Prediction-based Anomaly Detection in Multivariate Time-Series Data

Improving Wahoo Fitness Cycling Data Quality by Addressing Sensor Errors

M. Segers

Delft University of Technology

TU Delft

Prediction-based Anomaly Detection in Multivariate Time-Series Data

Improving Wahoo Fitness Cycling Data Quality by Addressing Sensor Errors

by

M. Segers

Responsible Supervisor :	H. Hung
Daily Supervisor:	C. Raman
Company Daily Supervisor:	K. Hendrickx
Thesis Committee:	H. Hung, S. Tan, C. Raman D. Tax and K. Hendrickx
Company Group:	K. Hendrickx, T. Camminady, M. Cassin F. van Nuland, B. van Vliet and T. Neel
Project Duration:	November, 2024 - June, 2025
Faculty:	Faculty of Electrical Engineering, Mathematics & Computer Science, Delft

Cover:	Heart Rate Monitors: Armband + Chest Strap | Wahoo Fitness EU by Wahoo Fitness used with permission (Modified)
Style:	TU Delft Report Style, with modifications by Daan Zwaneveld

(a) Facilitating University

(b) Facilitating Company

Abstract

Consumer-grade fitness trackers can produce unreliable physiological data due to sensor errors. The same holds for cycling data from Wahoo Fitness, where heart rate (HR) and power readings are essential for training and performance analysis. This thesis presents a prediction-based anomaly detection framework tailored to multivariate time-series cycling data. The approach reframes anomaly detection as a personalized physiological HR prediction problem. We define anomalies as deviations between measured sensor values and their predicted values, based on contextual activity metrics (e.g., power, cadence, speed, altitude, and gradient) and user-specific embeddings. The system combines ordinary differential equations (ODEs) modeling heart rate dynamics with machine learning techniques to capture non-linear, non-stationary, and individualized relationships. The model not only detects implausible values but reconstructs them with physiologically consistent alternatives. Compared to reconstruction-based methods, which are mostly used for anomaly detection in time series data, this physiologically grounded approach better differentiates between normal variation and true anomalies. Experimental results demonstrate effective identification and correction of HR and power anomalies, contributing to improved data quality and reliability in wearable fitness applications.

Contents

1 Introduction

1.1. Problem Context

Wearable technology for tracking workouts is a continuous trend in fitness tracking [1, 2, 3]. In cycling, users record physiological and mechanical metrics with a variety of devices during both indoor and outdoor rides. These sensors collect large volumes of time-series data over a workout. The resulting datasets are rich, multi-dimensional, and personalized.

Among the many available metrics, power and heart rate (HR) stand out as arguably the most important. Power reflects the mechanical output a cyclist produces (i.e. how hard they are pushing the pedals) while heart rate captures the body's physiological response to that effort. Together, these two signals form the foundation of most performance assessments in cycling. Section 2.2 provides more information on cycling metrics.

However, the reliability of this data is not guaranteed. Research has shown that consumer wearable technology is not always accurate and reliable [4]. Data can be missing, corrupted, delayed, or simply wrong due to sensor errors. Different users may also rely on different sensor types, each with its own measurement technique and level of precision, introducing further variation across users and sessions. Different sensors used are further described in Section 2.3. As a result, real-world cycling data often includes segments that are incomplete, inconsistent, or implausible. This thesis focuses specifically on detecting and correcting anomalies in power and heart rate data.

These large amounts of physiological data are increasingly used to make decisions, by athletes, coaches, and automated systems, to improve sport performance [5]. In cycling, heart rate and power data are core inputs for training plans, performance metrics (e.g., training load, VO_2max estimation), and fitness assessments. These data also feed into machine learning models that power coaching apps, activity recommendations, and athlete monitoring systems.

However, these applications depend critically on the accuracy of the input data. Erroneous sensor readings can lead to flawed training decisions, inaccurate feedback, and misinformed physiological modeling. For instance, overestimated power values can result in too large training loads, while heart rate dropouts may hide early signs of overtraining or illness. The impact is not limited to individual athletes. At scale, data-driven tools and recommender systems trained on corrupted datasets can reinforce false assumptions and amplify errors across entire user bases. Without reliable methods to identify and correct anomalous values, data pipelines built on wearables risk becoming untrustworthy and even harmful. Moreover, athletes are increasingly aware of these issues. Persistent errors reduce user trust in wearable technology brands and in digital coaching services. Providing accurate data is not just a technical problem, it is a product trust issue for wearable manufacturers.

To address these reasons we are working in collaboration with Wahoo Fitness[1], a manufacturer of cycling sensors and training devices, to create an anomaly detection system for cycling workouts. The

[1] https://eu.wahoofitness.com/

system identifies implausible data segments and reconstructs them with physiologically plausible replacements. An effective anomaly detection and reconstruction system can significantly improve the quality of cycling data by correcting implausible values. This leads to more accurate performance metrics, better informed training decisions, and increased trust in wearable devices, both for end users and for data-driven systems built on top of this information. For more background on the Wahoo Fitness data ecosystem, see Section 2.1.

1.2. Knowledge Gap

Most time-series anomaly detection models rely on reconstruction-based techniques. These models assume that if a data point cannot be accurately reconstructed from learned patterns, it is anomalous. While effective in controlled domains, this assumption fails in physiological data like heart rate during exercise. heart rate signals are not only non-stationary and nonlinear, but also highly individualized and context-sensitive. Identical heart rate patterns may be plausible in one workout but anomalous in another, depending on variables such as fatigue, hydration, terrain, or temperature. Current generic reconstruction models fail to capture any of these.

Moreover, reconstruction-based methods lack awareness of physiological context. It does not evaluate whether a heart rate value (body's response to effort) actually makes sense given the surrounding activity metrics, like power (effort of the cyclist). As a result, such models may incorrectly flag unfamiliar but valid heart rate patterns as anomalies, while failing to detect implausible ones that resemble normal sequences. Our objective is fundamentally different. We define anomalies not as statistical outliers, but as physiologically implausible responses. Values that contradict what we would expect given the user's individual characteristics and current level of exertion. In this view, the relationship between effort (e.g., power, cadence, gradient) and physiological output (e.g., heart rate) is central. A given heart rate response might be entirely appropriate for one rider but clearly anomalous for another. Accurately detecting such cases requires a model that incorporates both context and personalization.

Despite the extensive use of reconstruction, clustering, and graph-based methods in recent anomaly detection literature, no prior work has successfully modeled this physiological relationship between heart rate and effort as the basis for anomaly detection. Recent heart rate prediction models, such as the one by Nazaret et al. [6], demonstrate that we can predict heart rate from activity metrics using ODE-based models that incorporate individual physiological characteristics. However, these models are not leveraged for anomaly detection, even though they provide the deviation between measured heart rate and the physiologically expected heart rate.

This leaves a significant gap in the literature, there is currently no anomaly detection framework that uses physiological models tailored to individual users and context-dependent effort data to evaluate the plausibility of heart rate or power signals in cycling workouts.

1.3. Problem Statement

An anomaly in cycling data can be mathematically explained by having a difference in measured and true value that exceeds a certain threshold (i.e. a sensor error in capturing the correct physiological metric) and the reconstruction process replaces anomalous sensor readings with realistic values (i.e. as close as possible to the real physiological value). Below we define the problem mathematically following the notation of Bishop et al. [7]:

Let $\mathbf{t} = (1, 2, \ldots, t^k)$ be the sequence of time steps in a workout, where t^k is the total number of time steps in workout k. Each time step represents 1 second.

The set of metrics, $\mathcal{M} = \{\text{Power}, \text{HR}, \text{Speed}, \text{Cadence}, \text{Gradient}, \text{Altitude}\}$. Sensor s_i captures metric i, where $i \in \mathcal{M}$ (e.g., s_{power} is the sensor that captures power).

$\mathbf{X}$ represents all the true physiological values of the metrics $\mathcal{M}$ across the workout. $\mathbf{X}$ is a matrix of size $|\mathcal{M}| \times t^k$, where each column represents a time step and each row represents a metric.

$\tilde{\mathbf{X}}$ represents all the measured values, through sensor $s_i \quad \forall i \in \mathcal{M}$, across the workout. $\mathbf{X}$ is a matrix of size $|\mathcal{M}| \times t^k$, where each column represents a time step and each row represents a metric.

We include a masking matrix $\mathbf{M}$, similar size to $\tilde{\mathbf{X}}$, to make the model robust to missing sensor data.

$$\mathbf{M}_{i,j} = \begin{cases} 1, & \text{if data from metric } i \text{ is available at time } j \\ 0, & \text{if data from metric } i \text{ is missing at time } j \end{cases} \tag{1.1}$$

τ represents a threshold. Small deviations between measured and true heart rate or power are not automatically considered anomalies. Such deviations often arise from unobserved contextual factors, such as terrain variability, normal physiological fluctuations, or weather conditions, and do not necessarily indicate sensor errors. To avoid incorrectly labeling normal sensor behavior as anomalous, the model flags only deviations that exceed the threshold τ. The threshold accounts for noise, calibration offsets, and minor inaccuracies inherent to the sensor [8, 9].

We express non-anomalous workouts as:

$$|(\mathbf{X}_{i,j} - \tilde{\mathbf{X}}_{i,j}) \odot \mathbf{M}_{i,j}| \leq \tau \quad \forall i \in \mathcal{M}, \forall j \in \mathbf{t} \quad \text{(Non-anomalous workout)} \tag{1.2}$$

However, we detect anomalies only for the power and heart rate sensors, as Wahoo Fitness requires us to identify which sensor is anomalous.

A noteworthy mention is that power anomalies of short duration are infeasible to capture. Only once the power varies for a prolonged amount of time the correlation with other variables becomes infeasible. A short difference in power can originate from many different factors (e.g., a small hole in the road). By this we will only aim to detect time series power anomalies.

When a larger error component occurs (e.g., due to sensor malfunction, interference, or unexpected behavior), it causes the deviation to exceed the threshold, which triggers the system to flag an anomaly either over the full power time series or at specific heart rate time points.

Which leads to the following definitions of an anomaly in our work:

$$\begin{aligned} anom_{HR,j} &\iff |(\mathbf{X}_{HR,j} - \tilde{\mathbf{X}}_{HR,j}) \odot \mathbf{M}_{HR,j}| > \tau \quad \forall j \in \mathbf{t} \\ anom_{Power,:} &\iff \left(\frac{1}{|\mathbf{t}|} \sum_{j \in \mathbf{t}} \left| (\mathbf{X}_{Power,j} - \tilde{\mathbf{X}}_{Power,j}) \odot \mathbf{M}_{Power,j} \right| \right) > \tau \end{aligned} \tag{1.3}$$

In this work, we categorize anomalies by their temporal extent: point anomalies (isolated outliers), subsequence anomalies (corrupted time spans), and time series anomalies (entire workouts affected) based on the outlier types identified by Blazquez-García et al. [10]. We apply our detection and reconstruction approach at each time step, but evaluated across these different temporal extents to reflect realistic error patterns in cycling data. The core anomaly detection formulas operate at the level of individual time steps, making them directly applicable to all three anomaly types. Whether the anomaly occurs in isolation, across a subsequence, or over an entire session, the detection logic remains consistent.

It is important to note that while we adopt the categorization of anomalies from Blazquez-García et al. [10], distinguishing point, subsequence, and time series anomalies based on their temporal extent, we do not follow their definition of an anomaly in terms of statistical or behavioral outliers. In our work, we consider a heart rate or power measurement anomalous only if it deviates from the true physiological value due to a sensor error. This means that physiologically extreme or unusual values (e.g., a heart rate of 200 BPM while standing still) are not treated as anomalies if they are genuinely occurring and accurately captured by the sensor.

Following Equation 1.3, we could detect anomalies if we knew the true physiological value $\mathbf{X}$. We would simply flag a measurement whenever its deviation between true and measured values exceeds the threshold τ. However, in practice, we only observe the sensor measurement $\tilde{\mathbf{X}}$, not the true underlying value $\mathbf{X}$. This lack of ground truth makes direct anomaly detection infeasible and forces us to take an indirect approach.

1.4. Proposed Solution

We could address the unavailability of ground truth $\mathbf{X}$ by learning to predict heart rate, creating prediction $\mathbf{y}$. To predict heart rate, we use measured $\tilde{\mathbf{X}}$ values except the sensor measuring heart rate, defined as $\tilde{\mathbf{X}}_{-HR,:}$. We generate the prediction $\mathbf{y}_t$ based on:

$$\mathbf{y}_t = f(\tilde{\mathbf{X}}_{-HR,1:t} \odot \mathbf{M}_{-HR:,1:t}, \mathbf{u}_n^k) \quad \text{(Prediction)} \tag{1.4}$$

where we aim for $\mathbf{y} \approx \mathbf{X}_{HR,:}$. This means we try to learn to predict the expected response.

Learning the function f is a non-trivial task due to the following complexities:

- User dependency: The model conditions the function f on user-specific embedding $\mathbf{u}_n^k \in \mathbb{R}^z$, representing individual physiological characteristics of user n on the start of the users k^{th} workout. Therefore, f varies across users, i.e., $f(\tilde{\mathbf{X}}_{-HR,1:t} \odot \mathbf{M}_{-HR:,1:t}, \mathbf{u}_n^l) \neq f(\tilde{\mathbf{X}}_{-HR,1:t} \odot \mathbf{M}_{-HR:,1:t}, \mathbf{u}_m^k)$ for different users n and m but on identical $\tilde{\mathbf{X}}_{-HR,1:t}$. These user-specific embeddings are dynamically updated so that $f(\tilde{\mathbf{X}}_{-HR,1:t} \odot \mathbf{M}_{-HR:,1:t}, \mathbf{u}_n^{k-1}) \neq f(\tilde{\mathbf{X}}_{-HR,1:t} \odot \mathbf{M}_{-HR:,1:t}, \mathbf{u}_n^k)$ for two consecutive workouts $k-1$ and k for the same user n and $\tilde{\mathbf{X}}_{-HR,1:t}$, simulating fitness gain or loss over time. [6, 8, 11, 12, 13].
  - Note that not all users will have many varying workouts available to estimate their fitness. New users may have little workouts available. Some users will only have workouts with similar intensity available.
  - Note that we define users in our system by the account they use to track their workout. If different people (with different fitness levels) track workouts under the same account we cannot differentiate them and will have difficulty learning a fitness representation for this user.

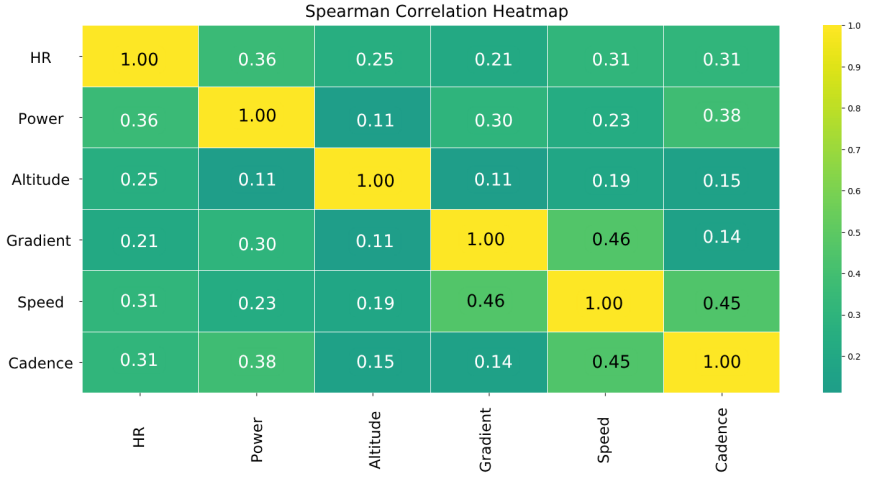

Figure 1.1: Heatmap of average Spearman's rank correlation coefficients between cycling activity metrics, illustrating the strength and direction of monotonic relationships among sensor data.

- Complex dependencies: The mapping f is highly non-linear, where sensor interactions involve complex dependencies [8, 9, 14]. A heatmap (Fig. 1.1) displays the average Spearman's rank correlation coefficient between all sensor values. Weak Spearman rank correlations between metrics indicate that their relationships are not consistently monotonic, which is characteristic of complex dependencies between sensors.
- Non-stationarity: The relationships between sensors are non-stationary; that is, the joint distribution of sensor data varies over time: $P(\tilde{\mathbf{X}}_{-HR,t} | \mathbf{u}_n^k, \tilde{\mathbf{X}}_{-HR,1:t}) \neq P(\tilde{\mathbf{X}}_{-HR,t+\Delta t} | \mathbf{u}_n^k, \tilde{\mathbf{X}}_{-HR,1:t+\Delta t})$. (e.g., early in the workout certain power will map to certain heart rate, once fatigue sets in the same power will map to different heart rate) [6, 8, 14].

- Unobserved contextual variables: Even for the same user-specific embedding and identical sensor readings the response can be different. While $f(\tilde{\mathbf{X}}_{-HR,1:t} \odot \mathbf{M}_{-HR:,1:t}, \mathbf{u}_m^k) = f(\tilde{\mathbf{X}}_{-HR,1:t} \odot \mathbf{M}_{-HR:,1:t}, \mathbf{u}_n^l)$ with identical $\tilde{\mathbf{X}}_{-HR,1:t}$ and $\mathbf{u}_m^k = \mathbf{u}_n^l$ we could still have different $\mathbf{X}_{i,:}$ for users m and n. This comes from unobserved contextual variables. (e.g., user m had better sleep then user n, even though their fitness is equal user m will have lower fatigue.) [6, 11]

However, detecting anomalies by predicting heart rate ($\mathbf{y}$) from other activity metrics ($\tilde{\mathbf{X}}_{-HR,:}$) introduces an ambiguity: when a large prediction error occurs, it is unclear whether the source is an anomalous input (power) or an anomalous output (heart rate), since the task proposed by Wahoo Fitness allowed those two metrics to be possibly anomalous. To disambiguate, we use a second model that predicts heart rate without using power, relying only on $\tilde{\mathbf{X}}_{-\{HR,Power\},:}$ to predict heart rate $\mathbf{y}$. If this model still detects an anomaly, the issue lies in the heart rate signal. If not, we attribute the anomaly to the power input. The notation for the model using power is $\mathbf{y}^{power}$ and for the model not using power $\mathbf{y}^{no-power}$.

With this information, we create a formula similar to Eq. 1.3. We detect an anomaly if:

$$\begin{aligned}
anom_{HR,j} \iff & |(\mathbf{y}_j^{power} - \tilde{\mathbf{X}}_{HR,j}) \odot \mathbf{M}_{HR,j}| > \tau_1 \\
& \text{and } |(\mathbf{y}_j^{no-power} - \tilde{\mathbf{X}}_{HR,j}) \odot \mathbf{M}_{HR,j}| > \tau_2 \quad \forall j \in \mathbf{t} \\
anom_{Power,:} \iff & \frac{1}{|\mathbf{t}|} \sum_{j \in \mathbf{t}} \left| (\mathbf{y}_j^{power} - \tilde{\mathbf{X}}_{HR,j}) \odot \mathbf{M}_{HR,j} \right| > \tau_1 \\
& \text{and } \frac{1}{|\mathbf{t}|} \sum_{j \in \mathbf{t}} \left| (\mathbf{y}_j^{no-power} - \tilde{\mathbf{X}}_{HR,j}) \odot \mathbf{M}_{HR,j} \right| \leq \tau_2
\end{aligned} \tag{1.5}$$

We reconstruct anomalous and missing sensor readings by imputing the anomalous or missing values. This can be easily done with:

$$\tilde{\mathbf{X}}_{HR,j} = \begin{cases} \tilde{\mathbf{X}}_{HR,j}, & \text{if } anom_{HR,j} = 0 \text{ and } \mathbf{M}_{HR,j} = 1 \\ \mathbf{y}_j, & \text{otherwise} \end{cases} \tag{1.6}$$

1.5. Contributions

We formulate the anomaly detection task not as a generic outlier detection problem, but as a context-aware prediction problem. Specifically, the learning problem becomes: given available sensor signals and user-specific information, predict the expected physiological response, and compare it to the measured signal. This reframing turns anomaly detection into a prediction task, while adding the complexity that we cannot observe the true ground truth $\mathbf{X}$. Prior work in anomaly detection typically flags deviations within a single sensor stream (e.g., heart rate) without considering context from related metrics (e.g., power, cadence, gradient). We model heart rate as a function of surrounding metrics, enabling anomaly detection that considers whether heart rate is plausible given the exertion, not just whether it is statistically unusual.

We summarize our contributions as follows:

- **Reframing anomaly detection:** We introduce a formulation that detects physiologically implausible sensor values by modeling the causal relationship between effort (power, cadence, gradient, speed) and physiological response (heart rate), rather than relying on statistical deviations within a single metric.
- **Leverage heart rate prediction for anomaly detection:** We transform an existing heart rate prediction model into an anomaly detector by interpreting deviations between predicted and measured heart rate as indicators of sensor errors. This approach allows us to reuse a physiologically grounded model for a new purpose.
- **Personalization through dynamic user embeddings:** We incorporate evolving user-specific embeddings to capture individual physiological characteristics and adaptation over time, enabling the model to differentiate between normal inter-user variability and true anomalies.

- **Heart rate prediction with ODEs:** We use a heart rate prediction model using ordinary differential equations (ODEs). We chose ODEs over black-box models (e.g., pure neural networks or transformers) because they embed domain knowledge from exercise physiology, ensuring more interpretable and physiologically plausible predictions.
- **Anomaly reconstruction capability:** Beyond detecting anomalies, our method reconstructs plausible heart rate values for corrupted or missing segments, improving data quality for downstream analysis.

To fulfill the requirements for effective anomaly detection in cycling data, a solution needs to address five key challenges defined in the problem statement: individual variability, complex dependencies, temporal dynamics, unobserved contextual variables, and noisy measurements. Without explicitly accounting for all five components, anomaly detection in this setting would be unreliable or overly sensitive to normal variations.

1.6. Research Questions

To guide the development and evaluation of this approach, the following research question has been formulated:

- RQ: How accurately can we identify and correct anomalies in heart rate and power data within cycling datasets?
  - How well do personalized user embeddings capture individual variability in physiological responses for anomaly detection?

