into lists based on physiological cycles, such as breath cycles or cardiac cycles. In this structure, a variable does not hold a single continuous segment of raw data, but rather a list of smaller segments, where each element corresponds to one complete breath or heartbeat. This approach enables cycle-level analysis, which is critical for extracting precise physiological metrics.

For instance, consider analyzing a blood pressure signal. The systolic pressure is defined as the peak pressure during each cardiac cycle, while the diastolic pressure corresponds to the minimum. If data is analyzed over a continuous 30-second window without segmentation, calculating the maximum value would yield only the single highest value across that entire interval. In contrast, when the data is partitioned into individual cardiac cycles, applying a maximum function would return a list of peak values, one for each cycle i.e. beat by beat maximum. Enabling the extraction of cycle-specific systolic measurements. This method significantly improves the granularity and accuracy of physiological signal analysis.

The `partition` command provides versatile segmentation of physiological signals by dividing continuous data into structured cycles or intervals based on multiple customization criteria. Signals can be partitioned according to their intrinsic features, such as peaks (e.g., R-waves in ECG or systolic pressure maxima) or troughs (e.g., end-diastolic points in cardiac cycles), creating segments that align with specific physiological events. A feature is cross-signal partitioning, where events detected in one signal, it can be processed in another signal. Such as ECG R-waves or respiratory phase transitions, are used to segment another, enabling synchronized multi-modal analysis. For example, all cardiac waveforms can be partitioned using the ECG R-R markers. This allows for all cardiac cycles to be evaluated against the same marker instead of each

being analyzed with their own markers. In particular, physiological transit times (e.g., pulse transit time) can be quantified, and cycle-to-cycle variations in these delays reflect changes in pulse wave velocity.

3.3.4 Adjustment Command

The `adjustment` command can integrate data across different sources. For instance, tabular data obtained from a duplex Doppler ultrasound using a handheld probe can be extracted with a separate device or tool, then combined with LabChart data for unified analysis. This flexibility supports multi-modal data integration within a single parsing framework.

The command operates based on start or end comments in LabChart. This allows external data to be inserted into the LabChart timeline by aligning it with a specific comment marking the start or end point of interest. This command is what allows data from multiple sources to be evaluated together.

3.3.5 Slice Command

The `slice` command enables temporal segmentation of physiological waveforms, allowing researchers to isolate specific portions of cyclic data for analysis. This functionality is particularly valuable for cardiovascular and respiratory studies where distinct waveform components carry important diagnostic information. The command operates by dividing each detected cycle into user-defined segments, either by percentage or absolute time intervals, while automatically aligning segments to key physiological markers such as R-waves or pulse onset points.

Some examples of how the `slice` command is to find the end diastolic wave of a cardiac cycle. Some cardiac waveforms such as the carotid artery blood velocity can be triphasic in nature [14]. Therefore finding the end-diastolic position is not as easy as taking the minimum value. First the waveform needs to be sliced so that only the last 80-100% of the cardiac cycle is being analyzed. The same goes for systolic peaks when doing a peak-peak marker model, the systolic peak is not necessarily the maximum value.

3.3.6 Plot Command

The `plot` command provides a way to visualize the physiological data extracted by PhysioMerge. This feature supports simultaneous plotting of multiple channels in a vertically stacked layout, enabling researchers to examine temporal relationships between different physiological signals such as ECG, blood pressure, and respiratory flow. Similar to LabChart, each channel is plotted in its own row. The command includes comprehensive customization options, including user-defined titles, and axis labels with units. Key physiological events can be highlighted using markers, which may be automatically generated from detected features (e.g., R-wave peaks). All plots can be automatically exported in multiple formats (PNG, EPS, SVG), and saved in designed locations with unique filenames, allowing you to save the visualization for publications or future reference.

3.3.7 Interpolate Command

The `interpolate` command enables data segments or individual cycles to be resampled to a different sampling frequency. This is particularly useful when working with signals of varying lengths or sampling rates, or when standardizing data for analysis such as machine learning or waveform comparison.

The command supports a variety of interpolation methods, allowing researchers to select the most appropriate technique for their data type and application. Available options include Piecewise Cubic Hermite Interpolating Polynomial (PCHIP) [15], Krogh interpolation [16], cubic splines [17], and others. These methods provide flexibility in how data is smoothed or reshaped, preserving key features of the signal while adapting it to the desired resolution or length.

By offering multiple interpolation strategies, the `interpolate` command accommodates a wide range of use cases. As different interpolation methods perform better on some types of signals, or effect the frequency domain differently than they affect the time domain.

A major advantage of the `interpolate` command is its ability to resample signals from one source with the frequency of another. For example, if LabChart recorded one channel at 1,000 Hz while another device logged data at 200 Hz, the command can upsample the lower-rate signal to match the higher resolution, or downsample the higher-rate signal to match the lower resolution. This ensures that when the data is combined, there wont be synchronization problems.

Beyond just adjusting sampling rates, the command can also standardize waveform lengths, helpful when comparing variable-length cycles (like heartbeats or breaths) for morphology studies. By interpolating each segment to the same duration, it allows for direct comparisons of shape, latency, and amplitude across cycles.

3.3.8 Normalize Command

The `normalize` command allows for the normalization of a data segment or cycle to a specified numerical range. This is useful for standardizing data across different scales, particularly when preparing inputs for comparative analysis or machine learning models.

Unlike fixed normalization schemes, this command provides full flexibility. Users can define any target range for normalization, such as 0–1, -1–1, or even custom bounds like 50–100, depending on the requirements of their analysis. This adaptability makes it possible to tailor the normalization process to specific physiological contexts, visualization preferences, or algorithmic needs.

3.3.9 Aggregation Command

The `aggregation` command is used to combine a series of segments or cycles into a single representative value, enabling summary statistics to be computed across a dataset. This is particularly useful for condensing repeated physiological events, such as cardiac or respiratory cycles; into a single descriptive metric.

A variety of aggregation functions are supported, including `min`, `max`, `mean`, `weighted mean`, `median`, `mode`, `std`, and `variance` allowing researchers to select the most appropriate statistical measure for their

analysis. For example, one might compute the mean of systolic peaks across multiple cardiac cycles or the maximum amplitude observed within a defined time window.

The `aggregation` command supports weights as well. This means that some cycles can be weighted more or less than others. For instance, if a researcher wants to average over time, the weight of each cycle should be their period. More unique weights can be provided as well such as those that favor earlier or later cycles as well.

The `aggregation` command provides flexible conditional processing through built-in quality checks, ensuring only valid data segments contribute to the final result. These checks act as filters, automatically excluding segments that fail predefined criteria such as signal quality thresholds or physiological plausibility limits setup by the users. This selective aggregation prevents corrupted or noisy data from skewing summary statistics while maintaining analysis rigor. The system handles errors intelligently, unlike traditional approaches that might discard entire time windows due to localized artifacts, this command can process valid sub-segments while excluding only the compromised portions.

3.3.10 Find Peaks Command

The `findpeaks` command provides robust detection of physiological events when native recording systems lack adequate marking capabilities. This functionality addresses a critical gap in many research workflows, as some commercial acquisition systems often are unable to identify or mark key physiological events like respiratory cycles or cardiac pulsations in recorded data [10]. The command implements a peak detection algorithm like that of LabChart, while offering more options for ensuring the intended peak was marked. Users can define minimum peak prominence to distinguish true physiological events from noise artifacts, set minimum height thresholds to exclude insignificant fluctuations, and adjust duration parameters to handle complex waveforms with multiple peaks per cycle (systolic and reflected peaks).

For respiratory studies, the command can accurately identify individual breaths. In cardiovascular applications, it reliably detects the systolic peak and end diastolic minimums of cardiac waves. Detected events are automatically marked with timestamps, creating a searchable database which can be used to partition data or extract pulse wave velocity measurements from.

3.3.11 Measurement Command

The `measurement` command is used to extract specific values from a data segment or cycle, allowing researchers to focus on key physiological metrics or derived features. This command supports a variety of calculations and can be customized to extract a wide range of measurements from the raw data.

For example, the command can be used to extract the maximum value from a specific channel, such as the peak $PETCO_2$ level from a PCO_2 channel, or the systolic peak from a blood pressure (BP) channel. Beyond these simple extractions, the `measurement` command also supports a range of statistical operations, such as calculating the mean, mode, median, minimum, and maximum of a segment.

Additionally, the command can be customized with more complex operations, such as calculating the period of a signal (based on the number of data points and the frequency), the amplitude (defined as the difference between the maximum and minimum values of a segment), or the frequency (defined as the inverse of the period). These calculations provide a way to derive additional insights from the raw data, such as determining the heart rate from cardiac cycles or assessing the variability of a signal over time.