    Hypothesis: Previous HR prediction work [6, 11, 12, 13, 15] shows that user embeddings improve HR prediction. We extend this to anomaly detection and hypothesize that dynamic embeddings will enable better separation of true anomalies from normal inter-individual variability.
  - How does anomaly detection performance vary across anomaly types (point, subsequence, full-session) in a physiologically grounded framework?

    Hypothesis: Literature suggests point anomalies are easier to detect [10], but it is unclear whether this holds when using a physiological model rather than a statistical one. We expect that using physiological causality may solve some ambiguity in full-session anomalies.
  - Can physiologically grounded heart rate predictions serve as effective replacements for anomalous or missing heart rate segments?

    Hypothesis: Reconstruction is a common approach for imputing missing or corrupted data, but it lacks physiological plausibility guarantees. We hypothesize that predictions grounded in effort metrics and user-specific physiology will yield more realistic imputations than generic reconstruction.

1.7. Thesis Structure

We organize this paper into seven main chapters. Chapter 2 outlines supplementary but interesting and related information that could help with understanding the thesis, but is not strictly necessary. Chapter 3 provides related work, presenting context to our scenario and the current state of research in anomaly detection divided into two systematic literature reviews. The first systematic literature review goes in depth about current anomaly detection techniques (Sec. 3.2), while the second covers heart rate prediction (Sec. 3.3). Chapter 4, Methodology, describes the approach adopted in this study. It begins with explaining the experimental objectives (Sec. 4.1) of the methodology. Afterward, we discuss the methodology of the heart rate prediction models in Section 4.2. The chapter then introduces the Dataset (Sec. 4.3). It concludes with an explanation of the anomaly detection and reconstruction approach (Sec. 4.4). Chapter 5, Results, starts with discussing the experimental setup (Sec. 5.1), then evaluates the heart rate prediction models (Sec. 5.2), showcases the results of anomaly detection (Sec. 5.3) and the results of reconstruction (Sec. 5.4). Chapter 6 discusses the implications, limitations, and potential future directions of the findings. Finally, Chapter 7 concludes the paper with a summary of contributions and key takeaways.

2 Background

This background section outlines supplementary information that, while not strictly necessary to follow the main thesis, provides useful context for understanding the data and problem setup. It covers the the metrics collected, types of sensors involved, and how anomalous segments originate. This information helps unfamiliar readers, or people interested in the broader scope of the thesis.

2.1. Data Collection Ecosystem

As mentioned, we work in collaboration with Wahoo Fitness. Wahoo is a company specializing in fitness technology, primarily focused on endurance sports such as cycling and running. Their product line includes bike computers, heart rate (HR) monitors, indoor smart trainers, power meters, smartwatches and a range of sensors that capture detailed performance metrics. Their devices collect high-resolution data across various physiological and mechanical dimensions, which serve as the foundation for the analysis conducted in this thesis.

The Wahoo data ecosystem is inherently decentralized and heterogeneous. Athletes often combine different devices, including third-party heart rate monitors or power meters, not all manufactured by Wahoo. As long as a Wahoo device, such as a bike computer or smartwatch, records the activity, we ingest the aggregated data, including data from external sensors, into the Wahoo cloud platform. These sensors vary widely in sampling accuracy, signal stability, latency, and error characteristics. Data are typically logged locally on the recording device and synced asynchronously, introducing additional variation in data completeness, temporal alignment, and quality. The result is a rich but noisy multivariate dataset characterized by missing values, inconsistent sensor reliability, and user-specific configurations. This raw, unfiltered input forms a realistic testbed for anomaly detection methods designed to operate in real-world, imperfect data environments.

2.2. Physiological and Mechanical Metrics

	Power	Heart rate	Cadence	Gradient	Altitude
Meaning	Force on the pedals	How hard your body is working	How fast you are pedaling	How steep is the road at this moment	How far above sea level
Unit	Watts	Beats per Minute	Rotations per Minute	% (Vertical / Horizontal distance)	Meters

Table 2.1: Overview of key metrics recorded during cycling activities, including their physiological or mechanical meaning and respective measurement units.

The key metrics relevant to our work, recorded by these devices, include power, heart rate, cadence, gradient, and altitude. Table 2.1 shows their meaning and units. All of these metrics are sample at a rate of 1 Hertz (one sample every second). We perform anomaly detection on two signals: heart rate and power. Different types of sensors collect both of these signals.

To ensure that all input variables contribute equally during model training, we standardize each variable to zero mean and unit variance in our preprocessing pipeline. Additionally, we adopt the original variable names used in the Wahoo ecosystem with a `_standardized` suffix to clearly indicate that we have transformed the values. Specifically, cadence becomes `cad_rpm`, power becomes `pwr_watts`, altitude becomes `alt_m`, gradient becomes `grade_perc`, and speed becomes `spd_mps`. This approach ensures clear traceability of each variable's origin while avoiding implicit weighting due to scale differences across metrics.

Power, measured in watts, represents the cyclist's instantaneous mechanical output and serves as a direct, objective indicator of exertion. Manufacturers calculate it from the torque applied to the pedals and angular velocity, and generally consider it the gold standard for quantifying physical effort during cycling. Heart rate, measured in beats per minute, reflects the internal physiological response to that mechanical load. However, heart rate responds non-linearly and with a lag, and many non-effort-related factors—including fatigue, dehydration, caffeine, psychological stress, ambient temperature, and altitude acclimatization influence it. Cadence, expressed in rotations per minute, quantifies pedaling frequency. Although it does not directly reflect effort or strain, it modifies power output dynamics and influences muscle efficiency and cardiovascular response. Low cadence with high power generally increases muscular load, while high cadence with lower power can elevate heart rate through cardiovascular strain. Gradient, defined as the percentage of vertical rise over horizontal distance, acts as a proxy for terrain difficulty. When combined with speed, it directly impacts the power required to maintain motion. Altitude, expressed in meters above sea level, influences air density and oxygen availability, indirectly affecting both power output and heart rate.

These metrics are not independent; they exhibit complex, non-linear, and user-specific interdependencies that vary over time. For example, the same power output at different gradients or altitudes may elicit different heart rate responses. Likewise, two athletes with similar cadence and power may exhibit different heart rate dynamics due to fitness levels or adaptation history. Understanding and modeling these dependencies is crucial for detecting anomalies that violate expected physiological relationships rather than merely statistical norms.

2.3. Sensor Technologies

Figure 2.1: Chest strap for HR measuring

Figure 2.2: Optical wrist-based sensor for HR measuring

Figure 2.3: Smart bike measuring power output

Figure 2.4: Power pedals measuring power output

Figure 2.5: Smart trainer measuring power output

Figure 2.6: Examples of Wahoo heart rate and power sensing technologies used in cycling. Adapted from Wahoo Fitness.

Monitoring physiological signals such as heart rate and power output during cycling relies on a range of sensor technologies, each with different underlying principles, form factors, and levels of accuracy. Even when measuring the same metric, different devices capture data differently and yield varying results. We will briefly outlines the main sensor types used for measuring heart rate and power in cycling and highlights how these differences lead to inconsistencies in the recorded data.

For heart rate measuring the gold standard technique is electrocardiograph (ECG) with 12 electrodes measuring changes of electrical potential. In the domain of activity tracking this however not very feasible and thus two more common approaches of heart rate measuring dominate. One being ECG based chest straps (Fig. 2.1) and one being optical sensors (Fig. 2.2) based on photoplethysmography (PPG). Chest straps which also register varying electrical potentials, but only with two electrodes, show high similarity 0.85-0.99 to ECG [16] showcasing why they are the gold standard in wearable heart rate measuring. Optical sensors on the other hand, being much cheaper and wider spread, use a light

source and detector. While the pulse wave of heart rate is running through the veins the amount of red bloods cells is slightly increased. The red blood cells absorb the light leading to a different return of light to the detector. Although widely used for their affordability, these devices have notable limitations. These sensors often fail to capture rapid increases and decreases in heart rate and various error sources influence them, such as skin color or tattoos, ambient light interference, pulse latency, and the reliance on algorithms that convert pulse signals to heart rate. As a result, their similarity to ECG readings varies significantly, ranging between 0.11-0.99 [17, 18].

Different kind of devices are suitable for capturing power. These include smart bikes (Fig. 2.3) which are fully integrated bicycles for indoor riding, power pedals (Fig. 2.4) or cranks for measuring power during outdoor rides, or smart trainers (Fig. 2.5) which are the most commonly used type of power measurement. A smart trainer replaces the rear wheel of a standard road bike to provide an indoor cycling experience. This last option is the most affordable alternative for power measurement. For power measuring the gold standard technique is using strain gauges to measure torque applied combined with the angular velocity. Their accuracy mainly lies in research and development of the producing companies. Decathlon reports an accuracy of $\pm 5\%$ for their Van Rysel D100 trainer, while Wahoo Fitness reports an accuracy of $\pm 1\%$ for their Wahoo Kickr, showcasing the difference between different manufacturers.

In both heart rate and power sensing, the choice of hardware introduces variability that leads to different levels of accuracy across users and workouts. This inconsistency complicates the interpretation of recorded signals and motivates the need for robust anomaly detection systems that account for such sensor-dependent variation.

2.4. Sources of Anomalies

Wahoo Fitness has identified a range of realistic anomaly scenarios that occur frequently in field data. These issues affect both heart rate and power measurements and highlight the diversity we must consider when developing anomaly detection systems.

Heart rate signals are particularly sensitive to sensor placement, environmental conditions, and the underlying measurement technology. Several common anomaly scenarios include:

- Signal Dropouts: heart rate data may be missing for short or extended periods due to temporary signal loss, poor skin contact, or connectivity issues.
- Optical Sensor Issues:
  - These sensors often fail to detect rapid changes in heart rate, introducing a lag between actual exertion and the recorded signal.
  - In cold weather conditions, reduced blood flow near the skin surface causes optical sensors to struggle with accurate detection.
- Chest Strap Issues: While generally more reliable, chest straps can produce spurious high heart rate readings due to static electricity buildup, especially in dry conditions or during vigorous motion with static clothing.

Power measurement is also prone to various types of errors, often stemming from hardware setup or calibration issues. Common sources of anomalies include:

- Signal Dropouts: Similar to heart rate, power data may be intermittently missing, either due to sensor disconnection or communication errors.
- Incorrect Calibration: If a power meter is not calibrated correctly, the entire workout may record systematically inaccurate values, either overestimating or underestimating power.
- Pedal Malfunction: In dual-sided power meters, a pedal failure can result in unbalanced or misleading power data, as the system reports only half of the actual power.
- Low-Precision Devices: Some power meters, especially low budget models, may have inherently lower accuracy or less robust algorithms, contributing to noisy or biased measurements.

These anomaly scenarios reflect real-world challenges and underscore the need for detection systems that can account for both transient and systematic deviations in sensor data.

2.5. Hardware Setup

Component	influ5 (CPU only)	gpu01 (GPU accelerated)
CPU Model	2 × AMD EPYC 7452 (32-core)	2 × AMD EPYC 7413 (24-core)
Total CPU Cores	64	48
Clock Speed	2.35 GHz	2.65 GHz
RAM	503.6 GB	503.4 GB
GPU	None	3 × NVIDIA A40 (48 GB each)
Job Scheduler	Slurm	Slurm
Python Version	3.12	3.12
PyTorch Version	2.2.1	2.2.1
CUDA Version	Not used	12.1

Table 2.2: Hardware specifications of the two DAIC [19] HPC nodes used for training and inference.

We conducted all experiments on two high-performance computing (HPC) nodes provided by the DAIC [19] infrastructure: influ5 and gpu01. We executed the CPU-based experiments on influ5, which features two AMD EPYC 7452 processors (32 cores each, 2.35 GHz), totaling 64 CPU cores and 503.6 GB of RAM. For GPU-accelerated experiments, we used gpu01, which features two AMD EPYC 7413 processors (24 cores each, 2.65 GHz), 48 CPU cores, and the same RAM capacity. Additionally, we use gpu01, which is outfitted with three NVIDIA A40 GPUs (48 GB VRAM each), to enable efficient training and inference of deep learning models. Both nodes use Slurm as the job scheduler and were configured with Python 3.12, PyTorch 2.2.1, and CUDA 12.1 (Table 2.2). Logging the setup also supports fair comparison between models and aids reproducibility in future experiments.

3

Related Work

The objective of the literature review is to understand the current state of research, identify gaps and determine if current solutions similar to the problem we try to solve already exist and how we could leverage them in our specific context. The structure of this literature review reflects a progressive deepening of insight, with each section building on the previous one. As new perspectives and findings emerged, they naturally guided the focus and direction of the subsequent sections.

3.1. Initial Anomaly Detection Literature

Initial exploration of anomaly detection involved reviewing key surveys [10, 20] on anomaly detection, identifying methods applicable to non-parametric, multivariate time series as our workout data. This analysis revealed four main types of anomaly detection for this domain:

- Histogram-based methods [21] optimize data representation but fail to detect all local anomalies or handle subsequence outliers effectively.
- Model-based approaches [22] relying on smooth trends also proved unsuitable due to abrupt changes in cycling data.
- Dissimilarity-based methods [23, 24, 25] struggle with the non-linear, context-dependent relationships in our data.
- Isolation techniques [26] struggle with the non-stationary nature of cycling data.

As explained for each method, none of the proposed methods could handle the nature of the data from the cycling dataset. To address these limitations in initial literature, we initiated a systematic literature review following the PRISMA[1] framework. This allows us to use insights into the shortcomings of these existing techniques and to gain a deeper understanding of the underlying data characteristics.

3.2. Systematic Literature Review on Anomaly Detection

We searched for relevant papers in three databases: ACM Digital Library, IEEE Xplore, and Scopus. Firstly, ACM Digital Library and IEEE Xplore cover a wide array of computer science topics making it a valuable literature source for our research. Secondly, Scopus is a multidisciplinary database chosen as it provides a broad range of articles from different academic fields that can help us get a more diverse understanding of the topic.

The Query Expression box displays the query used for the review. We developed this query by identifying key factors in our research and selecting synonyms commonly found in the literature. We want to remain focused on unsupervised anomaly detection for multivariate time series data, but incorporate the flaws of initial literature exploration to strengthen our outcomes. We do this by incorporating the non-stationary nature of the cycling data in the query.

[1] https://www.prisma-statement.org/

Query Expression

(”Anomaly detection” **OR** ”Outlier detection”) **AND** (”Multivariate” **OR** ”Multi-dim*” **OR** ”High dim*”) **AND** ”Time Series” **AND** (”Non-param*” **OR** ”Data-driven” **OR** ”Unsuperv*” **OR** ”Cluster*”) **AND** (”Non-stationar*” **OR** ”nonstationar*” **OR** ”non-linear*” **OR** ”nonlinear*” **OR** ”Time-var*” **OR** ”Heteroscedastic”)
The search was performed on 29/11/2024.

We removed duplicate records (n=6), conference reviews (n=5), and non-English records (n=2) during the identification process.

During the screening step, we examine the title, keywords, and abstract of each record to determine whether it meets the selection criteria.

Inclusion criteria:

- Follow the requirements of the search query: Anomaly detection on multivariate time series with non-parametric techniques and difficult non-stationary relations between time series.
- Proposes a novel algorithm/technique.

Exclusion criteria:

- The main topic of the paper is not solely anomaly detection.
- The main data type of the paper are not time series.
- Paper focuses on univariate time series.
- The paper is a review of anomaly detection techniques.
- Focuses on computer vision applications.

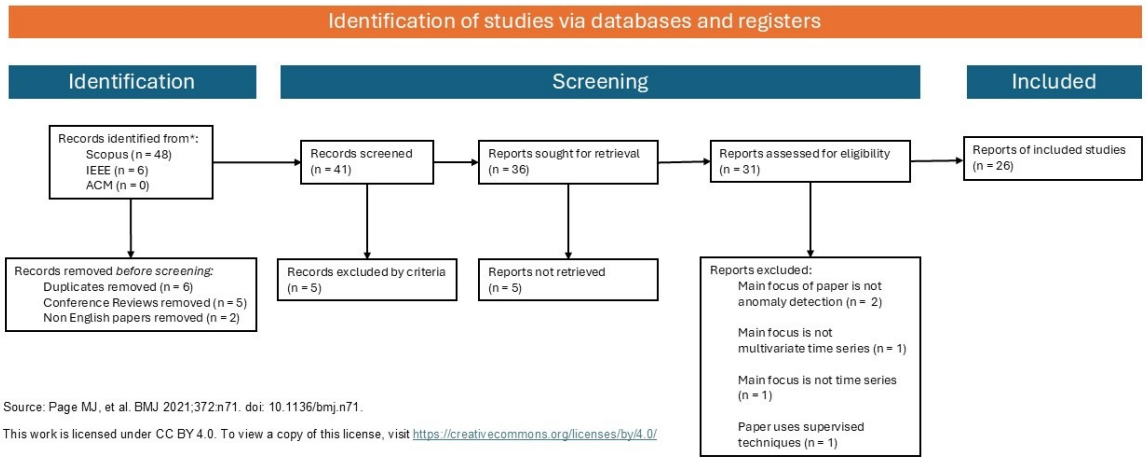

Figure 3.1: PRISMA flowchart illustrating the selection process for papers included in the systematic review on multivariate time-series anomaly detection in non-stationary data.

After filtering on the exclusion criteria, we included 36 records and excluded 5. During the retrieval process, we could not access 5 papers. After full-text eligibility assessment, we excluded five additional papers. Although they mentioned multivariate time series anomaly detection, they did not primarily focus on anomaly detection, lacked a multivariate or time series component, or did not apply unsupervised methods. Ending with a total of 26 papers included in the literature review (Figure 3.1). The literature review organizes papers into themes that frequently appear in the included studies. Many of the papers combine different techniques and thus appear in different subsections.

3.2.1. Clustering

A frequently used technique in anomaly detection is clustering [27, 28, 29, 30]. Clustering groups data points based on similarity in a feature space. Clustering serves various purposes, such as pattern recognition and classification, and is also highly useful for anomaly detection in many different use cases.

Clustering groups a model's output into non-anomalous and anomalous clusters. Dairi et al. [27] explores this by detecting influent conditions of wastewater treatment plants. They propose a novel method combining recurrent neural networks (RNN) and restricted Boltzmann machines (RBM) with one-class support vector machines for clustering to detect anomalies in multivariate time-series data. The RNN-RBM model captures both short- and long-term temporal dependencies into extracted features. The extracted features are then classified using one-class support vector machines for clustering normal and abnormal points. They claim prior methods either ignored temporal dependencies or relied on standalone clustering. Vishwakarma et al. [28] does something similar by applying clustering on transformed time-series to detect anomalies. The method transforms time series data into a lagged bivariate dataset to capture temporal dependencies. It then applies robust clustering using Mahalanobis distance to identify outlier pairs based on statistical deviations. Clusters of outliers are iteratively refined to exclude false positives. The method trains a single-layer hybrid functional neural network for forecasting using the outlier-free dataset, ensuring better predictions by eliminating the influence of outliers. The paper addresses the challenge of robust outlier detection in non-stationary multivariate time series, where traditional methods struggle with clustered outliers and high-dimensional data.

Another clustering technique used in outlier detection is clustering to reduce dimension to uncover latent spatio-temporal features. Oucheikh et al. [29] proposes a deep learning framework for real-time anomaly detection in connected autonomous vehicles using spatio-temporal clustering. The framework employs a Long Short-Term Memory (LSTM) autoencoder for dimensionality reduction, Grey Wolf Optimizer for clustering that reduced dimensional data. After forming the clusters, the method trains a specific anomaly detection model for each cluster. The paper addresses the challenge of detecting anomalies in high-dimensional, heterogeneous spatio-temporal data, where traditional methods struggle with context-aware detection. However, the method uses context in the form of location and time and is tailored for connected vehicle telemetry. This context differs fundamentally from the physiological context we require: the relationship between effort metrics (e.g., power, cadence) and heart rate (HR). The model also lacks personalization and does not model causal effort-to-HR dependencies. It fails to support our mathematical formulation (e.g., Eq. 1.4) where the anomaly depends on a mismatch between expected and measured heart rate for a specific user.

A complete other way of using clustering is in post-processing once the anomalies are already detected. He et al. [30] uses fuzzy c-means clustering, not to identify anomalies, but to group anomalies for identifying patterns in the anomalies. He et al. [30] combines a nonlinear autoregressive with exogenous inputs model for detrending, sequential collective and point anomaly detection for identifying anomalies, and only then uses fuzzy c-means clustering based on empirical cumulative distribution functions to group anomalies. Validated on telecommunications and benchmark datasets, the method excels in detecting and clustering diverse anomalies. By integrating detection and clustering in a single framework, it addresses limitations of traditional methods in handling dynamic data distributions and enables more effective anomaly classification for real-time decision-making.

In our setting, anomaly detection demands personalized mappings from multivariate effort metrics to expected physiological responses, as formalized in what makes Equations 1.4 complex. Clustering treats anomalies as statistical outliers in the feature space without evaluating whether values are physiologically coherent. As such, clustering techniques offer neither personalization nor interpretability aligned with our modeling objectives. Methods cannot assess physiological plausibility and fail to address the user-specific, non-linear, and dynamic dependencies central to our model. The approach of He et al. [30] could be an interesting approach for future work to cluster identified anomalies for anomaly classification.

3.2.2. Memory Units

Many papers use a memory unit to support a model in capturing the temporal dependencies of the time series. For time-dependent data, memory units like Long Short-Term Memory (LSTM) and Gated

Recurrent Units (GRU) are widely used [27, 29, 31, 32, 33, 34, 35, 36, 37]. Recurrent neural networks (RNN) use these components to handle sequential data, retaining both short- and long-term dependencies across time series. Their ability to model temporal patterns and adapt to abrupt shifts in data makes them well-suited for detecting anomalies in non-stationary datasets, such as those observed in cycling data.

Autoencoders are a powerful tool for identifying the most important features in data by reducing its dimensions and learning a compressed representation. However, for time series data, they cannot capture temporal dependencies in data which is necessary for the non-stationary nature of the cycling data. Unsurprisingly many authors [29, 31, 32, 33] discuss the addition of a memory unit in the autoencoder architecture. Other models like the Restricted Boltzmann Machines from Dairi et al. [27] are also extended with a RNN memory unit to capture the temporal correlation.