3.3.12 Arithmetic Command

The `arithmetic` command provides a powerful and flexible tool for performing mathematical operations on individual channels or derived measurements. This functionality enables users to apply custom transformations to their data, such as unit conversions or cycle-based calculations. For example, to convert a frequency measurement (in Hz) to beats per minute (BPM), a simple arithmetic operation (*60) can be applied to the raw data.

Beyond basic scaling, this command supports complex, cycle-level computations that are essential for advanced physiological analysis. An example is the equation used to calculate mean blood pressure from the diastole (dias) and systole (sys) values $(2 * \text{dias} + \text{sys})/3$. In this case two measurements are taken to find sys and dias, then three `arithmetic` commands. The first would be to multiply dias by 2, then add systole, and finally divide by 3. If the measurements were done on partitioned data, then this would occur for each cycle, otherwise it would be for 1 segment.

3.3.13 Morphology Command

The `morphology` command is a specialized tool designed to extract structural features from physiological waveforms. It is primarily used with signals such as BP [18]–[20] and blood velocity [21]–[23], with future support planned for intracranial pressure (ICP) [24], [25]. These types of signals often contain detailed waveform patterns that can provide important insights into physiological or pathological conditions.

Users can select which morphological features they want to extract, based on the type of signal being analyzed. For example, in blood pressure waveforms, features such as the systolic peak, dicrotic notch, and augmentation index are often of interest. The augmentation index is calculated by dividing the height of the second peak by the height of the first, which requires the identification and measurement of both peaks.

The command currently supports the extraction of various features, including the height, latency, width, prominence, curvature, slope, area under the curve, and angle of peaks and valleys. These features capture the shape and structure of the waveform in detail, making it possible to analyze not just the magnitude of physiological signals, but also their form and progression over time.

By focusing on the morphology of signals, this command allows researchers to move beyond basic metrics like amplitude and frequency, enabling more comprehensive and meaningful analysis that can reflect subtle changes in physiological state.

3.3.14 Marker Command (Time Difference)

The `marker` command is used to calculate the time difference between specific points, or markers, within or across cardiac cycles. This is especially important for physiological measurements that depend on timing relationships between signals, such as pulse wave velocity, which is often derived from the time delay between the onset or peak of a waveform recorded at two different sites in the body.

This command allows researchers to measure the time interval between key landmarks in the waveform, such as the start, peak, or end of individual cardiac cycles. By identifying these markers and computing the time between them, PhysioMerge can extract valuable temporal features that are critical for assessing cardiovascular dynamics.

4 Experiment

An experiment was run to evaluate how PhysioMerge compares to manual data extraction methods (tool vs operator), how data extraction from two different users compare against each other (interoperator), and how a user's data extraction compares against their own re-analysis (intraoperator). Two human examiners were asked to extract what they considered to the be longest period of continuously valid cardiac cycles from a LabChart file within a 30 second bin. There were a total of eight 30 second bins over the 4 minute period analyzed. All participants were asked to extract from 5 different files, whose order was randomized using an online list randomizer.

The LabChart files were preset to record different physiological measurements: diastolic blood pressure (DBP), mean blood pressure (MBP), systolic blood pressure (SBP), diastolic middle cerebral artery(MCA) blood velocity (DMCAv), mean MCA blood velocity (MMCAv), systolic MCA blood velocity (SMCAv), partial pressure end tidal CO2 (PETCO2), cardiac output (CO), and MCA pulsatility index (PI). The largest and average error within each examiner, between examiners and against PhysioMerge was recorded.

5 Example applications

5.1 Extracting data for group-level statistics

PhysioMerge can extract data from multiple interventions for different individuals and organize it into a structured table for statistical analysis. It efficiently aggregates data across various treatments, conditions, or time points while maintaining participant-specific metrics. The output is formatted into a clean, tabular structure, with columns representing variables and rows containing individual or aggregated data points. The extracted dataset in its exported format is compatible with statistical software (such as R or Python) which can handle processing steps such as handling missing values, outlier removal, or generating summary statistics (mean, standard deviation). Additionally, PhysioMerge allows customization, enabling researchers

to select specific variables, merge data from different sources (multiple sensors), and prepare outputs in raw or normalized formats. This functionality can be useful for clinical trials (comparing treatment effects), longitudinal studies (tracking changes over time), and behavioral research (analyzing group differences).

5.2 Extracting physiological waveform features for machine learning

High-quality data is the critical for machine learning (ML) models. The performance and reliability of a model are directly influenced by the data used during training. While automated data extraction is not inherently complex, the real challenge lies in ensuring that the extracted data are both clean and reliable. This is a critical issue from both a physiological standpoint and a ML perspective. If data extraction processes did not include checks to ensure that the data were valid, then the model might build on patterns of noise. One of the key advantages of PhysioMerge is its ability to identify valid waveform data based on the specific criteria of a given study and isolate only the waves that meet those standards.

Waveform morphology has increasingly been used in ML applications [7]. Morphology offers valuable insights into systemic and clinical conditions without the need for additional data. PhysioMerge excels at rapidly extracting clean, reliable morphological data from large datasets. Allowing researchers to focus more on building and refining ML models, rather than spending excessive time on data processing.

6 Comparison to Other Systems

Tools for waveform morphology exist such as the MOCAIP [26]. The tool clusters waves together, and based on a reference library would select waves that have the least amount of noise. While capable of identifying when a particular peak is missing, the MOCAIP tool requires a database of clean waveforms to compare against. Additionally, it does not allow for the opportunity to look at each cardiac cycle with its bin. Because of the clustering portion of the algorithm, it takes what it determines as single best cardiac cycle to represent all the cardiac cycles in that cluster. This might not be suitable for studies looking at rapid fluctuations in waveform morphology such as arterial blood pressure changes in studies using a sit-stand to elicit fluctuations in blood pressure. PhysioMerge allows for aggregation of data to eliminate noise, but it is not an requirement of the system. Currently, PhysioMerge does not support clustering, or a database to compare against.

PhysioMerge is not just a waveform extraction tool, it helps with automation of data extraction. In LabChart, automation is achieved using macros. Although macros streamline the workflow, they do not validate individual waves or support batch processing.

Ensemble or ensembleR is a data extraction tool designed by Elucimed [27]. PhysioMerge is similar to Ensemble, but these programs differ in their approaches. Ensemble provides pre-defined workflows that guide users through common procedures from start to finish. While it allows for error reconstruction, it does not support custom validation criteria. Its reconstruction feature is designed primarily for device recalibration

during data collection, rather than handling error-prone data. In contrast, PhysioMerge operates using customizable commands, each with configurable options and order. It processes files in batches according to a predefined script. Within the script, checks can be setup to determine what is considered a valid wave allowing for it to operate on both clean and noisy signals.

7 Results

One of the key foundations for research is reliability. If another researcher replicates a previous study but is unable to see the same effect, this can pose uncertainty moving forward. Biological signals are highly variable; depending on how, when, or where you sample the effect might be different [28].

Herein we summarize the findings from our experiment to evaluate tool vs user, intra- and inter-operator variability of data extraction. The instruction of "*select the longest continuous segment of valid cardiac cycles in all channels, over a 30 s bin*" lead to different results within the same person and between different people. Evaluator one (11 hours of experience extracting data from LabChart) had a maximum difference of 6.27 mmHg within their 3 trials for systolic blood pressure, and a max difference of 7 cm/s for systolic MCAv. Evaluator 2 (about 1 hour of experience extracting data from LabChart) had a maximum difference of 11.56 mmHg for systolic blood pressure, and 8.66 cm/s for diastolic MCAv within their 3 trials. These are differences, that can lead to changes in how one interprets integrated physiology. This variability will also lead to skewing the results of a homogeneous group potentially increasing Type 1 error. PhysioMerge had a maximum error of 0.00 internal difference for all metrics.

Taking the average of each person's three trials and comparing the averages together shows less difference. For systolic blood pressure, a intraperson difference of 11.56 mmHg is reduced to 2.83 mmHg for interperson, and a comparison against PhysioMerge showed a maximum error of 2.30 mmHg. This shows that the natural variability of humans can be reduced by repeatedly extracting the same information, but at the cost of time.

PhysioMerge is substantially faster than the human evaluators. The faster of the human evaluators took 12 min 34 seconds to complete the data extraction process, whereas the fastest PhysioMerge evaluation was 15 seconds. Out of the 15 seconds, 14 of them were directed towards reading the LabChart files and converting it to numpy arrays to be processed by Python. The data processing took about 1 second to complete. It should be noted, that opening and closing files was not included in the time evaluations of examiners, but file reading and saving was included for PhysioMerge. Human evaluators improved in extraction speed over time, especially when running the same files multiple times. Comparatively PhysioMerge was consistent in processing time, with the biggest bottleneck being read speed. PhysioMerge required about 14 seconds to read the full LabChart files. But, PhysioMerge was able to process the 5 files together in about 1 second, whereas humans with different levels of skills were able to process the 5 files between 21-12 minutes. This means that a simple extraction of 5 people would take between 43-54 minutes if the intent was to reduce the effect human error and variability during the extraction process by having them repeat extraction 3 times.