Zhang et al. [34] extends the LSTM autoencoder approach by using a Convolutional Gated Recurrent Unit (ConvGRU) and a Variational Autoencoder. ConvGRU incorporates convolutional layers into the gating mechanism, enabling it to capture dependencies across features (spatial structure) alongside temporal patterns. The convolutional operations also reduce the complexity by focusing on localized feature interactions, leading to better representation learning. Similarly, papers [35, 36] extend their model with attention-based BiLSTM. Attention-based memory units compute a score for each time step representing the importance of the time step. BiLSTM extends regular LSTMs by processing input sequences in both forward and backward directions, allowing it to incorporate context from both past and future timesteps. This bidirectional nature makes BiLSTM particularly effective for offline applications, where predictions can leverage future information to improve accuracy.

Dai et al. [37] on the other hand proposes a model that incorporates a Switching Gaussian Mixture Variational Recurrent Neural Network. The RNN in this architecture captures temporal dependencies in multivariate Key Performance Indicators, while the switching mechanism handles non-stationary temporal characteristics.

Architectures incorporating memory units (e.g., LSTMs, GRUs) are widely used to model temporal dependencies in sequential data. These models, in principle, could represent time-evolving physiological responses (e.g., Eq. 1.4). However, temporal modeling alone is insufficient. Without a physiologically grounded, multi-modal prediction framework, these methods cannot fulfill the context-aware requirements of our problem.

3.2.3. Graph-based

Different papers [36, 38, 39, 40, 41, 42, 43, 44, 45, 46] uses a graph structure for anomaly detection. The method frames anomaly detection as a graph problem, representing time series from different sensors or variables as nodes and dynamically learning edges based on similarity metrics. The learned structure evolves based on the varying relationships between nodes. Anomalies are then classified by detecting abrupt shifts in node relationships.

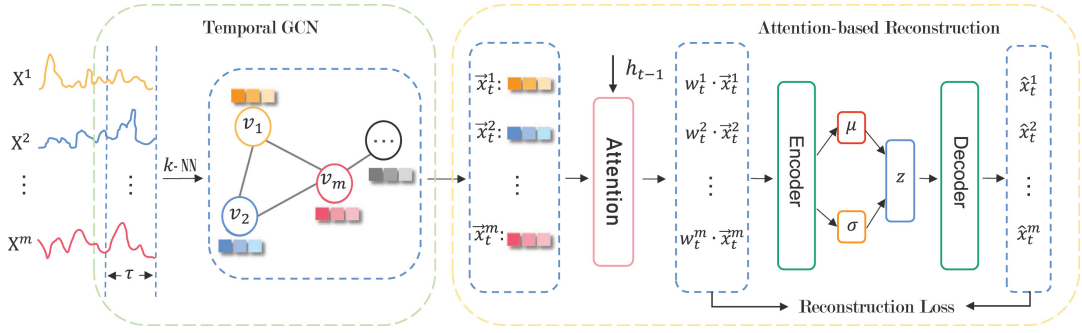

Figure 3.2: Graph-based encoder-decoder architecture for time-series anomaly detection. Inputs are processed via a Temporal Graph Convolutional Network (GCN), followed by attention mechanisms and reconstruction through a variational encoder-decoder. Adapted from Shi et al. [40].

It's important to note that many papers do not rely solely on graph-based models, but instead combine

them with other deep learning techniques. Often frameworks [38, 39, 40] use the graph representation as an input for an encoder-decoder model that detects the anomalies (Fig. 3.2). Other graph-based approaches include the paper of Zheng et al. [41], which uses a Spatial-Temporal Graph Neural Network combined with Dilated Convolutions for the temporal modeling. Similarly, Yang et al. [42] apply a transformer to a Graph Neural Network to learn long-range dependencies between nodes. In the HCroSTG framework, Ding et al. [43] construct two graphs, one dynamic graph capturing the evolving non-stationary relationships between variables and one static graph to encoded persistent associations between variables. The method employs a Dual Temporal Graph Attention Module to extract non-stationary temporal features, such as trends and seasonality.

Amil et al. [47] uses graphs in a different manner. The study proposes two graph-based methods. The first a Percolation-Based Method uses graphs where nodes represent data points while edges represent the distance between points. The method removes edges sequentially, starting from the highest weight downward, monitoring graph fragmentation. The method flags nodes as outliers if they disconnect early from the largest connected component. The second, IsoMap-Based Method applies the IsoMap algorithm for non-linear dimensionality reduction, mapping data onto a low-dimensional manifold. Compares geodesic distances in the reduced space with original distances using Pearson correlation. Points poorly fitting the manifold receive high outlier scores. The paper addresses the need for flexible, graph-based outlier detection methods that are parameter-free (Percolation-Based) or parameter-optimized (IsoMap-Based).

Graph-based methods represent sensor data streams as dynamic graphs and detect anomalies via changes in topological structure. While they can uncover complex inter-sensor relationships, these models do not incorporate domain knowledge about physiological correctness, nor do they model user-specific adaptation (e.g., through personalized fitness embeddings as in the required user dependency in Eq. 1.4). In our case, anomaly detection hinges not on generic correlation shifts but on whether heart rate is plausible given observed effort metrics. Graph approaches remain agnostic to this causality.

3.2.4. Reconstruction-based

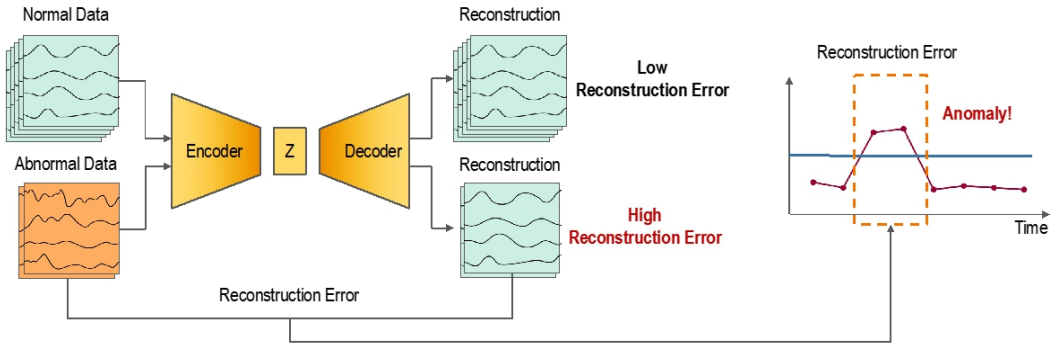

Figure 3.3: Reconstruction-based anomaly detection using an encoder-decoder architecture. Normal and abnormal data are encoded into a latent representation and reconstructed; anomalies are identified based on high reconstruction errors. Adapted by Yeseul [48].

The most common technique used however was reconstruction-based anomaly detection [29, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 42, 49, 50, 51, 52, 53] (Fig. 3.3). In essence, the model takes a multivariate time series as input and tries to reproduce the measured target variable as accurately as possible. Let $Y^m = \{y_1^m, y_2^m, \ldots, y_n^m\}$ denote the actual measured values of the variable of interest over time, and let $X^m = \{x_1^m, x_2^m, \ldots, x_n^m\}$ represent auxiliary context variables, and let $Y^p = \{y_1^p, y_2^p, \ldots, y_n^p\}$ be the model's reconstruction (i.e., its best guess of what the signal should have been, based on learned patterns). Deviations between Y^m and Y^p indicate potential anomalies.

However, the mapping from Y^m to Y^p is not direct. Instead of learning a one-to-one function, the model is deliberately constrained to first compress the input into a lower-dimensional latent space. This compression forces the model to capture only the most essential features of the data, discarding

noise and less relevant variation. The model decodes the reconstructed signal Y^p from this latent representation. This setup regularizes the learning and makes the model sensitive to abnormal patterns it cannot compress and recover accurately.

Formally, they implement this using an encoder-decoder architecture, defined as:

$$z = \text{Encoder}_\theta(Y^m, X^m) \quad \text{(latent representation)} \tag{3.1}$$
$$Y^p = \text{Decoder}_\theta(z) \quad \text{(reconstructed signal)} \tag{3.2}$$
$$Y^p = f_\theta(Y^m, X^m) = \text{Decoder}_\theta(\text{Encoder}_\theta(Y^m, X^m)) \quad \text{(overall model)} \tag{3.3}$$

To detect anomalies, they compute the reconstruction error:

$$\varepsilon = \|Y^m - Y^p\| \tag{3.4}$$

They classify each point in time as anomalous or not by comparing this error to a threshold τ:

$$anom_t = \begin{cases} 1 & \text{if } \varepsilon_t > \tau \quad \text{(anomaly)} \\ 0 & \text{otherwise (normal)} \end{cases} \tag{3.5}$$

Optionally, they can replace anomalous values with the model's reconstruction:

$$Y_t^{\text{reconstruct}} = \begin{cases} Y_t^p & \text{if } anom_t = 1 \\ Y_t^m & \text{if } anom_t = 0 \end{cases} \tag{3.6}$$

Some papers used transformers for this, originally developed for natural language processing, have emerged as a powerful alternative for time-series anomaly detection [49, 50]. Unlike RNNs, which process sequences sequentially, transformers utilize self-attention mechanisms to focus on relevant portions of the data regardless of their position. By learning normal patterns in multivariate time-series datasets, transformers can predict or reconstruct portions of the data based on contextual information from the rest of the sequence.

While others focused on the use of autoencoders [29, 31, 32, 33, 34, 35, 37, 38, 39, 40, 52]. They are frequently employed for their ability to model latent features and reconstruct normal patterns in data. By encoding inputs into a compressed representation and then reconstructing them. Their ability to model non-linear interdependencies makes them particularly suitable for multivariate datasets, where relationships between variables are often intricate and challenging to capture with simpler methods.

Again different paper combine different techniques. Yang et al. [42] uses both and autoencoder and graph transformer network to detect anomalies. The method combines the outputs of both models to determine anomalies. Zhang et al. [34] uses a transformer to support the encoder from the autoencoder to allow attention based encoding.

Feng et al. [53] highlights issues with standard reconstruction approaches stating they are insensitive ti spatio-temporal dependencies and fail to handle heteroscedastic uncertainty. The paper solves this by introducing statistical feature removal and adding a heteroscedastic uncertainty estimation.

Analysis of the 26 selected papers highlights key techniques in comparison to the initial literature explored from the literature surveys. Whilst the initial literature survey focused on algorithms and classical machine learning the systematic literature uncovered the importance of deep learning. Deep learning offers several key advantages for our problem: it effectively handles high-dimensional sensor data, automatically learns relevant features, adapts to the non-stationary nature of cycling data, demonstrates robustness against sensor noise, and captures temporal dependencies.

The nature of reconstruction-based anomaly detection (eq. 3.5-3.6) aligns very well with the mathematical notation of the problem (eq. 1.3). Beyond detection, this approach can also correct anomalies or impute missing values by replacing them with the model's reconstructed output, which would be very

valuable for our approach. However, reconstruction-based anomaly detection relies on the assumption that a model trained on normal data will fail to accurately reconstruct anomalous inputs, thereby exposing anomalies through increased reconstruction errors. This assumption does not hold in the context of physiological time series such as heart rate during exercise. These signals are inherently individual and highly context-dependent. The heart rate response to power output varies substantially between athletes, across fitness levels, and even within the same individual depending on factors such as fatigue, hydration status, and environmental conditions [54]. Consequently, identical heart rate patterns may be entirely appropriate in one context and physiologically implausible in another. We cannot learn or generalize a universal ”normal” heart rate sequence across individuals or sessions. Attempts to reconstruct heart rate, even when incorporating related variables such as power, cadence, and gradient, fail because reconstruction models do not evaluate the plausibility of heart rate given the external conditions. Instead, these models detect deviations from learned temporal patterns within the sequence itself [55]. In this case, however, we define anomalies not by irregular sequence patterns but by violations of expected physiological responses. Reconstruction models inherently overlook such context-driven discrepancies, making them unsuitable for detecting anomalies where correctness depends on the relationship between heart rate and the surrounding effort and conditions.

3.3. Systematic Literature Review on Heart Rate Prediction

No existing anomaly detection paper from the previous literature review satisfies the full spectrum of requirements dictated by our problem formulation. Specifically:

- Personalization: Accurate modeling of physiological signals necessitates dynamic, user-specific representations. A framework should encode this through a function $f(X, u_n)$ where u_n captures temporal adaptations in fitness and physiology. Previous literature approaches assume static, global models that disregard inter-individual variability, a critical omission given the heterogeneity in human physiological responses.
- Physiological Grounding: Many anomaly detection frameworks conflate statistical deviance with physiological implausibility. However, we define anomalies in our domain as violations of a causal, effort-to-response relationship, not merely as statistical outliers. Without modeling the physiological mapping from effort metrics to heart rate, these methods fail to distinguish true anomalies from rare but valid states.

These limitations show that existing anomaly detection methods do not merely fall short; they target the wrong kind of problem. Most of them look for unusual patterns in the data itself, without asking whether those patterns make sense from a physiological point of view. Forecasting and prediction literature is more aligned with this goal because it centers on estimating the value of a variable based on temporal or contextual input features. Regression models offer a structured way to learn these relationships by explicitly modeling how multiple variables interact to produce an expected outcome. These models learn functional relationships between variables and can therefore simulate what heart rate should be under normal physiological conditions. The prediction error then becomes a meaningful, interpretable measure of abnormality. These approaches still allow us to correct faulty values.

For clarity, we distinguish between these terms as follows:

- Reconstruction: Attempts to reproduce the original signal (e.g., heart rate) via compression and decompression, flagging high reconstruction errors as anomalies. However, it ignores physiological context, making it poorly suited for detecting implausible values that look normal but violate known effort-to-response relationships. We explicitly move away from this approach.
- Prediction: Estimating the expected value of a physiological variable (e.g., heart rate) at a given point in time, conditioned on input signals such as power, cadence, speed, gradient, and user-specific embeddings (e.g., fitness level).
- Forecasting: Refers to estimating the future value or values of a variable based on past values of the same or other time series. For example, forecasting HR at time $t + 1$ based HR and other related variables observed at times $t, \ldots, t - n$.
- Regression: Refers to learning the functional relationship between one or more input variables (e.g., power, cadence, speed, gradient, user embeddings) and a continuous output variable (e.g.,

heart rate). The goal is to fit a model that maps inputs to outputs, enabling the estimation of the expected value of the output under varying conditions.

Prediction, forecasting and regression frameworks offer advantages over traditional anomaly detection by enabling contextual reasoning about heart rate values and supporting plausible data imputation. This is crucial for our use case, where we aim not just to detect anomalies based on statistical deviations, but to assess whether measured heart rate is physiologically plausible given observed cycling metrics. Therefore, building on the insights from Section 3.2, we conduct a second systematic literature review, this time focused on heart rate prediction methods. The goal is to identify approaches most suitable for modeling user-specific heart rate dynamics during cycling workouts.

Query Expression

(”heart rate prediction*” **OR** ”predict* heart rate” **OR** ”heart rate forecast*” **OR** ”heart rate regression” **OR** ”HR regression” **OR** ”HR response prediction*” **OR** ”heart rate respons* to exercise” **OR** ”model* heart rate” **OR** ”heart rate estimation” **) AND (** ”workout*” **OR** ”exercise*” **OR** ”physical activit*” **OR** ”cycl*” **) AND ((** ”personalized” **OR** ”user-specific” **OR** ”subject-specific” **OR** ”person-specific” **OR** ”” **) OR (** ”wearable*” **OR** ”fitness tracker*” **OR** ”smartwatch*” **OR** ”activity tracker*” **)) AND**
NOT (”PPG” **OR** ”photoplethysmograph*” **OR** ”ECG” **OR** ”electrocardiogram” **)**
The search was performed on 12/3/2025.

The Query Expression box displays the query used for the review. We design the query to retrieve research on heart rate prediction models while maintaining relevance to our specific context. The first component of the query targets heart rate prediction models, ensuring that the retrieved papers focus on estimating heart rate rather than unrelated physiological metrics. The second component narrows the scope to studies that apply heart rate prediction within the context of exercise, training, or physical activity, as our interest lies in modeling heart rate response to exertion. The third component enforces personalization, a crucial aspect highlighted by multiple papers [8, 56]. They emphasize that heart rate response to exercise varies between individuals, making user-specific modeling essential for accurate predictions. The fourth component filters for studies involving wearable activity trackers, as these devices generate the noisy real-world data relevant to our use case. Finally, we added the last component after the initial search revealed an overwhelming number of studies focused on photoplethysmography (PPG)-based or electrocardiogram (ECG)-based heart rate estimation. By excluding PPG- and ECG-related papers, we refined our results to studies that align better with our focus on predicting heart rate response rather than reconstructing heart rate from sensor data.

We removed conference reviews (n=4) during identification. During the screening step, we review the title, keywords, and abstract of each record to determine whether it meets the selection criteria.

Inclusion criteria:

- The paper proposes an algorithm to predict heart rate based on other activity metrics
- The paper reviews techniques for heart rate prediction

Exclusion criteria:

- The paper is an evaluation between heart rate measuring capabilities of different smartwatches.
- The paper is about detecting different activity types from wearable data.
- The paper uses computer vision to estimate heart rate.
- The paper tries to forecast heart rate, only using heart rate.

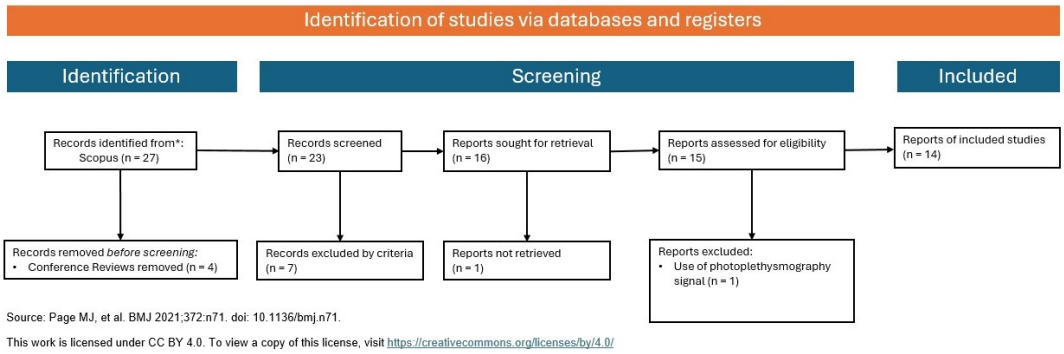

Figure 3.4: PRISMA flowchart illustrating the selection process for papers included in the systematic review on heart rate prediction models relevant to exercise physiology.

After the screening step, we included 14 records and excluded 9 (Fig. 3.4).

The oldest paper retrieved from the query, a 2018 literature survey on heart rate prediction within exercise, provides a comprehensive overview of existing research at the time. Ludwig et al. [8] summarize the state of heart rate prediction methods, but due to its age, it misses many recent and promising approaches. Nevertheless, its classification of different system types used for heart rate prediction remains valuable.

Ludwig et al. [8] defines 4 main tasks of Measurement, Prediction, and Control of Individual Heart Rate Responses to Exercise.

- Approximation: Mathematically, approximation is just a curve fitting problem, which is a specific type of optimization problems. The goal of curve fitting is to find the best solution to a specific problem by finding the minimum of a error function which correlates to the problem.
- Short-term prediction: Using past heart rate and activity metrics, predict the heart rate response to changes in load over a specified time horizon.
- Session prediction: Predict entire heart rate time series only based on activity metrics without using the activity heart rate time series.
- Controlling: Special case of short-term prediction, where the model uses heart rate to regulate intensity and keep it within a specified range.

We find session prediction, where we predict heart rate for an entire workout, the most relevant. While we can technically adapt short-term prediction models for session-level prediction, as noted by Ludwig et al. [8], this often reduces accuracy. If a short-term model relies on past heart rate values, the corresponding session prediction model can use previously predicted heart rate values instead. However, doing so may quickly lead to accumulating prediction errors.

The review then goes into detail about different techniques used for heart rate prediction, namely:

- Artificial Neural Networks (ANNs): At that point in time only used for short periods (1s) of one single time step since longer predictions degraded in accuracy. ANN models often struggle with generalization due to high parameter counts. While they excel in short-term predictions, they tend to overfit when attempting to model heart rate over an entire session.
- Differential Equation (DE) Models: They model the relationship between exercise stress and heart rate response over time. Paradiso et al. [57] uses this model to control a cycling ergo meter to keep users within a desired heart rate range [57]. DE models are advantageous for their interpretability and ability to capture physiological heart rate dynamics, but they often rely on manually tuned parameters.
- Regression Models:Regression models analyze heart rate by establishing probabilistic relationships between heart rate and various influencing factors. These models help identify correlations rather than directly predicting heart rate dynamics.

- Hammerstein and Wiener Models: They describe systems where the relationship between input and output is not purely linear. These models break the system into two parts: a linear part (which handles predictable relationships) and a nonlinear part (which accounts for more complex behaviors). Hammerstein: In heart rate modeling, this setup means that exercise effort is first processed through a nonlinear relationship (e.g., fatigue effects, metabolism), and then heart rate increases in a more predictable way. Wiener: In heart rate terms, this means that your heart rate responds somewhat predictably to workload at first, but nonlinear factors like exhaustion, hydration, and stress modify the final response. Gonzalez et al. [58] found that the Hammerstein-Wiener approach performed best in heart rate modeling within the cycling context [58].

Other interesting fields the review touches upon are parameter-reduced models. These models aim to simplify existing models by reducing the number of parameters while maintaining predictive accuracy. These models are particularly useful for applications in wearables and real-time heart rate monitoring. Parameter-reduced models offer a promising balance between accuracy and simplicity. By minimizing the number of free parameters, these models become more suitable for real-time heart rate monitoring in wearable devices, though their generalizability to different individuals and activities remains a challenge.

We will use the four main tasks (Approximation, Short-term prediction, Session prediction and Controlling) and four main techniques (Artificial Neural Networks, Differential Equation models, Regression models and Hammerstein and Wiener models) discussed in the literature review from Ludwig et al. [8] to further classify the results from the literature survey.

3.3.1. Linear Regression for Session Prediction

Linear regression is a fundamental statistical method that models the relationship between heart rate and exercise-related features by fitting a straight line through the data. It assumes that a weighted sum of the inputs plus an error term expresses the output.