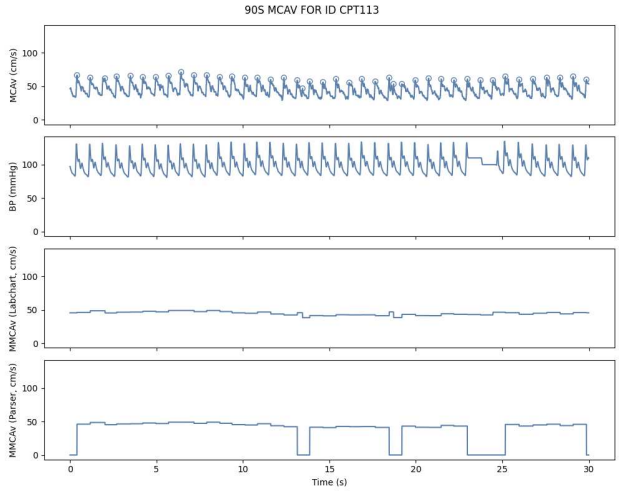

Figure 1: A 30 second bin, showing the raw MCAv value (top row) with markers as blue circles, the raw blood pressure signal (2nd row), the LabChart recorded Mean MCAv per cardiac cycle (3rd row), and the PhysioMerge error checked Mean MCAv per cardiac cycle (bottom row). For visualization purposes invalid cardiac cycles are shown as flat-lines at 0 cm/s.

8 Discussion

The definition for *valid* is subjective, and without explicit definitions its largely open to interpretations. Many researchers have selected valid cardiac cycles based on human opinion [1], [2], [8]. A limitation of this study was that the explicit definition of an invalid waveform was not defined, and user's expert judgment was meant to be used. We would like to note that most errors present in these files were dropouts or calibration errors; but there were occasional artifacts due to noise or movement. We would recommend researchers provide explicit numeric definitions for what defines an invalid cardiac cycle when using PhysioMerge. Because there is no true definition of *valid* cardiac cycles, it also means that there is no ground truth for comparison with respect to outcomes from Physiomerge.

When it comes to intraperson variability, PhysioMerge had none whereas the individual examiners had extremely high maximum errors such as 11.56 mmHg for SBP, but had substantially smaller average errors of 0.72 mmHg for SBP. Because the average value for each examiner is taken for the interperson comparison, any outlier will be reduced (but not eliminated) due to the aggregation of data.

Another area where human judgment plays a role is what defines the longest segment. Throughout the study, it was shown that the same person occasionally changed their start/end position within that 30 second segment because they determined that the longest segment was somewhere else. Because PhysioMerge does not have to rely on visual judgment, its more likely to spot minor errors that human participants would miss. For instance in figure 1 there are two not easily noticeable errors in the MCAv channel occurring between 14 and 19 seconds. Visually, they might look acceptable, but based on LabChart's wave detection algorithm

it identified two waves where one should be. In this instance none of the examiners during any of their three attempts noticed the two double waves; but they all detected the flat-line present at 23-25 seconds. Ultimately, each of those double waves are not valid signals. Because PhysioMerge and the examiner were told to take the longest continuous region of valid waves, PhysioMerge took from 1-13 seconds whereas each time the examiners missed spotting the error and took the waves from 1-23 seconds. Because, the MCAv slightly drops after 15 seconds, the total difference for this person at this bin was 1.98 cm/s difference with PhysioMerge value being slightly higher.

Where PhysioMerge is different than human evaluators is that it makes the same choice each time, and that choice is consistent across each participant. Additionally, it is no longer restricted to purely longest continuously valid segment. A key advantage of PhysioMerge is that, as long as the pre-configured criteria do not change, it will consistently provide the same output each time. Unlike human judgment, which can fluctuate, PhysioMerge's automated approach ensures that the same rules are applied to every signal, offering a reliable, repeatable process. Human examiners had massive fluctuations in data extraction, while their average intraperson error was only slightly higher than the interperson error, the one off slips resulted in massive errors of 11 mmHg or 8 cm/s difference within the same dataset and bin. This makes PhysioMerge a valuable tool for eliminating inconsistencies, leading to more accurate data analysis.