Bychkov et al. [9] takes a simplistic approach by applying linear regression to predict heart rate based on fitness tracker data, using three basic input features: Very Active Distance, Fairly Active Minutes, and Calories. The paper derives the following linear regression formula for predicting heart rate: $HR = \beta_0 + \beta_1 \times VeryActiveDistance + \beta_2 \times FairlyActiveMinutes + \beta_3 \times Calories + \epsilon$. This formula flaws the experiment because including calories burned as an independent variable in this equation is incorrect because Fitbit estimates calorie expenditure based on heart rate [59]. This setup creates a circular dependency, where heart rate influences the calorie estimate, which is then used to predict heart rate. Garrido et al. [60] follows the linear regression approach to see if there is a linear relationship between measured oxygen uptake and heart rate. He found a relation between the two, however measured oxygen uptake is already a very advanced metric which is not directly measurable by wearables, needing in lab testing.

Fang et al. [61] builds a more suitable Bayesian inference-based federated learning for heart rate prediction, integrating autoregression with exogenous variables. The paper uses past heart rate and speed variables to predict the heart rate at the current time step in a linear regression manner: $y_t = \theta_0 + \sum_{i=1}^{p} \theta_i y_{t-i} + \sum_{j=0}^{q} \omega_j z_{t-j} + e_t$.

Since the reviewed papers report reasonable predictive performance, it is possible that these models are still sufficiently reliable to detect anomalies as violations in the expected relationship between activity metrics and heart rate. However it is important to mention that they oversimplify heart rate dynamics by assuming a purely linear relationship, ignoring non-linear dependencies and external influences. Despite these limitations, such linear regression models can serve as a fundamental baseline for evaluating the added value of more advanced predictive approaches.

3.3.2. ANNs for Short-term Prediction

Short term heart rate prediction is a widely tackled field where the papers aim to forecast heart rate over a small horizon based on past heart rate and activity values. This approach establishes an initial baseline to determine whether heart rate will rise or fall over a short period.

Extending from simple linear regression, the Bayesian combined predictor from Zhang et al. [62] improves heart rate prediction by integrating multiple models. The method combines a linear regression

predictor and a neural network predictor into a single framework. While linear regression captures basic linear relationships between variables like power, cadence, and heart rate, it struggles with non-linear and non-stationary relationships. The Bayesian approach dynamically adjusts the weight of each predictor based on its recent accuracy, allowing it to adapt over time. This results in more robust and accurate predictions, especially for multi-step forecasting, where single models tend to accumulate errors. By leveraging both statistical and machine learning methods, this approach balances interpretability with predictive power, making it particularly effective for heart rate modeling in real-world conditions.

Mutijarsa et al. [63] proposes a very simple short-term heart rate prediction model using a Feedforward Neural Network. The Feedforward Neural Network takes heart rate and cadence at a specific time step as inputs and predicts heart rate for the next time step. Whilst achieving high performance this likely comes mainly from having the previous heart rate time step. The paper does not try to estimate heart rate for any longer horizon in the future, since it will likely decay quickly in accuracy.

Both Namazi et al. [14] and Zhu et al. [64] propose a framework for heart rate prediction for short periods of time. Namazi predicts the next 30s, based on 25 min of heart rate data. Singular Spectrum Analysis (SSA) and Copula-based analysis perform the heart rate prediction. SSA is a non-parametric method used for trend extraction, noise reduction, forecasting, and change-point detection in time series data. Copula functions model the dependency structure between variables, independent of their marginal distributions. They are particularly useful for handling complex relationships, including non-linear dependencies. Zhu et al. [64] uses a LSTM-based neural network to predict heart rate in the future. It takes in heart rate combined with activity metrics over a certain window size and is then tasked to predict a heart rate in the future. Predictions further into the future quickly lose accuracy.

Fan introduces a dual-context LSTM model that integrates both immediate exercise context and historical user data for better heart rate prediction [13]. The architecture features personalized embedding layers accounting for static user attributes and dynamic exercise history, significantly enhancing individualized estimations. The paper takes in 300 time steps of heart rate, activity metrics and user characteristics. A first LSTM layer focuses on capturing immediate context. A second LSTM layer focuses on both immediate and historical contexts. Misleading the author claims 250.000 workout records, this claim gives the false impression that the researchers trained the model on over 250,000 workouts. However, later the authors discuss it was only trained on 65 workouts. 250.000 workout records likely references to the individual time steps within all the workouts.

While these methods improve short-term prediction accuracy, they rely heavily on having a valid past heart rate as an input, making them vulnerable if the starting heart rate is erroneous or missing, a likely scenario in anomaly detection. Models primarily extrapolate recent heart rate trends without deeply modeling the physiological response to external effort metrics. For our goal of detecting anomalies based on expected heart rate from activity metrics alone, such models provide limited direct value.

3.3.3. ANNs and DE models for Session prediction

This section is highly relevant to our work. The models discussed here focus on predicting heart rate over an entire session based on other activity metrics, which is exactly the mechanism we require for anomaly detection. In our context, predicting the expected heart rate response from effort metrics (like power, cadence, and gradient) allows us to flag implausible deviations that may indicate sensor errors. For our goal, not just forecasting heart rate, but detecting anomalies based on physiological implausibility, models need to provide session-wide, user-specific, and context-aware heart rate predictions.

Ni et al. [11] performed foundational work. They created and shared a dataset containing over 250 thousand workout records coupled with hundreds of millions of parallel sensor measurements (e.g. HR, GPS) and metadata. They aim to address the challenges posed by this heterogeneous, noisy data, which varies in scale and resolution and exhibits complex interdependencies, making it difficult to model.

The paper introduces two models, Fitrec and Fitrec-Attn, designed for heart rate prediction in workouts. Fitrec-Attn is an encoder-decoder LSTM with attention that forecasts heart rate over the next few time steps during an ongoing workout, enabling users to adjust their pace or intensity in real time. In contrast, Fitrec predicts a user's heart rate and speed profile for an entire workout before it begins, leveraging

learned fitness patterns from past activities and planned workout features. This model employs a two-layer stacked LSTM.

Both are context-aware sequential models that captures the personalized and temporal patterns of fitness data. Both models capture two levels of context information: context within a specific activity, and context across a user's activity history. This dual-context approach has proven to be pivotal in the field, as supported by future studies [6, 12, 13]. By analyzing users' history, the model can create a user profile that accounts for past responses to workout intensity. The model maps this learned fitness to predict how the user will respond in the current workout, allowing for personalization of heart rate prediction.

Qiu et al. [15] develop personalized and course-specific heart rate prediction models for mountain biking. It improves on the paper from Ni et al. [11] by adding course-specific features for better heart rate prediction. It compares different types of heart rate prediction and finds similar result that LSTM models are the best choice for personalized, course-specific heart rate prediction.

As discussed earlier the review of Ludwig et al. [8] said the following about differential equations (DE):

> DE models are advantageous for their interpretability and ability to capture physiological HR dynamics, but they often rely on manually tuned parameters. [8]

Liu et al. [65] notes a similar challenge, they were the first to use non-linear Ordinary Differential Equations (ODEs) for heart rate prediction, later adopted by Nazaret et al. [6]. The ODE enables full-session heart rate prediction rather than short-duration predictions, which is an improvement over existing models. ODEs provide a well-established estimation of a physiological framework for modeling heart rate responses to exercise. The approach maintains interpretability, ensures smooth heart rate transitions, and aligns better with prior research in exercise physiology [56, 57, 66, 67, 68]. The Levenberg-Marquardt algorithm, a numerical optimization method, estimates the model parameters.

Nazaret et al. [6] address the issue of manually tuned parameters by leveraging machine learning to estimate user-specific parameters. Their approach builds on a physiological model [56] designed to describe heart rate dynamics based on activity metrics. This model consists of ODEs with user-specific parameters.

$$\begin{cases} \dot{D}(t) = B \cdot (f(I(t)) - D(t)), \\ \dot{HR}(t) = A \cdot (HR(t) - HR_{\text{min}})^{\alpha} \cdot (HR_{\text{max}} - HR(t))^{\beta} \cdot (D(t) - HR(t)), \\ HR(0) = HR_0, \\ D(0) = D_0. \end{cases} \tag{3.7}$$

In the system (3.7) the function $f(I(t))$ defines the intensity of the activity at timestamp t. User-specific parameter B controls how fast the demand $D(t)$ at time t adapts to $f(I(t))$. The second equation steers the heart rate towards the demand $D(t)$. User-specific parameter A controls how fast the heart rate can change. HR_{min} and HR_{max} are boundaries for the heart rate, while α and β control how difficult it is to reach these boundaries of rest or max heart rate. The third and fourth equation are there for initialization of the ODE.

In the physiological expert model we have 6 user-specific parameters, namely, A, B, α, β, HR_{min} and HR_{max}. Each user has unique parameters, which the original paper [56] estimates through lab testing. Nazaret et al. [6] found it is not feasible to learn these specific user parameters per user from laboratory testing for user products. Instead the paper transforms the ODE equation into an equation where all parameters are in function of the users fitness representation z. It also add component $g(W)$ and $h(t)$ for accounting weather and fatigue build up respectively.

$$\begin{cases} \dot{D}(t) = B(z) \cdot (f(z, l(t)) \cdot g(W) \cdot h(t) - D(t)), \\ \dot{\text{HR}}(t) = A(z) \cdot (\text{HR}(t) - \text{HR}_{\min}(z))^{\alpha(z)} \cdot (\text{HR}_{\max}(z) - \text{HR}(t))^{\beta(z)} \cdot (D(t) - \text{HR}(t)), \\ \text{HR}(0) = \text{HR}_0(z), \\ D(0) = D_0(z). \end{cases} \tag{3.8}$$

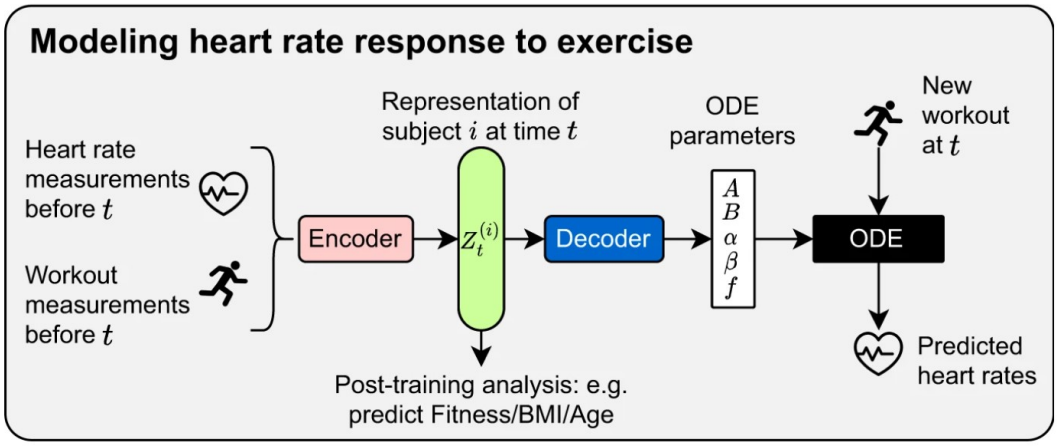

Figure 3.5: Overview of the ODE-based heart rate prediction model proposed by Nazaret et al., which combines exercise intensity inputs with personalized physiological parameters to model heart rate response. Adapted from [6]

The model works by an encoder-decoder architecture where the encoder learns a fitness representation z of the user and the decoder then turns z into the specific ODE parameters (Fig. 3.5). In this manner all users can have global shared neural networks while using user-specific fitness representations for user dependent variables. The function $f(z, I(t)$ (drive function) is also learned via a neural network (multi-layer perceptron with two hidden layers) that maps activity metrics and the users fitness representation z towards an estimated intensity.

The model works by passing previous workouts from a user, sorted by starting date, into the encoder and decoder to determine the ODE parameters. Over time newer workouts dynamically update the learned parameters. Once we processed all workouts from that user from the train set, it estimates an initial heart rate, ODE parameters, and processes the current workout activity metrics through the drive function to generate an intensity time series. With the intensity time series and the initial heart rate, the method solves the ODE to generate an heart rate prediction. After each prediction during training, the method updates the encoder and decoder using L2 loss from the heart rate prediction, combined with a regularization term.

Kayange et al. [12] builds further on the paper of Nazaret et al. [6] for developing a personalized heart rate model. The proposed approach integrates a physiological model using Dynamic Bayesian Networks (DBNs) to capture heart rate dynamics during workout sessions. The state transition function (Eq. 3.9) models how heart rate evolves over time based on exercise intensity and other factors. While the emission model (Eq. 3.10) relates the latent physiological state to the observed heart rate, accounting for uncertainty.

$$P\left(x_t \mid x_{t-1}, \mathcal{S}_t\right) = \mathcal{N}\left(x_t \mid f_{\text{trans}}(x_{t-1}, \mathcal{S}_t), \sigma^2\right), \tag{3.9}$$

$$Y_t = f_{\text{em}}(X_t) + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, \sigma^2), \tag{3.10}$$

A LSTM network processes historical workout data to learn user-specific physiological patterns which helps adjust the DBN parameters. Predicting heart rate directly from latent representations might work well for short-term heart rate forecasting. However, for long duration workouts, DBNs help maintain consistency by explicitly modeling heart rate evolution over time, considering factors such as fatigue and other environmental factors.

After reviewing the code provided by the author, we found major inconsistencies with the paper: key methods like personalized heart rate prediction and adaptive feature selection were missing, test and training data were improperly mixed, and the model only predicted short sequences instead of full workouts. The author later clarified that the paper described a conceptual framework rather than the

exact implementation and acknowledged that the validation setup could lead to biased performance estimates.

These session-based models are more aligned with our project goals. By predicting heart rate over an entire workout using contextual variables, they create a foundation for detecting deviations between measured and expected heart rate at any point in time, a critical capability for prediction-based anomaly detection. Despite some methodological weaknesses in the reviewed papers, the general approach of combining effort metrics, user-specific embeddings, and dynamic modeling offers the most promising path forward.

We select the models by Nazaret et al. [6] and Kayange el al. [12] as the most useful for our work, because they uniquely combine physiological interpretability with automatic personalization. Both model the heart rate response to exercise, enabling predictions grounded in known HR-effort dynamics. Crucially, they estimate user-specific parameters from data, allowing the predicted heart rate to reflect individual differences in fitness and response, which is essential for our anomaly detection framework. Unlike other models that simply forecast heart rate for short periods of time, the two papers [6, 12] approaches allow us to determine whether a measured heart rate value is physiologically plausible given the effort, making them directly applicable to detecting anomalies in wearable data.

3.3.4. Research Gap

Through the literature review, it became clear that linear regression models offer a simple and interpretable approach to heart rate prediction by modeling the relationship between input variables such as power and cadence using fixed weights. While this simplicity makes them attractive, particularly as a baseline for comparison, it also limits their ability to capture the complex, non-linear, and context-dependent nature of heart rate dynamics. Factors like fatigue, hydration, and environmental conditions influence heart rate response to effort, which also shows time-dependent behaviors such as lag and adaptation. These introduce non-stationarity that linear models cannot address. Moreover, the relationship between effort and heart rate is highly individual-specific, making it difficult for a single global model to generalize across users. Despite these limitations, linear regression remains a valuable benchmark. Its transparency and low computational cost make it useful for understanding baseline performance and for highlighting the added value of more complex, personalized, and physiologically grounded models. In this context, we do not expect linear regression to match state-of-the-art performance, but it serves as a meaningful reference point for evaluating model improvements.

The literature also made it clear that while short-term models are often the most accurate for predicting the next heart rate value, they heavily rely on having a valid heart rate input as a starting point. This because they use autoregression, where future values depend on past values of itself. These models typically predict whether heart rate will increase or decrease based on recent trends, meaning that if the input heart rate is already erroneous or missing, the entire prediction quickly becomes unreliable. Effective anomaly detection requires models that account for historical trends, individual adaptation, and multi-modal dependencies rather than just extrapolating from recent data points.

Finally, reviewing session-wide heart rate prediction models [6, 12] revealed that although these models can accurately capture physiological trends over full exercise sessions, they are not leveraged for anomaly detection. This leaves a gap where models with strong predictive power are not used to flag deviations from expected patterns. Therefore, in Chapter 4, we design a prediction-based anomaly detection framework specifically tailored to these challenges.

4

Methodology

4.1. Experimental Objectives

In Chapter 3, we identified that traditional anomaly detection methods fail because they do not model the physiological plausibility of heart rate (HR) responses to exertion. To address this, we develop a prediction-based anomaly detection framework tailored to the cycling domain. We design the experiments in this chapter to answer the research questions and systematically build and test the framework.

The first objective is to get a heart rate prediction model that can accurately estimate heart rate from exertion metrics (e.g., power, cadence) and user-specific profiles, following equation 1.4. This predictive capability is foundational to the anomaly detection framework, without a reliable model of expected heart rate, it is impossible to determine whether observed deviations reflect true anomalies or normal variability. Therefore, this step aims to identify a model with minimal mean absolute error (MAE) and high correlation to ground truth heart rate, providing a robust basis for downstream anomaly detection and reconstruction tasks.

The second objective is to construct a clean evaluation dataset by first removing as many real anomalies as possible and then injecting synthetic anomalies, specifically point, subsequence, and time series anomalies, under controlled conditions. This process creates a known ground truth, which is critical for systematically evaluating both the detection and reconstruction capabilities of the proposed framework. The goal is to simulate realistic sensor failures while maintaining full control over anomaly type, location, and severity, enabling rigorous, quantitative performance benchmarking.

The third objective is to implement detection and reconstruction methods that operate by comparing measured heart rate values against model predictions. Detection involves identifying time segments where the prediction error exceeds a defined threshold, following equation 1.5. Reconstruction aims to replace these anomalous segments with the model's predicted values, following equation 1.6. This experiment tests the core hypothesis of the framework, that physiologically implausible values can be reliably isolated via prediction error, and that the model can generate plausible replacements. Our goal is to evaluate how accurately we can detect anomalies and how effectively we can reconstruct corrupted data.

4.2. Heart Rate Prediction Models

Our literature review (Sec. 3.3) identified three relevant modeling approaches suitable for our application. The first is a linear regression model, similar to those used by Bychkov et al. [9], Garrido et al. [60] and Fang et al. [61], serving as a baseline implementation to compare the other models against. Additionally, the models from Nazaret et al. [6] and Kayange et al. [12] predict heart rate over an entire session based on activity metrics and are thus very suitable for our approach. This is useful for us because it lets us compare the predicted, or in other words expected, heart rate to the measured heart rate and detect implausible deviations as anomalies. We will evaluate the performance of all three where we use the linear regression model as a baseline to evaluate the other two methods against.

4.2.1. Baseline

To establish a foundational benchmark in our heart rate prediction framework, we employ a linear regression model, similar to Bychkov et al. [9], Garrido et al. [60] and Fang et al. [61]. This model serves as a first step in quantifying the relationship between standardized activity metrics and heart rate. Despite the inherent non-linearity in cardiovascular response to exercise intensity, linear regression provides a transparent, interpretable, and computationally efficient approximation of heart rate dynamics. By incorporating 3 minute rolling averages, the model captures sustained physiological workload.

To ensure the model remains robust and interpretable, we rigorously controlled for multicollinearity by selecting features with a low Variance Inflation Factor (VIF). High VIF values indicate redundant or highly correlated predictors, which can lead to unstable regression coefficients and misleading conclusions. By eliminating such redundancy, our model isolates the true impact of each variable on heart rate, ensuring that features like power, which should logically be the dominant predictor, are not overshadowed by spurious correlations from secondary metrics like speed.

The formula we feed into the linear regression model is the following:

$$HR(t) = \beta_0 + \frac{1}{180}\sum_{i=0}^{179} \beta_1 \cdot power_{t-i}^{std} + \beta_2 \cdot cadence_{t-i}^{std} + \beta_3 \cdot gradient_{t-i}^{std}$$

While we acknowledge the limitations of a strictly linear approach such as its inability to account for heart rate lag effects, fatigue accumulation, and non-linear physiological thresholds, this baseline provides a crucial reference point for evaluating more advanced models.. The insights derived from this model help validate feature selection, establish expected error margins, and ensure future improvements are empirically justified rather than arbitrary.

4.2.2. Session Heart Rate prediction

Nazaret et al. [6] predicts heart rate using a physiological model based on ODEs, where parameters are dynamically adjusted by a neural network. The model learns personalized heart rate responses from past workouts and predicts the entire heart rate curve for new sessions by integrating workout intensity and environmental factors. Contrary, Kayange et al. [12] predicts heart rate by combining a DBN with an LSTM. The LSTM learns user-specific patterns from workout history, while the DBN models how heart rate evolves over time based on workout intensity and other inputs. Together, this setup predicts the full heart rate profile during workouts, handling temporal changes and individual differences. The later study [12] claims higher accuracy as they compare with Nazaret et al. [6] However, because of previously noted paper mismatches (Sec. 3.3.3), we need further study to determine which model is more accurate.

We expanded both papers to incorporate a validation set for guiding training and resolved initial issues with the code of Kayange et al. [12] We compared both models by running them on the dataset. Both received the same train, validation and test activities with identical embedding sizes for both the subject and the encoder.

In addition to evaluating the original model proposed by Nazaret et al. [6], we also introduce a parameter reduced version of their approach. This modified version removes the barriers learned as HR_{min} and HR_{max}, as a result also removing α and β, as well as the weather component which was unused because of unavailability of data, resulting in the final formula:

$$\begin{cases} \dot{D}(t) = B(z) \cdot (f(z, l(t)) \cdot h(t) - D(t)), \\ \dot{\mathsf{HR}}(t) = A(z) \cdot (D(t) - \mathsf{HR}(t)), \\ \mathsf{HR}(0) = \mathsf{HR}_0(z), \\ D(0) = D_0(z). \end{cases} \tag{4.1}$$

Three key considerations motivate this reduced version. First, Ludwig et al. [8] highlights the value of parameter-reduced models as they strike a balance between predictive accuracy and model simplicity. Second, reducing the number of user-specific parameters from six to two (A and B) simplifies the

learning task. It can adjust the fit of A and B better around the users fitness representation z. Finally, the inclusion of fixed HR_{min} and HR_{max} values may inadvertently suppress useful anomaly signals. For instance, when sensor error causes an athlete's predicted heart rate to exceed HR_{max} due to high power, the original model caps the predicted heart rate at HR_{max} reducing the difference between measured and predicted heart rate. The same holds if we predict their HR_{max} too low. The prediction caps while the measured heart rate will be larger, showcasing high error whilst there is no anomaly. This constraint weakens the anomaly detection mechanism by masking precisely the deviations we aim to detect. Removing these boundaries allows the model to produce uncapped predictions, better exposing implausible measurements. This last reason opts that even for equal or worse performance this new model will still be better to detect anomalies.