Because PhysioMerge is capable of greater consistency, it can look at smaller bins, part of the reason 30 second bins were taken was to ensure enough valid cardiac cycles can be found irrelevant of the data quality. LabChart is limited to continuous data extraction, but PhysioMerge is not. In the instance where a 15 second bin might be wanted, an examiner would first have to verify that the shortest continuous segment of valid data is 15 seconds. Alternatively, with PhysioMerge a warning is provided to indicate that one of the bins does not contain enough valid cycles. The user would then have to change the `cut` command to edit bin size and PhysioMerge would then operate at a larger bin size.

Research priorities and objectives often change over time, creating challenges when previously processed data must be revisited, reanalyzed, or extended. This can happen not only due to evolving research questions, but also in response to peer review feedback. In such cases, researchers typically need to return to each participant's dataset and reprocess data for all relevant time points, which is both time-consuming and labor-intensive. PhysioMerge addresses this challenge by enabling rapid reprocessing of large datasets in just seconds, streamlining what would otherwise be a tedious and error-prone task.

Even without reanalysis, PhysioMerge substantially decreases the time required to extract data from LabChart. This efficiency allows researchers to dedicate more time to high-value tasks, such as data collection, analysis, or manuscript preparation. Even without revisions, PhysioMerge reduces time spent extracting data from LabChart; this allows researchers to spend more time collecting data or writing papers.

9 Conclusion

PhysioMerge is a tool that allows for the rapid batch processing of files with checks to determine what is considered a valid breath or cardiac cycle. PhysioMerge performs similar to that of a human operator, but without any chance of error. PhysioMerge is consistent, compared to human operators which have occasional errors. Additionally, PhysioMerge is able to detect errors that are easily missed by human operators.

In the future *metacommands* are intended to be added. Metacommands are user created templates for combining multiple commonly run commands together. For instance, to calculate a single value for heart rate: the frequency of each cardiac cycle has be calculated (`measurement` command), the frequency needs to be multiped by 60 to get BPM (`arithmetic` command), and then the mean BPM has to be calculated (`aggregate` command). A metacommand would be made to indicate the order of Commands and their options, and from now on the metacommand can be called instead of multiple commands. These metacommands would replicate the process of macros in LabChart and workflows in ensembleR.

Acknowledgements

Funded by Natural Sciences Engineering Research Council Discovery Grant (granted to BKA), Heart and Stroke, Personnel Awards for Women's Heart & Brain Health (granted to TA), and Ontario Graduate Scholarship (granted to TA).

References

[1] B. Quachtran, R. Hamilton, and F. Scalzo, "Detection of intracranial hypertension using deep learning," in *2016 23rd international conference on pattern recognition (ICPR)*, IEEE, 2016, pp. 2491–2496.

[2] S. G. Thorpe, C. M. Thibeault, N. Canac, S. J. Wilk, T. Devlin, and R. B. Hamilton, "Decision criteria for large vessel occlusion using transcranial doppler waveform morphology," *Frontiers in neurology*, vol. 9, p. 847, 2018.

[3] C. Berka, D. J. Levendowski, M. N. Lumicao, A. Yau, G. Davis, V. T. Zivkovic, R. E. Olmstead, P. D. Tremoulet, and P. L. Craven, "Eeg correlates of task engagement and mental workload in vigilance, learning, and memory tasks," *Aviation, space, and environmental medicine*, vol. 78, no. 5, B231–B244, 2007.

[4] A. Tas, M. I. Bayhan, M. Yildiz, Y. Alan, Z. Atay, F. Sezer, C. Kitapli, I. K. Tas, S. Umman, and M. Sezer, "Alterations in cerebral artery flow velocity acceleration pattern correlate with cognitive impairment in diabetes mellitus," *Applied Neuropsychology: Adult*, pp. 1–10, 2025.

[5] S. G. Thorpe, C. M. Thibeault, N. Canac, K. Jalaleddini, A. Dorn, S. J. Wilk, T. Devlin, F. Scalzo, and R. B. Hamilton, "Toward automated classification of pathological transcranial doppler waveform morphology via spectral clustering," *PloS one*, vol. 15, no. 2, e0228642, 2020.