We tested the three models against each other and compared them with the baseline. As described earlier (Sec. 4.3) MAE median error holds more value against MAE mean error. Another important metric in our evaluation is the Spearman correlation coefficient, which measures how well the predicted heart rate captures the overall shape and trend of the true heart rate response. This is particularly relevant in our context, where the model isn't just used for prediction in isolation, but serves as the foundation for anomaly detection. A high Spearman correlation indicates that the model preserves the correct relative ordering and dynamics of heart rate over time, meaning it reliably tracks when heart rate should rise, plateau, or fall. This matters because anomaly detection relies on deviations from expected patterns; if the model captures the correct trend, even if it's off by a some BPM, it still can serve well for our anomaly detection framework.

4.3. Dataset

The original cycling dataset provided by Wahoo Fitness contains an enormous amount of variables and sensor data. For our research we use the following fields per workout: activity ID, user ID and start time. Each workout tracks the following variables per time step: heart rate, power, cadence, speed, gradient, and altitude. Resulting in an array of measurements equaling the workout length. Per time step one can see the heart rate in beats per minute, the power in watts, the cadence in rotation per minute, the speed in meters per second, the gradient in percentage and the altitude in meters.

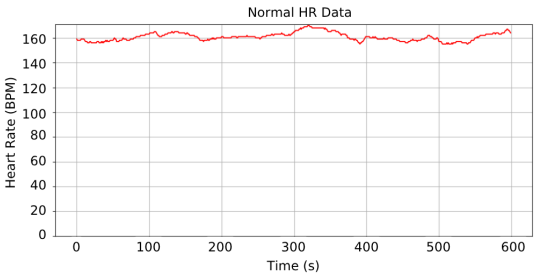

Figure 4.1: Heart rate during a 10-minute segment of a cycling workout. This data represents a normal, non-anomalous heart rate response over time, taken from the same workout and time interval as Figure 4.2.

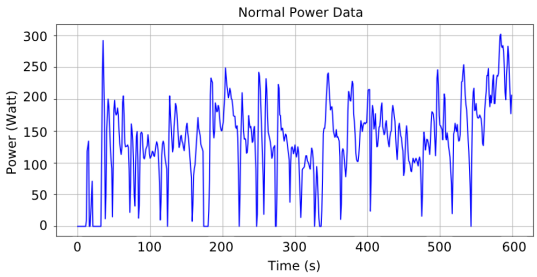

Figure 4.2: Power during a 10-minute segment of a cycling workout. This data represents a normal, non-anomalous power data over time, taken from the same workout and time interval as Figure 4.1.

Figure 4.1 and Figure 4.2 provide a visualization of normal, non-anomalous data taken from the same workout at the same time. heart rate data exhibits a dynamic nature during workouts, as it responds gradually to changes in intensity, making it dependent on its previous value HR_{t-1}. Power measures the immediate force output by the cyclist, which is not dependent on past values and can fluctuate instantaneously in response to changes in effort.

Even though the goal of our research is anomaly detection, we deliberately begin by removing all anomalous workouts from the dataset. This may seem counterintuitive, but it's a critical step. To train a heart rate prediction model (4.2) that reliably captures normal physiological behavior, we need data that reflects how heart rate typically responds to changes in effort. If we were to include anomalous data during training, the model could learn incorrect patterns, making it harder to distinguish between normal and abnormal later on. We filter out noisy or unrealistic workouts upfront to ensure the model trains

on representative, high-quality data. After training, we inject known anomalies ourselves. However, this filtering step might not be perfect. Some workouts that contain anomalies may still slip through our initial cleaning process and remain in the dataset with a non-anomalous label. As a result, the model might correctly flag them as anomalous later on, but since their ground-truth label is normal, these instances will appear as false positives during evaluation. This is a known limitation and reflects the inherent difficulty of guaranteeing a fully clean training set in real-world physiological data.

We base the filtering procedure on the data selection approach by Nazaret et al. [6], aiming to retain only physiologically realistic and representative workout sessions. The filtering process excluded extreme activities where the average heart rate fell below 45 bpm or exceeded 215 bpm, or where the total distance traveled was less than 2.5 kilometers. We trim missing values at the start and end of activities until all relevant sensors provided complete data entries. The method allows a maximum of 10 seconds of consecutive missing data per activity, and data gaps within this limit were linearly interpolated. To ensure realistic heart rate dynamics, the dataset included only activities lasting between 15 minutes and 2 hours. This range helps avoid biases from shorter activities, often reflecting warm-ups, failed workouts, or non-representative activities, and longer activities, where external factors like nutrition and hydration play a more significant role [69, 70, 71, 72]. These effects cause the same external effort to produce different heart rate responses over time, since we have no model input for these contextual variables, breaking the functional relationship the model aims to learn. Finally, the dataset included only users who completed at least 10 workouts that met these criteria, ensuring the derivation of user-specific parameters from a diverse set of activities. We extend the filtering approach by Nazaret et al. [6] by restricting the dataset to workouts recorded using reliable indoor trainers and specific chest straps that Wahoo Fitness identified as providing the most accurate and trustworthy measurements for power and heart rate. We collected 21,218 workouts from 1,435 unique users, with each workout averaging 4,025 time steps in duration. This represents only a fraction of the available data of Wahoo Fitness. We based our decision on findings from related work [11, 12], where similarly sized datasets proved sufficient for training accurate models. Thus, our final dataset strikes a balance between quality, quantity, and practical feasibility.

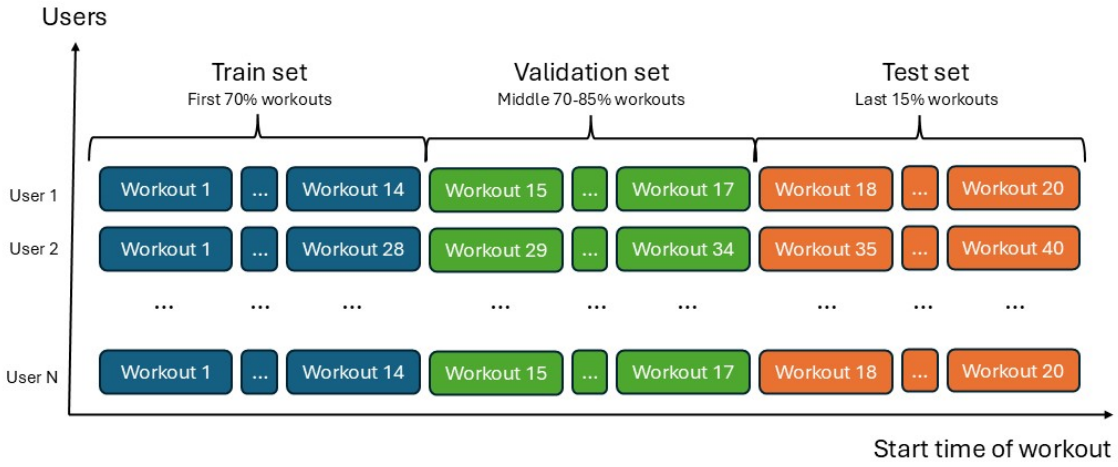

Figure 4.3: Per-user temporal split of cycling workouts into training, validation, and test sets. For each user, workouts are ordered by start time, with the first 70% assigned to training, the next 15% to validation, and the final 15% to test.

For training and evaluation, we split the data into training, validation, and test sets individually for each user. For every user, we assign the first 70% of their workouts, ordered by activity start time, to the training set, the next 15% to the validation set, and the final 15% to the test set, as shown in Fig. 4.3. Ending with 14.318 training workouts, 3.116 validation workouts and 3.820 test workouts.

4.3.1. Creating Anomalies in the Dataset

There is no ground truth for whether data was anomalous or not. We hold the assumption that most data is non-anomalous and aim to create anomalies ourself in the data to see if we can detect those. To do

meaningful research we need to recreate realistic anomalies, that could occur in real-world scenarios. Based on the scenarios where anomalies commonly occur, as discussed in Section 2.4, Wahoo Fitness translated these insights into a concrete list of anomaly types that can be systematically injected into our dataset. We divided them into three categories. Point anomalies occur when the power or heart rate value abruptly shifts at a single timestamp. Subsequence anomalies, where power or heart rate values shift for a variable amount of consecutive time ranging from 1 minute to ten minutes. Lastly, time series anomalies occur when the entire time series shifts for an entire activity. This categorization follows the taxonomy of Blázquez et al. [10], earlier discussed in Section 1.3.

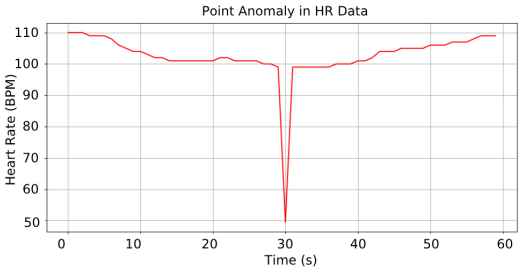

Figure 4.4: Point anomaly - Heart rate measurement is half the value of the true heart rate

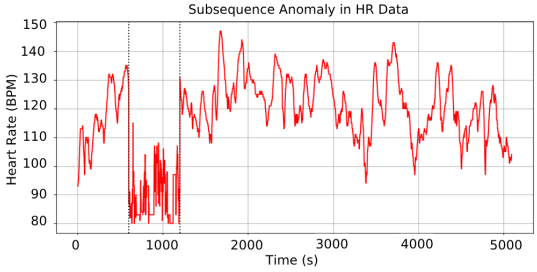

Figure 4.5: Subsequence anomaly - Heart rate measurements matches cadence

- Point anomalies
  - Heart rate measurement is half the value of true heart rate (Fig. 4.4)
  - Heart rate measurement matches cadence
- Subsequence anomalies
  - Heart rate lags behind during a rapid increase in heart rate.
  - Heart rate measurement matches cadence (Fig. 4.5)
- Time series
  - Heart rate measurement is half the value of true heart rate
  - Heart rate measurement matches cadence
  - Power data is half the value of true power
  - Power data overestimates true power by 20%.

The method introduces anomalies in 20% of the test set to simulate a realistic scenario where most workouts remain unaffected. The anomalies are only allowed to occur outside of 1 minute from each other. This is to isolate their effects from each other.

Since we evaluate the model using only anomalies we injected ourselves, any real-world anomaly that slipped through preprocessing (Fig. 5.4-5.6) won't be labeled as such. So even if the model correctly flags it as anomalous, we penalize it as a false positive. This artificially lowers our measured precision, even though the model is behaving correctly. While this is an important limitation to acknowledge, it should only affect a small portion of the dataset thanks to the extensive preprocessing and filtering steps applied upfront. Nonetheless, we must consider this factor when interpreting evaluation metrics.

4.4. Anomaly Detection

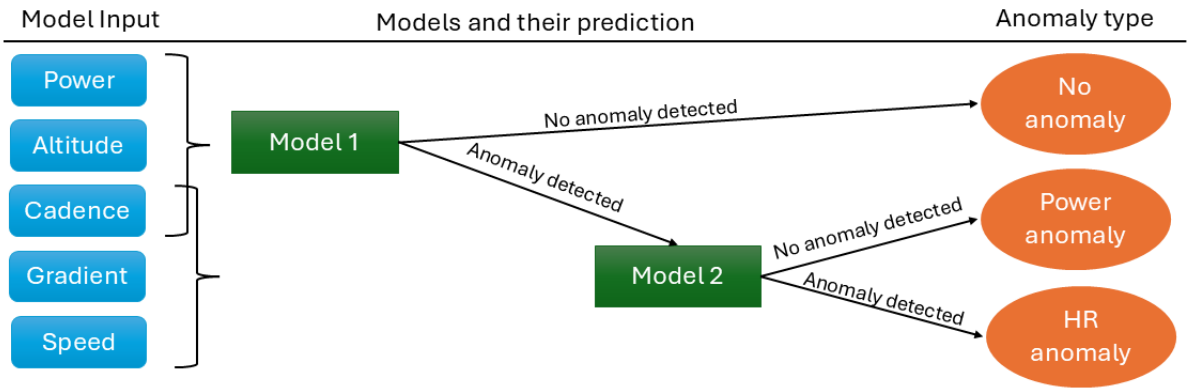

Figure 4.6: Dual-model approach for anomaly source attribution. Model A predicts heart rate using power, while Model B relies on different metrics. By comparing anomaly detections from both models, we determine whether the anomaly originates from the heart rate sensor or the power input.

We train the model on normal data, then introduce anomalies in 20% of all test workouts. We previously described the specific types of anomalies in Chapter 4.3.1. Our model predicts heart rate based on sensor input, with power identified as the most effective predictor of heart rate changes. However, we face a challenge when detecting an anomaly in the heart rate reconstruction. A key challenge is that when we detect an anomaly in the predicted heart rate, we don't immediately know whether the problem comes from the heart rate data itself or from the input power data used to predict it. This distinction is necessary as discussed in the context of the problem in the introduction (Eq. 1.3). To resolve this, we use a second, independent model that predicts heart rate based on different inputs, as stated in equation 1.5. If both models detect an anomaly, the heart rate data is likely at fault, since even a model that doesn't rely on power finds it implausible. But if only the first model flags an issue, the problem originates from the original power input, not the heart rate itself, as otherwise the model which is not using power should identify an anomaly as well. This approach helps us isolate which signal is causing the anomaly. Figure 4.6 visualizes this.

The model's ability to detect anomalies depends on the time scale at which the anomaly occurs. As discussed in the introduction we will follow the outlier types identified by Blazquez-Garcia et al. [10]. For the evaluation we will tackle the performance for each anomaly type individually. It is important to note that we make no changes to the model. The model is capable to detect different types of anomalies without finetuning for a specific anomaly type.

To ensure robustness and mitigate the effects of random variation from creating random anomalies in the test set we conduct 10 experiments with different random placed anomalies. For each metric we compute the average, minimum and maximum scores across all these runs.

The methods described in this section were originally devised for this thesis. While it draws conceptual inspiration from reconstruction-based anomaly detection approaches, particularly the use of data-driven error thresholds for identifying anomalies, its design choices, including the use of different thresholds for different temporal patterns and jump conditions are novel and application-specific. These custom formulations are necessary to move beyond standard reconstruction anomaly detection and capture more realistic anomalies aligned with how physiological deviations manifest over time.

4.4.1. Point Anomalies

Point anomalies refer to anomalies of isolated single unit length (i.e. 1 second anomaly). We detect these by sliding a three-point window over the residual error signal (i.e., the difference between predicted and true heart rate). We mark a point as anomalous if its deviation from neighboring error values exceeds three times the local standard deviation and if it significantly exceeds the surrounding context. This lets us detect singular, high-magnitude outliers caused by brief sensor errors or physiological artifacts.

This method is intentionally conservative, prioritizing precision over recall. It assumes that model forecasts are locally smooth and that genuine point anomalies will produce statistically unlikely errors that do not align with the trajectory of neighboring samples.

4.4.2. Subsequence Anomalies

Subsequence anomalies refer to deviations in heart rate that persist for a defined interval but do not span the entire workout. We detect anomalies based on the absolute error between predicted and measured heart rate values using two mechanisms:

- Threshold-based detection: We flag anomalies when the error exceeds a threshold ($\mu + 3 \cdot \sigma$) based on training data statistics. To improve robustness, we tolerate brief error drops below the threshold, preventing short correct segments or isolated point anomalies from splitting longer anomalous sequences.
- Abrupt change detection: We use sudden large increases in absolute error as indicators to capture abrupt onsets or terminations of anomalies.

4.4.3. Time Series Anomalies

Time series anomalies refer to sustained, systematic deviations between predicted and measured heart rate throughout an entire workout. We quantify these anomalies using the mean absolute error (MAE) between predicted and observed heart rate values over the full time series.

To determine the origin of the anomaly, whether it comes from heart rate or power sensor, we use two separate prediction models:

- Model 1 (with power): Uses cadence, power, and altitude as input.
- Model 2 (without power): Uses gradient, cadence and speed.

We label anomalies using a two-stage decision process:

1. We consider the workout anomalous if Model 1's prediction error exceeds the detection threshold.
2. We attribute the anomaly to heart rate data if Model 2 also exceeds the threshold at the same time points. We attribute the anomaly to power data if only Model 1 shows high residuals.

To convert model prediction errors into anomaly labels, we implement two complementary thresholding strategies: a statistical thresholding approach based on residual distributions and a performance-optimized approach using Monte Carlo optimization.

Static Thresholding

We define the threshold as: $\tau = \mu_\epsilon + k \cdot \sigma_\epsilon$, where μ_ϵ is the mean and σ_ϵ is the standard deviation of the residuals. We sweep values of k between 2.0 and 4.0 in 0.5 increments and report performance metrics (precision, recall, F1) for each.

Performance-Driven Thresholding via Monte Carlo PR-AUC Optimization

To optimize threshold selection based on detection performance, we perform a Monte Carlo–style evaluation consisting of ten repeated runs. In each run, we inject anomalies randomly into the test set. For each run, we compute the precision-recall curve and identify the threshold that maximizes the area under the curve (PR-AUC). We define the final threshold as the average of the optimal thresholds across all runs. This threshold is then evaluated on new, unseen anomaly placements to assess generalization performance.

Between the two thresholding methods, we adopt the one that demonstrates superior generalization performance on held-out data, thereby ensuring optimal detection accuracy in real-world scenarios. For the selected method, we document its precision, recall and F1-score to provide a transparent evaluation of its effectiveness.

4.5. Heart Rate Reconstruction

4.5.1. Point Anomaly Reconstruction

Reconstructing anomalous segments in our data can follow equation 1.6 due to the high correlation between the model predictions and the original, uncorrupted sequences. For point anomalies, our baseline method is a simple interpolation over the anomalous region, which provides a quick but context-agnostic fix. Hypothesis is that this will work well for point anomalies and will be hard to outperform. A more model-driven approach replaces the anomaly directly with the model's forecast, relying on its learned expectation of normal behavior but without considering error in the models forecast. Finally, we apply a more sophisticated strategy where the forecast is not just inserted blindly but adjusted so that the resulting prediction error aligns with the error levels observed immediately before and after the anomaly. This produces smoother transitions and better preserves the signal's local temporal dynamics. This technique builds on the high correlation obtained from the model.

4.5.2. Subsequence Anomaly Reconstruction

Similarly for subsequence anomalies, we compare three reconstruction strategies. The baseline approach again uses simple linear interpolation across the anomalous region. However, compared to point anomalies, this method is expected to perform poorly because longer durations and increased variability within these segments challenge interpolation. The second approach again leverages the forecasting model to directly replace the anomalous section with its predicted values. Unlike interpolation, this model-driven reconstruction accounts for intra-sequence variation, producing more realistic estimates by incorporating contextual input features. Lastly, we introduce the adjusted forecast method again that modifies the model's output to align its prediction error with the error levels observed immediately before and after the anomaly. This technique smooths transitions at anomaly boundaries and better preserves local temporal dynamics. It builds on the strong temporal correlation structure learned by the model, resulting in more coherent reconstructions.

4.5.3. Time Series Anomaly Reconstruction

Time series anomalies pose a greater challenge because they affect the entire workout, leaving no reliable non-anomalous context within the sequence. In these cases, we rely entirely on the model's forecast as the replacement for the corrupted signal. By doing so, we anchor the reconstruction to the model's learned expectation of what a typical workout should look like. This has the effect of pulling the overall reconstruction error of the anomalous activity back toward the model's global mean error. For full-session anomalies, interpolation is not a viable option due to the absence of valid context within the session. Instead, we replace the anomalous heart rate signal with the user's average heart rate, computed across all their previous sessions.

5

Results

5.1. Experimental Setup

This chapter evaluates the methodology developed in Chapter 4, with each experiment aimed at answering specific aspects of the research question.

We first evaluate the predictive models by measuring how accurately they can predict heart rate (HR) during clean, non-anomalous workouts. This establishes whether the predictive foundation is strong enough to support anomaly detection.

We then assess anomaly detection performance by applying the models to datasets with injected anomalies, examining detection accuracy across different types: point anomalies, subsequence anomalies, and full session anomalies. This evaluates whether deviations from predicted heart rate reliably indicate sensor errors rather than normal physiological variability.

Lastly, we measure the quality of reconstruction, replacing corrupted heart rate values with predicted values and assessing the resulting error reduction. We do this on the identified anomalies from the anomaly detection experiment and on the the ground truth of the injected anomalies, this to assess the total pipeline and the reconstruction mechanism itself. This demonstrates whether the model not only detects errors but can also realistically correct them.

5.2. Heart Rate Prediction Models

This section evaluates the performance of the heart rate prediction models developed in Section 4.2. The goal is to assess whether the model can generate accurate, physiologically plausible heart rate estimates based on exertion metrics and user-specific embeddings, as required for anomaly detection in future steps.

5.2.1. Model Comparison

In this subsection we compare the baseline and prediction-based models to determine which achieves the lowest MAE and highest correlation with true heart rate values. This directly tests the predictive quality required for anomaly detection.

We fitted our baseline linear regression model on 80% of the workouts and tested it on the remaining 20%. The final fitted formula was the following:

$$HR(t) = 131.16 + \frac{1}{180}\sum_{i=0}^{179} 18.95 \cdot power_{t-i}^{std} + 2.54 \cdot cadence_{t-i}^{std} + 0.58 \cdot gradient_{t-i}^{std}$$

VIF values (power = 1.94, cadence = 1.74, gradient = 1.15) remained low to show low correlation between the standardized rolling averaged activity metrics. The high weight for power highlights the importance of power as a heart rate predictor once more.

	MAE mean	MAE median [IQR]	RMSE median [IQR]	MAPE median [IQR]	correlation median [IQR]
Baseline Linear Regression	14.287	12.321 [8.875-17.791]	16.591 [11.164-20.316]	8.14% [5.44%-12.58%]	0.627 [0.484-0.741]
Nazaret et al. [6]	10.927	9.307 [6.679-13.441]	13.493 [8.575-16.486]	5.79% [3.92%-8.67%]	0.736 [0.563-0.840]
Kayange et al. [12]	14.275	13.182 [10.655-18.027]	17.214 [13.336-21.407]	8.45% [6.32%-12.52%]	0.166 [0.069-0.392]
Nazaret et al. [6] Parameter reduced	9.595	8.185 [5.979-11.441]	11.747 [7.728-13.907]	5.18% [3.68%-7.68%]	0.812 [0.697-0.886]

Table 5.1: Performance comparison of different models on heart rate prediction.

	Training time	Inference time (per new workout)	Number of epochs	Average time per epoch
Baseline Linear Regression	00:00:02	0.015 ms	-	-
Nazaret et al. [6]	2-06:52:36	58.115 ms	21	2:36:47
Kayange et al. [12]	4-19:37:54	7.592 ms	6	19:16:19
* GPU accelerated	2:42:10	2.356 ms	6	27:02
Nazaret et al. [6] Parameter reduced	1-13:00:05	27.487 ms	21	1:45:43

Table 5.2: Training and inference time analysis for each model.

The average error of the baseline in prediction was 14.28 beats per minute, but more meaningful for a realistic comparison is the median error of 12.32 beats per minute (Table 5.1). Training and inference time (Table 5.2) were, as expected, extremely low.

The results clearly show that the model by Nazaret et al. [6] significantly outperforms both the baseline linear regression and the Kayange et al. [12] model across all key metrics. With a median MAE of 9.307 and a correlation of 73.6%, it achieves notably better accuracy than the baseline, which sits at 12.321 MAE and 62.7%. This translates to an improvement of over 3 BPM in median error. The results from the Kayange et al. [12] however showcase a low performance having similar results to the baseline linear regression approach. Because of the DBN-LSTM model from Kayange et al. [12], a lot of noise enters the estimation, resulting in a very low correlation score.

As seen in Table 5.1, the model with the removed HR_{min}, HR_{max}, α and β parameters, referred to as the parameter-reduced version of Nazaret et al. [6], not only outperforms the original in all performance metrics, but also leads to a notably higher Spearman correlation (0.812 vs. 0.736). This suggests that removing the heart rate boundary constraints, reducing the total number of user-specific parameters from six to two, allows the model to better capture the dynamic shape of the heart rate signal across different users. This improved flexibility is particularly valuable for anomaly detection, where capped predictions can mask real deviations caused by anomalous input metrics.

The DBN-LSTM model, as implemented, is computationally expensive when executed on CPU. This is primarily due to the recurrent nature of LSTMs and the dense layers that require large matrix multiplications at each time step. On CPU, these operations are inherently slow because they are not optimized for the highly parallel computations deep learning models rely on. However, running the DBN-LSTM on a GPU yields a significant speedup. The reason is simple: GPUs efficiently handle large-scale parallel computations, especially matrix operations in LSTM layers and dense neural networks. This allows the DBN-LSTM to fully leverage the GPU architecture, drastically reducing runtime. In contrast, the ODE-based model does not experience the same performance gain when moving from CPU to GPU. Although all model components run on the GPU, the core computational bottleneck remains the numerical ODE solver (torchdiffeq.odeint). ODE solvers require sequential integration over time, where each time step depends on the solution of the previous one. This recursive nature limits the ability of the

GPU to parallelize the computation across time steps. While the ODE solver parallelizes across the batch dimension, it processes the time-stepping loop sequentially. As a result, even when executed on a GPU, the ODE model shows minimal speed improvement compared to the CPU version.

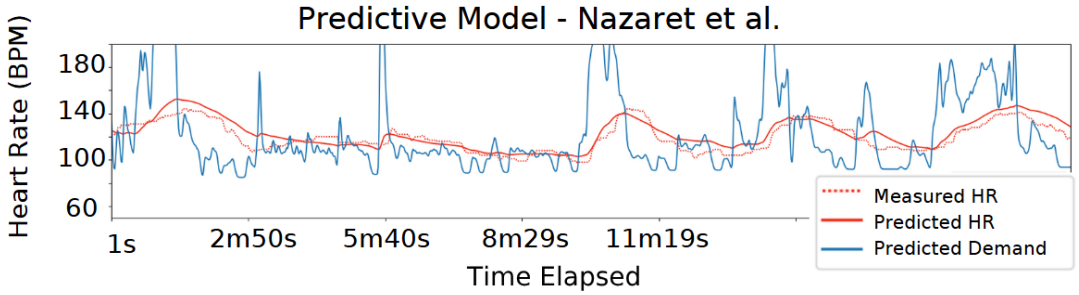

Figure 5.1: Output of the predictive model proposed by Nazaret et al. [6]. The figure shows the measured and model's predicted heart rate and demand over the same time period.

Figure 5.1 clearly demonstrates Nazaret et al. [6] model's ability to predict heart rate with high accuracy. The predicted heart rate curve closely follows the measured heart rate throughout the workout, capturing both gradual trends and rapid fluctuations. This alignment indicates that the model successfully learns the underlying relationship between input variables and heart rate response.

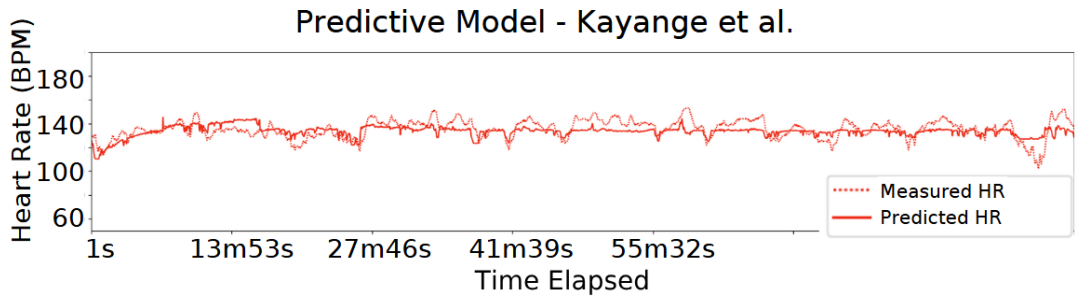

Figure 5.2: Output of the predictive model proposed by Kayange et al. [12]. The figure shows the measured and model's predicted heart rate over the same time period.

The DBN-LSTM model (Fig. 5.2) performs best in scenarios where the heart rate remains relatively stable, even if the input signals fluctuate significantly. In these cases, the model effectively ignores the noisy oscillations in the input and maintains consistent heart rate predictions. This suggests that the DBN-LSTM is able to extract underlying patterns that reflect the heart rate's inertia and resistance to rapid changes, rather than reacting directly to every input variation.

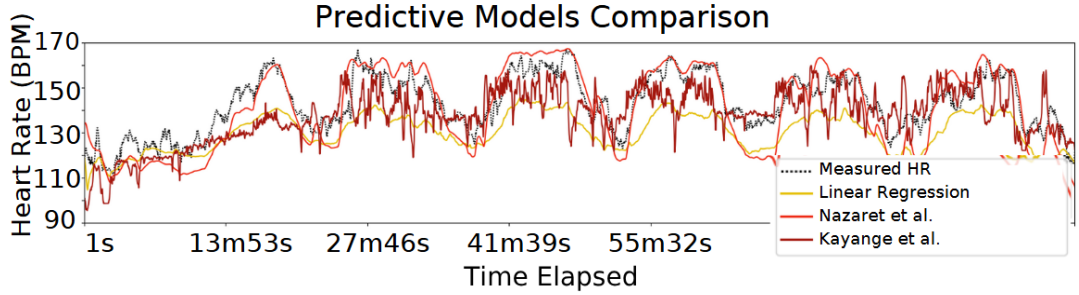

Figure 5.3: Comparative output from all heart rate prediction models, Linear Regression, Nazaret et al. [6], and Kayange et al. [12], on the same workout segment. The figure shows the measured and model's predicted heart rate over the same time period.

Figure 5.3 compares all different models (Linear Regression, ODE and DBN-LSTM). The baseline linear regression model shows it can capture some of the overall heart rate trend. It follows the general direction of the signal but lacks the flexibility to adapt to rapid changes or nonlinear dynamics. Still, it manages to roughly align with the trend, providing a basic approximation. In contrast, the ODE-based model tracks the heart rate dynamics much more accurately. It not only captures the general trend but also adapts to changes in heart rate levels over time. The model produces smooth predictions that align closely with the ground truth, reflecting its strength in modeling continuous physiological processes. The DBN-LSTM model, however, shows a different behavior. While it roughly follows the large-scale trends, the predictions exhibit significant oscillations around the true heart rate. These fluctuations are not present in the actual data and cause the model to diverge from the ground truth frequently. As a result, the excessive oscillations reduce the correlation score substantially, even though the model is occasionally in the correct range. This highlights a weakness making it less reliable for capturing the true underlying trend.

To evaluate which model performs best, we compare the mean absolute error (MAE) and correlation between predicted and true heart rate values across all workouts. MAE reflects how far off the predictions are on average and correlation indicates how well the model captures the trends and fluctuations in heart rate. We use the Wilcoxon Signed-Rank Test to assess whether the differences between models are statistically significant. This non-parametric test is appropriate for our paired, non-normally distributed error data. Results (Appendix D) show that the ODE Parameter Reduced model consistently outperforms the other models across both MAE and correlation, with extremely statistically significant differences ($p < 0.0001$). This confirms that the improvement is not due to chance but reflects a real performance advantage.

In addition to prediction accuracy, we also log both training time and inference time for each model. We did this at Wahoo Fitness's request, explicitly noting that these metrics are for their internal reference only and won't influence the research direction.While training time matters less because the model trains once or infrequently, inference time is crucial for our use case. Since the predicted heart rate feeds anomaly detection for many workouts daily, inefficient inference can bottleneck the process. Logging inference time allows us to assess the model's practicality for large-scale deployment and ensures that performance gains don't come at the cost of scalability. We trained models without GPU acceleration on the influ5 node, while GPU-accelerated jobs ran on gpu01; their specifications appear in Section 2.5.

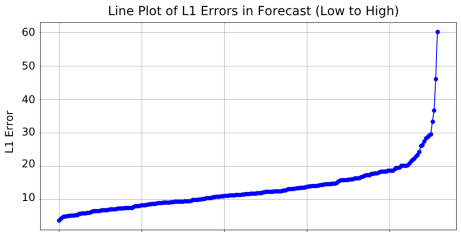

Figure 5.4: Ordered distribution of heart rate prediction errors across all test workouts. Each point represents a workout, sorted by increasing error magnitude.

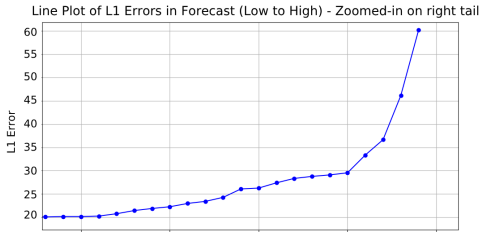

Figure 5.5: Zoomed-in view of the right tail of the ordered distribution of heart rate prediction errors across all test workouts.

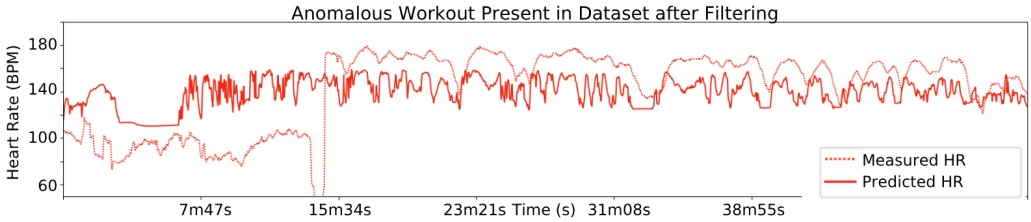

Figure 5.6: Example workout illustrating an unrealistic heart rate measurement segment. Combined with a more plausible predicted heart rate signal.

Besides the precaution of selecting high level data it can still occur that workout with measurement errors occur in the dataset. We should consider this when evaluating the results. Figure 5.4-5.5 show

sudden exponential growth in prediction errors. Figure 5.6 shows an example workout from the real dataset where the change in measured heart rate is unrealistically large. In those cases, large prediction errors are actually expected and even desirable, since the model should not try to fit faulty measurements. Using the median instead of the mean to track heart rate prediction performance makes sense here—it reduces the impact of outliers and better reflects typical model performance without skew from occasional bad data.

5.2.2. Anomaly Source Identification

We evaluate whether the heart rate prediction model can maintain high accuracy without using power as an input. If successful, this power-independent model enables us to isolate and attribute anomalies to the power or heart rate sensor specifically, providing insight into the source of data corruption.

Variables	MAE mean	MAE median	RMSE median	MAPE median	correlation median
Power Model	9.595	8.185 [5.979-11.441]	11.747 [7.728-13.907]	5.18% [3.68%-7.68%]	0.812 [0.697-0.886]
No Power Model	10.048	8.777 [6.807-11.672]	12.293 [8.628-14.206]	5.56% [4.18%-7.75%]	0.693 [0.536-0.803]

Table 5.3: Summary of prediction error ranges for the test set across prediction model with and without power. The table reports minimum, average, and maximum errors per metric over 10 test runs.

Our model predicts heart rate based on sensor input, with power identified as the most effective predictor of heart rate changes. However, we face a challenge when detecting an anomaly in the heart rate reconstruction: we cannot immediately determine whether the anomaly originates from power or heart rate data. To address this, we employ a secondary model, which predicts heart rate using gradient, cadence and speed (Table 5.3). Although this model is less accurate, it helps distinguish the source of the anomaly. If the secondary model also detects an anomaly, the issue lies with the heart rate data. If the secondary model detects no anomaly, we deem the power data anomalous.

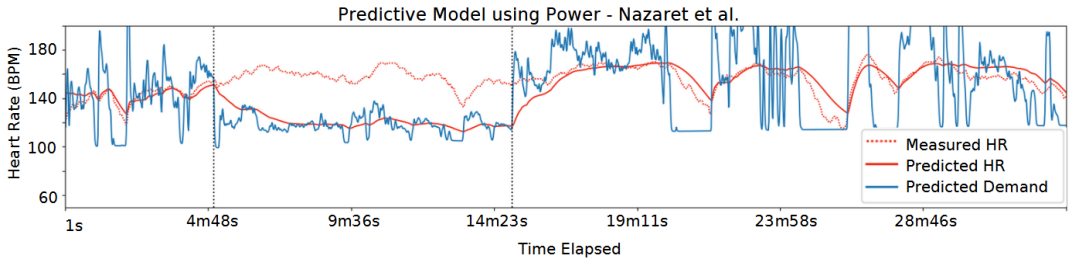

Figure 5.7: Output of the heart rate prediction model, from Nazaret et al. [6], that uses power as an input, with a simulated power anomaly (10% reduction) introduced between the dotted lines.

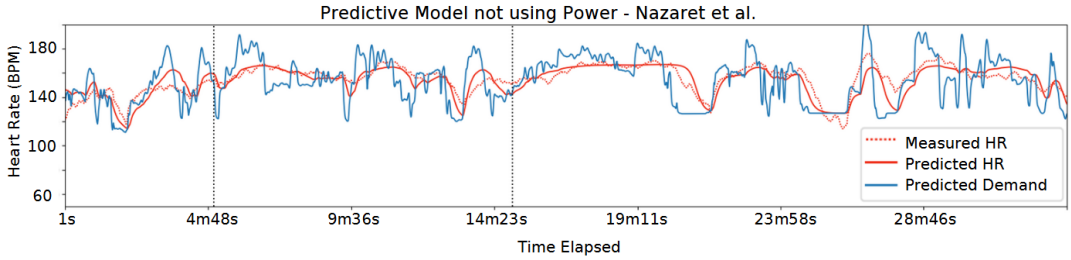

Figure 5.8: Output of the heart rate prediction model, from Nazaret et al. [6], that does not use power as an input, with a simulated power anomaly (10% reduction) introduced between the dotted lines.

From Fig. 5.7 we can see that the model using power detects anomalous heart rate, since the error between measured and predicted is higher than in the rest of the activity. While in Fig. 5.8 we can see that the model which does not use power does not detect anomalous heart rate data, as the error stays relatively consistent. From this we can derive that it is not the heart rate that is anomalous but the power input to the model.

5.3. Anomaly Detection

This section evaluates how well the trained models detect anomalies, injected following Section 4.3.1, as described in Section 4.4. We assess detection performance for point, subsequence, and time series anomalies to reflect realistic sensor error patterns.

For anomaly detection, we rely on several evaluation metrics to assess the performance of our model: precision, recall, and F1 score. For each metric we compute the average, minimum and maximum scores across all ten runs with random anomalies.

5.3.1. Point Anomaly Detection

Point Anomalies	Precision	Recall	F1-score
Half HR	0.84 [0.83-0.84]	1.00 [1.00-1.00]	0.91 [0.91-0.91]
HR matches cadence	0.83 [0.83-0.84]	0.99 [0.99-1.00]	0.91 [0.90-0.91]

Table 5.4: Average, minimal and maximal results of precision, recall and F1-score from 10 iterations of Point Anomaly Detection per anomaly type.

Table (5.4) shows the results of point anomaly detection on two types of artificially injected heart rate anomalies: Half HR and HR matches cadence. The model achieves perfect or near-perfect recall (1.00 and 0.99) in both cases, meaning it detects nearly all the injected anomalies. Precision is slightly lower (0.84 and 0.83), indicating that some of the flagged anomalies weren't the manually injected ones.

This pattern suggests that hard deviations like these, which occur on a very short time scale, are relatively easy for the model to detect, hence the perfect recall. However, since precision isn't 1.00, the model is also detecting additional points it considers anomalous, even though they weren't manually labeled. These could be real anomalies in the data, just not part of the injected ground truth. So, we likely lose some performance on precision due to false positives, though in practice these might still be valuable detections.

5.3.2. Subsequence Anomaly Detection

Subsequence Anomalies	Precision	Recall	F1-score
HR lags behind	0.59 [0.47-0.73]	0.25 [0.19-0.33]	0.35 [0.27-0.46]
HR matches cadence	0.57 [0.56-0.58]	0.86 [0.85-0.86]	0.68 [0.68-0.69]

Table 5.5: Average, minimal and maximal results of precision, recall and F1-score from 10 iterations of Subsequence Anomaly Detection per anomaly type.

Table 5.5 shows the results of subsequence anomaly detection. The two subsequence anomalies introduced in the dataset are subsequences where the heart rate lags behind the real heart rate and sequences where the heart rate matches the cadence. Subsequence anomalies prove significantly more difficult to detect than point anomalies. This is partly due to their more subtle nature, rather than a single sharp deviation, these anomalies often take the form of slow drifts or lagging responses that are harder to distinguish from natural physiological variability.

Anomalies where the heart rate lags behind after a sudden increase in power is challenging because such delayed responses can also occur during recovery, fatigue onset, or changes in conditions. The model struggles to confidently classify these patterns without falsely flagging valid but atypical physiological behavior. In contrast, anomalies matching cadence are easier to detect and the model reliably picks them up.

An additional challenge is the increased likelihood of false positives in the dataset. Some labeled normal segments may be actual anomalous segments due to subtle real errors slipping through the data cleaning phase. These mislabeled segments lead to artificially inflated false positive counts, reducing precision. This is especially relevant in our setup, where we deliberately prioritize precision over recall, we want to avoid flagging clean workouts as anomalous, even if it means missing some anomalies. The model still maintains high practical utility by preserving confidence in the anomalies it does detect, when accounting for this lowered precision due to working with a real-world dataset.

5.3.3. Time Series Anomaly Detection

Time Series Anomalies	Precision	Recall	F1-score
k=2.0	0.82 [0.81-0.82]	0.72 [0.68 – 0.76]	0.77 [0.74 – 0.79]
k=2.5	0.86 [0.86-0.87]	0.67 [0.64 – 0.70]	0.75 [0.73 – 0.77]
k=3.0	0.89 [0.89-0.89]	0.62 [0.60 – 0.67]	0.73 [0.72 – 0.76]
k=3.5	0.91 [0.90-0.91]	0.59 [0.56 – 0.63]	0.71 [0.69 – 0.75]
k=4.0	0.93 [0.93-0.94]	0.56 [0.52 – 0.60]	0.70 [0.67 – 0.73]
PR-AUC optimization	0.81 [0.81-0.82]	0.78 [0.76-0.81]	0.80 [0.78-0.81]

Table 5.6: Average, minimal and maximal results of precision, recall and F1-score from 10 iterations of Time Series Anomaly Detection with different thresholding techniques.

First we evaluate all metrics on the same set of diverse anomalies to see which thresholding technique works best as described in the methodology. This allows us to compare static thresholding (based on prediction error statistics) with dynamic, performance-optimized thresholding using PR-AUC maximization. For static thresholding, Model 1 yields a mean prediction error of 9.591 with a standard deviation of 5.665, while Model 2 shows a slightly higher mean of 10.048 and 5.361 standard deviation. In contrast, PR-AUC-based optimization selects significantly different thresholds: 19.925 ($= \mu + 1.824 \cdot \sigma$) for Model 1 and 32.630 ($= \mu + 4.212 \cdot \sigma$) for Model 2.

Table 5.6 shows that PR-AUC optimization consistently outperforms static thresholding. Its advantage lies in the added flexibility of tuning the threshold k independently per model. This allows Model 1 to apply a more lenient threshold, capturing broader anomaly patterns across a time series, while Model 2 uses a much stricter threshold, flagging only the most severe heart rate anomalies. Based on this improved performance, we select PR-AUC optimization as the preferred thresholding method and use it for evaluating each anomaly type separately in the next section.

Time Series Anomalies	Precision	Recall	F1-score
Half HR	0.85 [0.84-0.85]	1.00 [1.00-1.00]	0.92 [0.91-0.92]
HR matches cadence	0.84 [0.84-0.85]	1.00 [1.00-1.00]	0.91 [0.91-0.92]
Half Power	0.80 [0.78-0.82]	0.74 [0.64-0.81]	0.77 [0.70-0.82]
20% Power offset	0.72 [0.53-0.81]	0.51 [0.20-0.74]	0.59 [0.29-0.77]

Table 5.7: Average, minimal and maximal results of precision, recall and F1-score from 10 iterations of Time Series Anomaly Detection per anomaly type.

In Table 5.7 we can see the results of time series anomalies. We see that our model is very suitable in detecting most anomaly types. We find that sessions where heart rate is consistently too low or too high for the given effort are easier to flag, whereas miscalibrated power sensors (e.g., constant offset) are harder to catch, especially if the error remains within plausible human output ranges. This highlights the challenge of context ambiguity in time series anomaly detection.