[6] C. Fuente Mora, J.-A. Palma, H. Kaufmann, and L. Norcliffe-Kaufmann, "Cerebral autoregulation and symptoms of orthostatic hypotension in familial dysautonomia," *Journal of Cerebral Blood Flow & Metabolism*, vol. 37, no. 7, pp. 2414–2422, 2017.

[7] M. Hüser, A. Kündig, W. Karlen, V. De Luca, and M. Jaggi, "Forecasting intracranial hypertension using multi-scale waveform metrics," *Physiological measurement*, vol. 41, no. 1, p. 014 001, 2020.

[8] S. Aggarwal, D. M. Brooks, Y. Kang, P. K. Linden, and J. F. Patzer, "Noninvasive monitoring of cerebral perfusion pressure in patients with acute liver failure using transcranial doppler ultrasonography," *Liver Transplantation*, vol. 14, no. 7, pp. 1048–1057, 2008.

[9] *LabChart - Life Science Data Acquisition Software — Data Analysis*. [Online]. Available: `https://www.adinstruments.com/products/labchart?srsltid=AfmBOoqk51lF0QLRcWtlQpIWSYxABYHtBhh1D-NxGBfRzdgRLjaFziYQ`.

[10] R. Woodman, D. Playford, G. Watts, C. Cheetham, C. Reed, R. Taylor, I. Puddey, L. Beilin, V. Burke, T. Mori, *et al.*, "Improved analysis of brachial artery ultrasound using a novel edge-detection software system," *Journal of applied physiology*, vol. 91, no. 2, pp. 929–937, 2001.

[11] *RespirAct*, Feb. 2024. [Online]. Available: `https://thornhillmedical.com/next-tech/respiract/`.

[12] R. W. Schafer, "What is a savitzky-golay filter?[lecture notes]," *IEEE Signal processing magazine*, vol. 28, no. 4, pp. 111–117, 2011.

[13] G. Welch, G. Bishop, *et al.*, "An introduction to the kalman filter," 1995.

[14] G. Mahe, C. Boulon, I. Desormais, P. Lacroix, L. Bressollette, J.-L. Guilmot, C. Le Hello, M.-A. Sevestre, G. Pernod, J. Constans, *et al.*, "Statement for doppler waveforms analysis," *Vasa*, vol. 46, no. 5, pp. 337–345, 2017.

[15] F. N. Fritsch and R. E. Carlson, "Monotone piecewise cubic interpolation," *SIAM Journal on Numerical Analysis*, vol. 17, no. 2, pp. 238–246, 1980.

[16] F. T. Krogh, "Efficient algorithms for polynomial interpolation and numerical differentiation," *Mathematics of Computation*, vol. 24, no. 109, pp. 185–190, 1970.

[17] S. McKinley and M. Levine, "Cubic spline interpolation," *College of the Redwoods*, vol. 45, no. 1, pp. 1049–1060, 1998.

[18] W. W. Nichols, S. J. Denardo, I. B. Wilkinson, C. M. McEniery, J. Cockcroft, and M. F. O'Rourke, "Effects of arterial stiffness, pulse wave velocity, and wave reflections on the central aortic pressure waveform," *The journal of clinical hypertension*, vol. 10, no. 4, pp. 295–303, 2008.

[19] G. E. McVeigh, C. W. Bratteli, D. J. Morgan, C. M. Alinder, S. P. Glasser, S. M. Finkelstein, and J. N. Cohn, "Age-related abnormalities in arterial compliance identified by pressure pulse contour analysis: Aging and arterial compliance," *Hypertension*, vol. 33, no. 6, pp. 1392–1398, 1999.

[20] K. Takazawa, N. Tanaka, M. Fujita, O. Matsuoka, T. Saiki, M. Aikawa, S. Tamura, and C. Ibukiyama, "Assessment of vasoactive agents and vascular aging by the second derivative of photoplethysmogram waveform," *Hypertension*, vol. 32, no. 2, pp. 365–370, 1998.

[21] S. G. Thorpe, C. M. Thibeault, S. J. Wilk, M. O'Brien, N. Canac, M. Ranjbaran, C. Devlin, T. Devlin, and R. B. Hamilton, "Velocity curvature index: A novel diagnostic biomarker for large vessel occlusion," *Translational stroke research*, vol. 10, no. 5, pp. 475–484, 2019.

[22] H. Zulaika, A. Azhim, M. Saleh, P. Baghepour, Y. Kinouchi, and F. Ibrahim, "Assessment of first derivative of doppler blood flow velocity in vascular aging," in *The 15th International Conference on Biomedical Engineering: ICBME 2013, 4th to 7th December 2013, Singapore*, Springer, 2014, pp. 68–71.

[23] J. W. Robertson, C. T. Debert, R. Frayne, and M. J. Poulin, "Variability of middle cerebral artery blood flow with hypercapnia in women," *Ultrasound in medicine & biology*, vol. 34, no. 5, pp. 730–740, 2008.

[24] A. V. Lele, A. L. Hoefnagel, N. Schloemerkemper, D. A. Wyler, N. Chaikittisilpa, M. S. Vavilala, B. I. Naik, J. H. Williams, L. V. Raghavan, and I. P. Koerner, "Perioperative management of adult patients with external ventricular and lumbar drains: Guidelines from the society for neuroscience in anesthesiology and critical care," *Journal of neurosurgical anesthesiology*, vol. 29, no. 3, pp. 191–210, 2017.

[25] S. Tabassum, A. Ruesch, D. Acharya, J. Yang, F. A. Relander, B. Scammon, M. S. Wolf, J. Rakkar, R. S. Clark, M. M. McDowell, *et al.*, "Clinical translation of noninvasive intracranial pressure sensing with diffuse correlation spectroscopy," *Journal of neurosurgery*, vol. 139, no. 1, pp. 184–193, 2022.

[26] X. Hu, P. Xu, F. Scalzo, P. Vespa, and M. Bergsneider, "Morphological clustering and analysis of continuous intracranial pressure," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 3, pp. 696–705, 2008.

[27] *elucimed*, 2018. [Online]. Available: `https://www.facebook.com/elucimed/`.

[28] L. Labrecque, J. S. Burma, M.-A. Roy, J. D. Smirl, and P. Brassard, "Reproducibility and diurnal variation of the directional sensitivity of the cerebral pressure-flow relationship in men and women," *Journal of Applied Physiology*, vol. 132, no. 1, pp. 154–166, 2022.

Variable	W:E1	W:E2	B:E1-E2	W:P	B:P-E1/E2
DBP	4.35	5.81	1.54	0	1.15
MBP	4.47	7.93	1.72	0	2.22
SBP	6.27	11.56	2.83	0	2.30
DMCAv	5.08	8.66	5.35	0	2.79
MMCAv	5.60	8.03	3.01	0	2.30
SMCAv	7.00	4.14	3.69	0	2.30
PETCO2	0.65	2.19	1.42	0	1.52
CO	0.675	0.455	0.260	0	0.171
PI	0.080	0.107	0.073	0	0.046

Table 1: Maximum error within and between evaluators and PhysioMerge. W:E1 for within error for evaluator 1, W:E2 for within error for evaluator 2, B:E1-E2 for the maximum error between the average of E1 and average of E2. The units of error are in the respective unit of the variable: mmHg for DBP, MBP, SBP, and PETCO2; cm/s for DMCAv, MMCAv, and SMCAv; and L/min for CO.

Variable	W:E1	W:E2	B:E1-E2	W:P	B:P-E1/E2
DBP	0.40	0.33	0.27	0	0.24
MBP	0.43	0.40	0.31	0	0.33
SBP	0.49	0.72	0.40	0	0.48
DMCAv	0.56	0.46	0.41	0	0.26
MMCAv	0.62	0.47	0.39	0	0.32
SMCAv	0.68	0.41	0.45	0	0.39
PETCO2	0.09	0.18	0.10	0	0.12
CO	0.076	0.045	0.043	0	0.039
PI	0.010	0.007	0.006	0	0.006

Table 2: Average error within and between evaluators and PhysioMerge. W:E1 for within error for evaluator 1, W:E2 for within error for evaluator 2, B:E1-E2 for the maximum error between the average of E1 and average of E2. The units of error are in the respective unit of the variable: mmHg for DBP, MBP, SBP, and PETCO2; cm/s for DMCAv, MMCAv, and SMCAv; and L/min for CO.

Trial	Evaluator 1	Evaluator 2	PhysioMerge-Reading	PhysioMerge-Processing	PhysioMerge-Overall
1	16:30	20:48	0:14	0:01	0:15
2	14:07	18:34	0:14	0:01	0:15
3	12:34	14:59	0:14	0:01	0:15

Table 3: Time to complete extract all data from five participants per trial. All time recordings are in min:sec format