5.4. Heart Rate Reconstruction

This section tests whether anomalous or corrupted heart rate values can be plausibly replaced by model predictions, as described in Section 4.5. The aim is to evaluate whether imputation using predicted heart rate improves signal quality.

We evaluate our model's reconstruction performance in two ways: using only anomalies detected by the model, and using the ground truth anomalies. The first is realistic in cases where we want to reconstruct anomalies identified by the model. The second is realistic for reconstructing data dropout, where no data is available. For both cases, we calculate the deviation of the reconstructed heart rate from the true measured heart rate. We document the average, minimum and maximum over the 10 test runs.

5.4.1. Point Anomaly Reconstruction

Anomaly Source	Method	Half HR MAE	HR Matches Cadence MAE
Model Anomalies	Before Reconstruction	84.138 [84.05–84.59]	54.285 [52.08–55.76]
	Baseline Interpolation	0.982 [0.97–1.00]	0.998 [0.97–1.03]
	Forecast	12.098 [11.87–12.45]	12.254 [11.94–12.57]
	Constant Error	1.067 [1.05–1.07]	1.082 [1.05–1.11]
Ground Truth Anomalies	Before Reconstruction	100.751 [99.98–101.31]	64.790 [62.58–66.28]
	Baseline Interpolation	0.305 [0.28–0.32]	0.302 [0.29–0.33]
	Forecast	9.454 [9.13–9.83]	9.652 [9.35–10.06]
	Constant Error	0.333 [0.31–0.35]	0.331 [0.32–0.36]

Table 5.8: Reconstruction MAE for Point Anomalies of Two Types: Half HR and HR Matches Cadence, across Model-Injected and Ground Truth Anomalies. Results show average, minimum, and maximum error across 10 test runs.

The results (Table 5.8) show that anomalies are reconstructed with very little error. As expected the baseline solution, interpolation, still outperforms the other methods for such time scale. Although we can see that our more sophisticated method of keeping constant errors already performs relatively close to this, highlighting its potential.

5.4.2. Subsequence Anomaly Reconstruction

Anomaly Source	Method	HR lags behind MAE	HR Matches Cadence MAE
Model Anomalies	Before Reconstruction	17.470 [12.65–23.35]	51.870 [51.10–52.71]
	Baseline Interpolation	18.631 [18.16–19.06]	46.915 [45.52–48.30]
	Forecast	19.245 [16.16–22.14]	11.324 [11.06–11.76]
	Constant Error	16.533 [15.65–17.22]	9.702 [8.37–11.14]
Ground Truth Anomalies	Before Reconstruction	17.423 [14.22–22.00]	65.073 [63.76–66.64]
	Baseline Interpolation	11.947 [10.80–13.67]	7.945 [7.76–8.20]
	Forecast	10.287 [9.86–10.63]	9.470 [9.24–10.01]
	Constant Error	8.048 [7.47–8.93]	5.889 [5.74–6.15]

Table 5.9: Reconstruction MAE for Subsequence Anomalies of Two Types: HR lags behind and HR Matches Cadence, across Model-Injected and Ground Truth Anomalies. Results show average, minimum, and maximum error across 10 test runs.

Table 5.9 highlights a clear contrast between controlled and real-world performance of the reconstruction model. When provided with ground truth anomaly segments, the constant-error reconstruction approach performs well, especially because it can rely on accurate context before and after the anomaly.

However, in the full pipeline, where anomalies are automatically detected, the reconstruction process also acts on false positives. These segments wrongly flagged as anomalous introduce unnecessary corrections, which inflate reconstruction error. This effect is particularly pronounced in the heart rate matches cadence anomaly type, where the models identified anomalies are heavily corrected due to the large discrepancy between heart rate and cadence. As a result, many normal sequences are mistakenly altered, leading to substantial extra error. The forecast reconstruction approach seems to handle this in a more robust manner. In contrast, heart rate lag anomalies exhibit smaller deviations, making them harder to detect. Consequently, the model produces a large number of false positives for this category. Because reconstruction also applies to false positives in model anomalies, the net effect is minimal improvement or even degradation. This explains why the reconstruction error for heart rate lag anomalies remains nearly the same before and after applying reconstruction, roughly half the reconstructed segments were not anomalous in the first place.

Overall, this analysis suggests that forecast-based imputation is a more reliable fallback than simple reconstruction when the detection step cannot guarantee high precision.

5.4.3. Time Series Anomaly Reconstruction

Anomaly Source	Method	Half HR MAE	HR Matches Cadence MAE
Model Anomalies	Before Reconstruction	65.251 [64.88-65.58]	63.827 [63.42–64.63]
	Baseline Average HR	17.974 [17.44–18.50]	17.956 [17.49–18.38]
	Forecast	10.289 [10.00–10.50]	10.412 [10.17–10.83]
Ground Truth Anomalies	Before Reconstruction	66.896 [66.42-67.24]	64.786 [64.39–65.48]
	Baseline Average HR	17.331 [16.73-17.78]	17.457 [16.97-17.91]
	Forecast	9.543 [9.30–9.72]	9.599 [9.32–10.07]

Table 5.10: Reconstruction MAE for Time Series Anomalies of Two Types: Half HR and HR Matches Cadence, across Model-Injected and Ground Truth Anomalies. Results show average, minimum, and maximum error across 10 test runs.

Reconstruction of time series anomalies is notably effective, as shown in Table 5.10. Unlike point or subsequence anomalies, we cannot apply the constant error approach here because no clean context exists within the session. Despite this limitation, the forecasting model significantly outperforms the baseline across all anomaly types, both when operating on ground truth labels and within the full anomaly detection pipeline.

Interestingly, even when the model detects anomalies less precisely, reconstruction performance remains strong. This suggests that, in the case of full-session corruption, the forecasting model is not only effective but also resilient to occasional misclassifications of sessions.

6

Discussion

6.1. Summary

This thesis demonstrates that prediction-based anomaly detection can be effectively applied to multivariate cycling data to identify and reconstruct heart rate (HR) and power anomalies. The results show a clear advantage of physiological model-based predictors over a naive baseline. Specifically, the parameter-reduced ODE model provided not only competitive predictive performance but also more reliable anomaly detection due to the absence of hard constraints like HR_{min} and HR_{max}.

Across all experiments, the parameter reduced prediction-based model by Nazaret et al. [6] demonstrated strong capability in modeling personalized heart rate dynamics. It consistently outperformed baseline and other approaches. Anomaly detection results confirm the hypothesis that point anomalies are easiest to detect, while subsequence anomalies, specifically the heart rate lag anomaly type, poses greater challenges. Reconstruction experiments show that replacing corrupted heart rate values with model predictions significantly reduces error, supporting the hypothesis that predicted values are more physiologically plausible than corrupted inputs. These findings validate the framework's ability to detect and correct implausible heart rate data.

6.2. Interpretations

By modeling heart rate as a function of effort metrics through a personalized ODE framework, we were able to identify anomalies not just based on statistical deviation but on physiological implausibility. The model correctly detects unrealistic heart rate behaviors, such as abrupt spikes or drops, that are not justified by surrounding activity metrics. This indicates that it has internalized core dynamics of cardiovascular response during exercise.

It is important to recognize that perfect anomaly detection and reconstruction are fundamentally unattainable in this setting. Our model predicts expected values based on contextual metrics and user embeddings, but unobserved factors such as fatigue, hydration, stress, or sensor drift introduce variability that no model can fully capture. Moreover, the very definition of what constitutes a "true" heart rate or power value is inherently noisy and dynamic. As a result, the model may flag some valid but unusual data points. The goal is not perfection, but to strike a practical balance: to reliably correct the most impactful and obvious errors while avoiding over-correction of plausible physiological variation. Our results demonstrate that the model meets this goal, improving data quality in a measurable way, while acknowledging that a certain level of residual error is inevitable in any real-world wearable data pipeline.

6.3. Implications

This work demonstrates that anomaly detection in physiological time-series data can go beyond black-box reconstruction models. For real-world applications, where flawed data can lead to incorrect training decisions, poor athlete feedback, or compromised recommender systems, detection models need to

understand the underlying physiological mechanisms. By framing anomaly detection as a prediction task and grounding the model in domain knowledge, we move closer to trustworthy data pipelines.

Moreover, the model's ability to not only detect but also correct erroneous heart rate readings has practical value. Replacing corrupted values with realistic predictions enables continued downstream analysis without throwing away entire sessions. This makes it suitable for real applications, where missing or unusable data can be costly. This type of model can increase user trust in wearables, instead of logging flawed data, systems can now flag, explain, and correct it.

6.4. Limitations

Despite extensive preprocessing, the dataset is not perfectly clean. Some real-world anomalies likely slipped through the filtering phase. Because evaluation relies only on injected anomalies, any model-detected anomaly not labeled as such counts as a false positive. This penalizes the model unfairly and artificially lowers the measured precision, even though it behaves correctly. While we must account for this in evaluation, the volume of such mislabeled cases should be small thanks to rigorous filtering. Although good performance on synthetically injected anomalies, the system's practical value depends on its ability to handle real anomalies. The use of synthetic anomaly injection could oversimplify the problem. Real-world sensor noise is not always as structured or isolated as simulated dropouts or mismatches with cadence.

Additionally, the model trains to infer a user's fitness from scratch, a challenging task with limited data or workouts in a narrow intensity range. However, this may not be strictly necessary. Many users have already completed Wahoo's fitness test, which provides an estimated fitness level. Incorporating these values into the model could serve as a prior or guiding signal for the fitness estimation process, potentially significantly improving the accuracy and stability of user-specific fitness learning for those users who have this data available.

6.5. Recommendations

While the primary purpose of the heart rate predictor in this project is to support anomaly detection, the model itself offers broader potential value to Wahoo Fitness. By accurately predicting an athlete's heart rate response from contextual variables, this system could enhance several features. For example, the predictor could improve real-time pacing guidance, alerting athletes when their physiological response deviates from expected effort, helping avoid overtraining or early burnout. Training workouts have an intended effort level, but it doesn't adjust in real-time. The model could monitor deviations from the intended intensity and adapt workouts in real-time to better fit the planned intensity. Additionally, the model could enhance adaptive training recommendations, where future sessions adjust automatically based on how the athlete's heart rate trends compared to predictions, capturing fatigue, fitness, or environmental effects. Overall, the heart rate predictor could serve as a foundation for Wahoo to deliver smarter, physiology-driven insights.

Notably, the original heart rate prediction paper that inspired this work focused on running exercises rather than cycling. Interestingly, the model achieved even lower prediction errors in that context than what we observed for cycling, suggesting that its physiological modeling may generalize well in running workouts. As Wahoo Fitness continues to expand its presence in the running market this predictive capability could provide direct value for running-focused applications as well.

An important aspect of the heart rate predictor is that the model learns a user-specific fitness representation. This representation captures how an individual's heart rate responds to changes in power and other variables, modeling the athlete's physiological profile. Beyond anomaly detection, Wahoo Fitness could repurpose this learned fitness embedding for other applications on their platform. For example, by adding a new decoder, the model could predict an athlete's 4DP (Four-Dimensional Power) profile, a Wahoo test estimating performance across Neuromuscular Power, Anaerobic Capacity, Maximal Aerobic Power, and Functional Threshold Power (FTP). These metrics give a more complete picture of a cyclist's strengths and weaknesses. Integrating the fitness representation with 4DP estimation could reduce the need for frequent testing and enable continuous updates of a rider's profile based on regular training data. This would open the door for smarter, personalized training recommendations and performance tracking.

7

Conclusion

This thesis set out to address the increasingly pressing issue of erroneous sensor data in wearable fitness technology, particularly within the cycling domain. Leveraging the rich but often noisy dataset provided by Wahoo Fitness, we proposed a novel prediction-based framework for anomaly detection that moves beyond traditional statistical outlier detection. Instead, we rooted anomaly detection in physiological plausibility by modeling individual heart rate (HR) responses to exercise metrics like power, cadence, gradient, altitude and speed.

A central contribution of this work was the use of user-specific embeddings and a hybrid ODE-based model proposed by Nazaret et al. [6], enabling the system to predict what a plausible heart rate response should look like for each individual athlete. This design allowed the model not only to detect implausible heart rate values but also to distinguish between anomalies originating from heart rate sensors and those stemming from corrupted input signals like power. Through dual-model architecture and carefully tuned prediction error thresholds, we effectively isolated error sources, addressing a critical real-world challenge that existing literature had largely neglected.

Our parameter-reduced version of Nazaret et al. [6] model consistently outperformed both the linear regression baseline and the DBN-LSTM model, confirming that simpler yet physiologically grounded models can excel when appropriately personalized. The experimental results validate the hypothesis. Point anomalies, as expected, were the easiest to detect and reconstruct, benefiting from short temporal disruptions and strong local correlation. Full-session anomalies posed greater challenges but were still handled effectively through adjusted thresholding techniques. Subsequence anomalies remained the hardest to reconstruct, specifically the heart rate lag.

While our approach improves detection and reconstruction accuracy, it is not without limitations. The reliance on injected synthetic anomalies, while necessary to establish ground truth, limits the realism of the evaluation. Moreover, real-world anomalies that slipped through preprocessing likely penalized the model unfairly by inflating false positives. The lack of full contextual data availability, like sleep, stress and hydration, also prevent us from getting near perfect results.

Nevertheless, the implications of this work are far-reaching. Beyond the immediate benefit of improving Wahoo Fitness data quality, our method lays the groundwork for smarter wearable analytics. By embedding an athlete's physiological signature into the model, we move closer to continuous, individualized monitoring systems capable of detecting early signs of overtraining, illness, or equipment malfunction. Moreover, the reconstructed heart rate signals can serve as high-quality input for downstream tasks such as training load estimation, adaptive workout generation, and long-term performance tracking—functions that hinge on accurate physiological data.

Looking forward, several avenues for future research emerge. One is to expand the model input space to include environmental variables or subjective user input, thereby increasing robustness against unmeasured context. Additionally, real-time implementation could transform this approach from a retrospective analysis tool into an active feedback mechanism. For a detailed reflection on the methodological and practical lessons encountered during this project, see Appendix E: Lessons Learned.

In summary, this thesis has shown that by rethinking anomaly detection as a problem of physiological plausibility, grounded in individualized models of heart rate dynamics, we can not only identify data corruption more accurately but also correct it with meaningful replacements. This prediction-based approach represents a significant step toward restoring trust in wearable fitness data and unlocking its full potential for both athletes and the systems that support them.

References

[1] Walter R. Thompson. “WORLDWIDE SURVEY OF FITNESS TRENDS FOR 2017”. en-US. In: *ACSM’s Health & Fitness Journal* 20.6 (Dec. 2016), p. 8. ISSN: 1091-5397. DOI: 10.1249/FIT.0000000000000252. URL: https://journals.lww.com/acsm-healthfitness/fulltext/2016/11000/WORLDWIDE_SURVEY_OF_FITNESS_TRENDS_FOR_2017.6.aspx?casa_token=gMDe_XmomUkAAAAA:JdvUiD59uFjTikCEOtOXShdzuMTfn57ncr8-JhvC7LFbd31cuGaRG3YQeWZi8rTzrtdKVd6dh8CtBFkKLkvOUcdTkWI (visited on 12/18/2024).

[2] Walter R. Thompson. “WORLDWIDE SURVEY OF FITNESS TRENDS FOR 2019”. en-US. In: *ACSM’s Health & Fitness Journal* 22.6 (Dec. 2018), p. 10. ISSN: 1091-5397. DOI: 10.1249/FIT.0000000000000438. URL: https://journals.lww.com/acsm-healthfitness/fulltext/2018/11000/WORLDWIDE_SURVEY_OF_FITNESS_TRENDS_FOR_2019.6.aspx?amp_device_id=RgTfKwhtEKFtwhzYNOsjzM&casa_token=i5e-BVnPUj0AAAAA:VplolMIC-N2tws6rGK-Vn8NBLoXM0lxotDNVxd5PZxfYH-Et8KogoTCHf1qroWvtJoaYLHLXRBVsr5SggqAv6EVSQqo (visited on 12/18/2024).

[3] A’Naja M. Newsome et al. “2024 ACSM Worldwide Fitness Trends: Future Directions of the Health and Fitness Industry”. en-US. In: *ACSM’s Health & Fitness Journal* 28.1 (Feb. 2024), p. 14. ISSN: 1091-5397. DOI: 10.1249/FIT.0000000000000933. URL: https://journals.lww.com/acsm-healthfitness/fulltext/2024/01000/2024_acsm_worldwide_fitness_trends__future.7.aspx?casa_token=reDPu-sbPH4AAAAA:dUgf_OM5loDBbMfCCLROkbejk5kFY94qx6DJcTmdRci70cqsqwekrFRMhk5VSet7agX95ivzcTQe83AQqT9SNp4s7OI (visited on 12/18/2024).

[4] Cailbhe Doherty et al. “Keeping Pace with Wearables: A Living Umbrella Review of Systematic Reviews Evaluating the Accuracy of Consumer Wearable Technologies in Health Measurement”. en. In: *Sports Medicine* 54.11 (Nov. 2024), pp. 2907–2926. ISSN: 1179-2035. DOI: 10.1007/s40279-024-02077-2. URL: https://doi.org/10.1007/s40279-024-02077-2 (visited on 12/18/2024).

[5] Aleksei Karetnikov, Wim Nuijten, and Marwan Hassani. “Data-driven Support of Coaches in Professional Cycling using Race Performance Prediction: 9th International Conference on Sport Sciences Research and Technology Support, icSPORTS 2021”. In: *icSPORTS 2021 - Proceedings of the 9th International Conference on Sport Sciences Research and Technology Support* (2021). Ed. by Pedro Pezarat-Correia, Joao Vilas-Boas, and Jan Cabri. Publisher: SciTePress Digital Library, pp. 43–53. URL: http://www.scopus.com/inward/record.url?scp=85146197945&partnerID=8YFLogxK (visited on 12/18/2024).

[6] Achille Nazaret et al. “Modeling personalized heart rate response to exercise and environmental factors with wearables data”. en. In: *npj Digital Medicine* 6.1 (Nov. 2023). Publisher: Nature Publishing Group, pp. 1–7. ISSN: 2398-6352. DOI: 10.1038/s41746-023-00926-4. URL: https://www.nature.com/articles/s41746-023-00926-4 (visited on 12/05/2024).

[7] Christopher M. Bishop and Hugh Bishop. *Deep Learning: Foundations and Concepts*. en. Cham: Springer International Publishing, 2024. ISBN: 978-3-031-45467-7 978-3-031-45468-4. DOI: 10.1007/978-3-031-45468-4. URL: https://link.springer.com/10.1007/978-3-031-45468-4 (visited on 06/09/2025).

[8] Melanie Ludwig et al. “Measurement, Prediction, and Control of Individual Heart Rate Responses to Exercise—Basics and Options for Wearable Devices”. English. In: *Frontiers in Physiology* 9 (June 2018). Publisher: Frontiers. ISSN: 1664-042X. DOI: 10.3389/fphys.2018.00778. URL: https://www.frontiersin.org/journals/physiology/articles/10.3389/fphys.2018.00778/full (visited on 02/24/2025).

[9] Oleksii Bychkov et al. “Medical Card Information System for Data Analysis from Fitness Bracelets”. en. In: ().

[10] Ane Blázquez-García et al. “A Review on Outlier/Anomaly Detection in Time Series Data”. In: *ACM Comput. Surv.* 54.3 (Apr. 2021), 56:1–56:33. ISSN: 0360-0300. DOI: 10.1145/3444690. URL: https://dl.acm.org/doi/10.1145/3444690 (visited on 11/11/2024).

[11] Jianmo Ni, Larry Muhlstein, and Julian McAuley. “Modeling Heart Rate and Activity Data for Personalized Fitness Recommendation”. In: *The World Wide Web Conference*. WWW ’19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 1343–1353. ISBN: 978-1-4503-6674-8. DOI: 10.1145/3308558.3313643. URL: https://dl.acm.org/doi/10.1145/3308558.3313643 (visited on 02/24/2025).

[12] Hyston Kayange et al. “A Hybrid Approach to Modeling Heart Rate Response for Personalized Fitness Recommendations Using Wearable Data”. In: *Electronics* 13 (Sept. 2024), p. 3888. DOI: 10.3390/electronics13193888.

[13] Yujie Fan. “Wearable-Based Personalized Exercise Heart Rate Estimation and Distribution Analysis Using Dual-Context LSTM Model”. en. In: *Internet Technology Letters* n/a.n/a (). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/itl2.627, e627. ISSN: 2476-1508. DOI: 10.1002/itl2.627. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/itl2.627 (visited on 02/24/2025).

[14] Asieh Namazi. “On the improvement of heart rate prediction using the combination of singular spectrum analysis and copula-based analysis approach”. en. In: *PeerJ* 10 (Dec. 2022). Publisher: PeerJ Inc., e14601. ISSN: 2167-8359. DOI: 10.7717/peerj.14601. URL: https://peerj.com/articles/14601 (visited on 02/24/2025).

[15] Xiaoxing Qiu, Jules White, and Douglas Schmidt. “A Study of Machine Learning Models for Personalized Heart Rate Forecasting in Mountain Biking:” en. In: *Proceedings of the 9th International Conference on Sport Sciences Research and Technology Support*. Online Streaming, — Select a Country —: SCITEPRESS - Science and Technology Publications, 2021, pp. 87–94. ISBN: 978-989-758-539-5. DOI: 10.5220/0010630600003059. URL: https://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0010630600003059 (visited on 03/06/2025).

[16] Matthias Weippert et al. “Comparison of three mobile devices for measuring R–R intervals and heart rate variability: Polar S810i, Suunto t6 and an ambulatory ECG system”. en. In: *European Journal of Applied Physiology* 109.4 (July 2010), pp. 779–786. ISSN: 1439-6327. DOI: 10.1007/s00421-010-1415-9. URL: https://doi.org/10.1007/s00421-010-1415-9 (visited on 02/25/2025).

[17] N. Selvaraj et al. “Assessment of heart rate variability derived from finger-tip photoplethysmography as compared to electrocardiography”. In: *Journal of Medical Engineering & Technology* 32.6 (Jan. 2008). Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/03091900701781317, pp. 479–484. ISSN: 0309-1902. DOI: 10.1080/03091900701781317. URL: https://doi.org/10.1080/03091900701781317 (visited on 02/25/2025).

[18] Axel Schäfer and Jan Vagedes. “How accurate is pulse rate variability as an estimate of heart rate variability?: A review on studies comparing photoplethysmographic technology with an electrocardiogram”. In: *International Journal of Cardiology* 166.1 (June 2013), pp. 15–29. ISSN: 0167-5273. DOI: 10.1016/j.ijcard.2012.03.119. URL: https://www.sciencedirect.com/science/article/pii/S0167527312003269 (visited on 02/25/2025).

[19] Delft AI Cluster (DAIC). *The Delft AI Cluster (DAIC), RRID: SCR_025091*. https://doi.org/10.4233/rrid:scr_025091. 2024.

[20] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. “Anomaly detection in time series: a comprehensive evaluation”. en. In: *Proceedings of the VLDB Endowment* 15.9 (May 2022), pp. 1779–1797. ISSN: 2150-8097. DOI: 10.14778/3538598.3538602. URL: https://dl.acm.org/doi/10.14778/3538598.3538602 (visited on 11/14/2024).

[21] Senthilmurugan Muthukrishnan, Rahul Shah, and Jeffrey Vitter. “Mining deviants in time series data streams”. In: July 2004, pp. 41–50. ISBN: 978-0-7695-2146-6. DOI: 10.1109/SSDM.2004.1311192.

[22] Yuxun Zhou et al. “Non-Parametric Outliers Detection in Multiple Time Series A Case Study: Power Grid Data Analysis”. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 32 (Apr. 2018). DOI: 10.1609/aaai.v32i1.11632.

[23] Xiaolei Li et al. “Temporal Outlier Detection in Vehicle Traffic Data”. en. In: ().

[24] Haibin Cheng et al. “Detection and Characterization of Anomalies in Multivariate Time Series”. In: Apr. 2009, pp. 413–424. ISBN: 978-0-89871-682-5. DOI: 10.1137/1.9781611972795.36.

[25] Michael Jones et al. “Anomaly Detection in Real-Valued Multidimensional Time Series”. In: June 2014. URL: https://www.semanticscholar.org/paper/Anomaly-Detection-in-Real-Valued-Multidimensional-Jones-Nikovski/799cb22802bf4beb9fcbc06e707b57065b8973a5 (visited on 11/14/2024).

[26] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. “Isolation Forest”. In: *2008 Eighth IEEE International Conference on Data Mining*. ISSN: 2374-8486. Dec. 2008, pp. 413–422. DOI: 10.1109/ICDM.2008.17. URL: https://ieeexplore.ieee.org/document/4781136 (visited on 11/15/2024).

[27] A. Dairi et al. “Deep learning approach for sustainable WWTP operation: A case study on data-driven influent conditions monitoring”. English. In: *Sustainable Cities and Society* 50 (2019). Publisher: Elsevier Ltd. ISSN: 22106707 (ISSN). DOI: 10.1016/j.scs.2019.101670. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85068923171&doi=10.1016%2fj.scs.2019.101670&partnerID=40&md5=35f7e670041267053e36d78b3f2902e1.

[28] G.K. Vishwakarma, C. Paul, and A.M. Elsawah. “A hybrid feedforward neural network algorithm for detecting outliers in non-stationary multivariate time series”. English. In: *Expert Systems with Applications* 184 (2021). Publisher: Elsevier Ltd. ISSN: 09574174 (ISSN). DOI: 10.1016/j.eswa.2021.115545. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85109934463&doi=10.1016%2fj.eswa.2021.115545&partnerID=40&md5=e0ddc78ce56acc61dcebb9644c5ac0d4.

[29] R. Oucheikh et al. “Deep Anomaly Detector Based on Spatio-Temporal Clustering for Connected Autonomous Vehicles”. English. In: *Lect. Notes Inst. Comput. Sci. Soc. Informatics Telecommun. Eng.* Ed. by Foschini L. and El Kamili M. Vol. 345. Journal Abbreviation: Lect. Notes Inst. Comput. Sci. Soc. Informatics Telecommun. Eng. Springer Science and Business Media Deutschland GmbH, 2021, pp. 201–212. ISBN: 18678211 (ISSN); 978-303067368-0 (ISBN). DOI: 10.1007/978-3-030-67369-7_15. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85101399478&doi=10.1007%2f978-3-030-67369-7_15&partnerID=40&md5=2cde532a731f25a558fcde00c0ae833b.

[30] C. He, D.S. Leslie, and J.A. Grant. “Online Detection and Fuzzy Clustering of Anomalies in Non-Stationary Time Series †”. English. In: *Signals* 5.1 (2024). Publisher: Multidisciplinary Digital Publishing Institute (MDPI), pp. 40–59. ISSN: 26246120 (ISSN). DOI: 10.3390/signals5010003. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85188888548&doi=10.3390%2fsignals5010003&partnerID=40&md5=c75eceac367ed913e93d0d5176cc3863.

[31] M.C. Altindal et al. “Anomaly detection in multivariate time series of drilling data”. English. In: *Geoenergy Science and Engineering* 237 (2024). Publisher: Elsevier B.V. ISSN: 29498910 (ISSN). DOI: 10.1016/j.geoen.2024.212778. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85189752839&doi=10.1016%2fj.geoen.2024.212778&partnerID=40&md5=c4868a88f95633d837e6fcf0841e0173.

[32] I. Giurgiu and A. Schumann. “Additive explanations for anomalies detected from multivariate temporal data”. English. In: *Int Conf Inf Knowledge Manage*. Journal Abbreviation: Int Conf Inf Knowledge Manage. Association for Computing Machinery, 2019, pp. 2245–2248. ISBN: 978-145036976-3 (ISBN). DOI: 10.1145/3357384.3358121. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85075461204&doi=10.1145%2f3357384.3358121&partnerID=40&md5=a9ac296a76e411bab56123406059d624.

[33] F. Hosseinpour et al. “An Unsupervised Method for Anomaly Detection in Multi-Stage Production Systems Based on LSTM Autoencoders”. English. In: *Proc. Eur. Saf. Reliab. Conf. - Underst. Manag. Risk Reliab. Sustain. Future*. Ed. by Leva M.C. et al. Journal Abbreviation: Proc. Eur. Saf. Reliab. Conf. - Underst. Manag. Risk Reliab. Sustain. Future. Research Publishing, 2022, pp. 1346–1352. ISBN: 978-981185183-4 (ISBN). DOI: 10.3850/978-981-18-5183-4_R22-19-604-cd. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85178345415&doi=10.3850%2f978-981-18-5183-4_R22-19-604-cd&partnerID=40&md5=5b44126c04aaac6a50ecc355e965d2f4.

[34] K. Zhang et al. “Federated Variational Learning for Anomaly Detection in Multivariate Time Series”. English. In: *Conf. Proc. IEEE Int. Perform. Comput. Commun. Conf.* Vol. 2021-October. Journal Abbreviation: Conf. Proc. IEEE Int. Perform. Comput. Commun. Conf. Institute of Electrical and Electronics Engineers Inc., 2021. ISBN: 10972641 (ISSN); 978-166544331-9 (ISBN). DOI: 10.1109/IPCCC51483.2021.9679367. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85125197897&doi=10.1109%2fIPCCC51483.2021.9679367&partnerID=40&md5=eb293e4b337817db6d82f2e1d4a1c9f3.

[35] Y. Zhang et al. “Unsupervised Deep Anomaly Detection for Multi-Sensor Time-Series Signals”. English. In: *IEEE Transactions on Knowledge and Data Engineering* 35.2 (2023). Publisher: IEEE Computer Society, pp. 2118–2132. ISSN: 10414347 (ISSN). DOI: 10.1109/TKDE.2021.3102110. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85112672674&doi=10.1109%2fTKDE.2021.3102110&partnerID=40&md5=c45ed35fd5535a8517757ce1c027dc86.

[36] L. Liang et al. “Stationary Multi-scale Hierarchical Dilated Graph Convolution for Multivariate Time Series Anomaly Detection”. English. In: *Commun. Comput. Info. Sci.* Ed. by Tian Y., Ma T., and Khan M.K. Vol. 2100 CCIS. Journal Abbreviation: Commun. Comput. Info. Sci. Springer Science and Business Media Deutschland GmbH, 2024, pp. 52–66. ISBN: 18650929 (ISSN); 978-981974389-6 (ISBN). DOI: 10.1007/978-981-97-4390-2_5. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85200480818&doi=10.1007%2f978-981-97-4390-2_5&partnerID=40&md5=846212edc8afde3b5e099bfe8666cad9.

[37] L. Dai et al. “Switching Gaussian Mixture Variational RNN for Anomaly Detection of Diverse CDN Websites”. English. In: *Proc IEEE INFOCOM*. Vol. 2022-May. Journal Abbreviation: Proc IEEE INFOCOM. Institute of Electrical and Electronics Engineers Inc., 2022, pp. 300–309. ISBN: 0743166X (ISSN); 978-166545822-1 (ISBN). DOI: 10.1109/INFOCOM48880.2022.9796836. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85133218312&doi=10.1109%2fINFOCOM48880.2022.9796836&partnerID=40&md5=e2743ae8e095dec8a9e35c0ed20614bd.

[38] E.S. Miele, F. Bonacina, and A. Corsini. “Deep anomaly detection in horizontal axis wind turbines using Graph Convolutional Autoencoders for Multivariate Time series”. English. In: *Energy and AI* 8 (2022). Publisher: Elsevier B.V. ISSN: 26665468 (ISSN). DOI: 10.1016/j.egyai.2022.100145. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85124645066&doi=10.1016%2fj.egyai.2022.100145&partnerID=40&md5=02e6ea9612324b5ad09463596d12a651.

[39] J. Long, C. Luo, and R. Chen. “Multivariate Time Series Anomaly Detection with Improved Encoder-Decoder Based Model”. English. In: *Proc. - IEEE Int. Conf. Cyber Secur. Cloud Comput. IEEE Int. Conf. Edge Comput. Scalable Cloud, CSCloud-EdgeCom*. Journal Abbreviation: Proc. - IEEE Int. Conf. Cyber Secur. Cloud Comput. IEEE Int. Conf. Edge Comput. Scalable Cloud, CSCloud-EdgeCom. Institute of Electrical and Electronics Engineers Inc., 2023, pp. 161–166. ISBN: 979-835031246-1 (ISBN). DOI: 10.1109/CSCloud-EdgeCom58631.2023.00036. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85168239585&doi=10.1109%2fCSCloud-EdgeCom58631.2023.00036&partnerID=40&md5=567a383953c0dbb078f835d1a9344697.

[40] Y. Shi et al. “Robust anomaly detection for multivariate time series through temporal GCNs and attention-based VAE”. English. In: *Knowledge-Based Systems* 275 (2023). Publisher: Elsevier B.V. ISSN: 09507051 (ISSN). DOI: 10.1016/j.knosys.2023.110725. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85163554843&doi=10.1016%2fj.knosys.2023.110725&partnerID=40&md5=1d0ed301a5876b7b5a8a24b839284442.

[41] Y. Zheng et al. “Correlation-Aware Spatial-Temporal Graph Learning for Multivariate Time-Series Anomaly Detection”. English. In: *IEEE Transactions on Neural Networks and Learning Systems* 35.9 (2024). Publisher: Institute of Electrical and Electronics Engineers Inc., pp. 11802–11816. ISSN: 2162237X (ISSN). DOI: 10.1109/TNNLS.2023.3325667. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85177076018&doi=10.1109%2fTNNLS.2023.3325667&partnerID=40&md5=8f0d9d6a7e9cbd155e7d33b89819b592.

[42] Q. Yang et al. “Graph Transformer Network Incorporating Sparse Representation for Multivariate Time Series Anomaly Detection”. English. In: *Electronics (Switzerland)* 13.11 (2024). Publisher: Multidisciplinary Digital Publishing Institute (MDPI). ISSN: 20799292 (ISSN). DOI: 10.3390/electronics13112032. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-

85195850252&doi=10.3390%2felectronics13112032&partnerID=40&md5=5da9e239ae17bc9d4333d3ab31a3c7af.

[43] G. Ding, Y. Zhu, and Y. Ren. “Dynamic-Static Fusion for Spatial-Temporal Anomaly Detection and Interpretation in Multivariate Time Series”. English. In: *Lect. Notes Comput. Sci.* Ed. by Zhang W. et al. Vol. 14963 LNCS. Journal Abbreviation: Lect. Notes Comput. Sci. Springer Science and Business Media Deutschland GmbH, 2024, pp. 46–61. ISBN: 03029743 (ISSN); 978-981977237-7 (ISBN). DOI: 10.1007/978-981-97-7238-4_4. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85203141198&doi=10.1007%2f978-981-97-7238-4_4&partnerID=40&md5=94c6a143392fb7bed488eb3f1b135db2.

[44] L. Weiwei et al. “An Anomaly Detection Method Based on GCN and Correlation of High Dimensional Sensor Data in Power Grid System”. English. In: *Lect. Notes Inst. Comput. Sci. Soc. Informatics Telecommun. Eng.* Ed. by Wang X. et al. Vol. 396 LNICST. Journal Abbreviation: Lect. Notes Inst. Comput. Sci. Soc. Informatics Telecommun. Eng. Springer Science and Business Media Deutschland GmbH, 2021, pp. 444–454. ISBN: 18678211 (ISSN); 978-303090195-0 (ISBN). DOI: 10.1007/978-3-030-90196-7_38. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85119865678&doi=10.1007%2f978-3-030-90196-7_38&partnerID=40&md5=96c809114c05abcf3c5981996a99d366.

[45] C. Feng, C. Liu, and D. Jiang. “Unsupervised anomaly detection using graph neural networks integrated with physical-statistical feature fusion and local-global learning”. English. In: *Renewable Energy* 206 (2023). Publisher: Elsevier Ltd, pp. 309–323. ISSN: 09601481 (ISSN). DOI: 10.1016/j.renene.2023.02.053. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85149133075&doi=10.1016%2fj.renene.2023.02.053&partnerID=40&md5=6b23488f72985e6c88f96f94506d5700.

[46] M. Zhao and O. Fink. “DyEdgeGAT: Dynamic Edge via Graph Attention for Early Fault Detection in IIoT Systems”. English. In: *IEEE Internet of Things Journal* 11.13 (2024). Publisher: Institute of Electrical and Electronics Engineers Inc., pp. 22950–22965. ISSN: 23274662 (ISSN). DOI: 10.1109/JIOT.2024.3381002. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85189141870&doi=10.1109%2fJIOT.2024.3381002&partnerID=40&md5=e4b5e3b060fd461ad3915d869ed65fa7.

[47] P. Amil, N. Almeira, and C. Masoller. “Outlier Mining Methods Based on Graph Structure Analysis”. English. In: *Frontiers in Physics* 7 (2019). Publisher: Frontiers Media SA. ISSN: 2296424X (ISSN). DOI: 10.3389/fphy.2019.00194. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85076687478&doi=10.3389%2ffphy.2019.00194&partnerID=40&md5=2d469da0e7241bf5bb4c59dc0b2bc932.

[48] Yeseul Shim. *[ICLR 2022] Part 2: Time Series Anomaly Detection - LG AI Research BLOG*. URL: https://www.lgresearch.ai/blog/view?seq=231 (visited on 06/10/2025).

[49] J. He, Z. Dong, and Y. Huang. “Multivariate Time Series Anomaly Detection with Adaptive Transformer-CNN Architecture Fusing Adversarial Training”. English. In: *Proc. IEEE Data Driven Control Learn. Syst. Conf., DDCLS*. Journal Abbreviation: Proc. IEEE Data Driven Control Learn. Syst. Conf., DDCLS. Institute of Electrical and Electronics Engineers Inc., 2024, pp. 1387–1392. ISBN: 979-835036167-4 (ISBN). DOI: 10.1109/DDCLS61622.2024.10606841. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85202436449&doi=10.1109%2fDDCLS61622.2024.10606841&partnerID=40&md5=8bda4c2e36fefdce5c41bcbdefabc4d3.

[50] X. Yang et al. “Variable-wise generative adversarial transformer in multivariate time series anomaly detection”. English. In: *Applied Intelligence* 53.23 (2023). Publisher: Springer, pp. 28745–28767. ISSN: 0924669X (ISSN). DOI: 10.1007/s10489-023-05029-x. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85174021839&doi=10.1007%2fs10489-023-05029-x&partnerID=40&md5=0c66d9d74a5db080a4cb7e52f56a74c7.

[51] H. Zhang et al. “Unsupervised Anomaly Detection in Multivariate Time Series through Transformer-based Variational Autoencoder”. English. In: *Proc. Chin. Control Decis. Conf., CCDC*. Journal Abbreviation: Proc. Chin. Control Decis. Conf., CCDC. Institute of Electrical and Electronics Engineers Inc., 2021, pp. 281–286. ISBN: 978-166544089-9 (ISBN). DOI: 10.1109/CCDC52312.2021.9601669. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85125172347&

doi=10.1109%2fCCDC52312.2021.9601669&partnerID=40&md5=5445499096e388a4942ef1b1aeecab61.

[52] H. Choi, S. Kim, and P. Kang. “Recurrent auto-encoder with multi-resolution ensemble and predictive coding for multivariate time-series anomaly detection”. English. In: *Applied Intelligence* 53.21 (2023). Publisher: Springer, pp. 25330–25342. ISSN: 0924669X (ISSN). DOI: 10.1007/s10489-023-04764-5. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85167340073&doi=10.1007%2fs10489-023-04764-5&partnerID=40&md5=c5a1cdf3f24517074ef9c619245727d9.

[53] Y. Feng et al. “SensitiveHUE: Multivariate Time Series Anomaly Detection by Enhancing the Sensitivity to Normal Patterns”. English. In: *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.* Journal Abbreviation: Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. Association for Computing Machinery, 2024, pp. 782–793. ISBN: 2154817X (ISSN); 979-840070490-1 (ISBN). DOI: 10.1145/3637528.3671919. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85203707908&doi=10.1145%2f3637528.3671919&partnerID=40&md5=2c467ade0969fdef3bbcd725ee3f1b33.

[54] Mohammad Reza Chopannavaz and Foad Ghaderi. *An Empirical Investigation of Reconstruction-Based Models for Seizure Prediction from ECG Signals*. arXiv:2504.08381 [eess]. Apr. 2025. DOI: 10.48550/arXiv.2504.08381. URL: http://arxiv.org/abs/2504.08381 (visited on 04/19/2025).

[55] Dor Bank, Noam Koenigstein, and Raja Giryes. *Autoencoders*. arXiv:2003.05991 [cs]. Apr. 2021. DOI: 10.48550/arXiv.2003.05991. URL: http://arxiv.org/abs/2003.05991 (visited on 04/19/2025).

[56] Michael Mazzoleni et al. “Modeling and predicting heart rate dynamics across a broad range of transient exercise intensities during cycling”. In: *Sports Engineering* 19 (Jan. 2016), pp. 117–127. DOI: 10.1007/s12283-015-0193-3.

[57] Michele Paradiso et al. “Experimental Heart Rate Regulation in Cycle-Ergometer Exercises”. In: *IEEE transactions on bio-medical engineering* 60 (Oct. 2012). DOI: 10.1109/TBME.2012.2225061.

[58] Alexander Artiga Gonzalez, Raphael Bertschinger, and Dietmar Saupe. “Modeling V?O2 and V?CO2 with Hammerstein-Wiener Models:” en. In: *Proceedings of the 4th International Congress on Sport Sciences Research and Technology Support*. Porto, Portugal: SCITEPRESS - Science and Technology Publications, 2016, pp. 134–140. ISBN: 978-989-758-205-9. DOI: 10.5220/0006086501340140. URL: http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0006086501340140 (visited on 02/27/2025).

[59] Google. *How does my Fitbit device calculate calories burned?* Accessed: 2025-02-28. n.d. URL: https://support.google.com/fitbit/answer/14237111?hl=en.

[60] Nuno Domingos Garrido et al. “Precision of wearable heart rate to predict oxygen uptake in endurance vs. sprint-trained runners”. en. In: *Motricidade* 18.2 (June 2022). Number: 2, pp. 215–218. ISSN: 2182-2972. DOI: 10.6063/motricidade.27172. URL: https://revistas.rcaap.pt/motricidade/article/view/27172 (visited on 04/28/2025).

[61] L. Fang et al. “Bayesian inference federated learning for heart rate prediction”. eng. In: Accepted: 2021-05-10T14:30:14Z ISSN: 1867-8211. Springer, 2021. ISBN: 978-3-030-70568-8. DOI: 10.1007/978-3-030-70569-5_8. URL: https://research-repository.st-andrews.ac.uk/handle/10023/23145 (visited on 03/12/2025).

[62] Haibin Zhang, Bo Wen, and Jiajia Liu. “The Prediction of Heart Rate During Running Using Bayesian Combined Predictor”. In: *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*. ISSN: 2376-6506. June 2018, pp. 981–986. DOI: 10.1109/IWCMC.2018.8450342. URL: https://ieeexplore.ieee.org/abstract/document/8450342?casa_token=yVMMiP9N-IIAAAAA:Y-9MALGrHgLz5ljLBfOORKUtmXHAuz8zgUCoOzsv0vuF8Gj9n-GRhI7lQP4d6vNYdfZzx8Fp (visited on 03/12/2025).

[63] Kusprasapta Mutijarsa, Muhammad Ichwan, and Dina Budhi Utami. “Heart rate prediction based on cycling cadence using feedforward neural network”. In: *2016 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*. Oct. 2016, pp. 72–76. DOI: 10.1109/IC3INA.2016.7863026. URL: https://ieeexplore.ieee.org/abstract/document/7863026?casa_token=8_2ZVFzpoDgAAAAA:jwvfwkt4bTw2lHTuDljTF4XX8rq2lKCHCVVFQ2_xi8lbDVGowCkXafJdvJwZ_wjeM3whpYrp (visited on 03/12/2025).

[64] Zetao Zhu et al. “A fitness training optimization system based on heart rate prediction under different activities”. In: *Methods* 205 (Sept. 2022), pp. 89–96. ISSN: 1046-2023. DOI: 10.1016/j.ymeth.2022.06.006. URL: https://www.sciencedirect.com/science/article/pii/S1046202322001463 (visited on 02/24/2025).

[65] Xiaoli Liu et al. “Predicting the Heart Rate Response to Outdoor Running Exercise”. en. In: *2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS)*. Cordoba, Spain: IEEE, June 2019, pp. 217–220. ISBN: 978-1-7281-2286-1. DOI: 10.1109/CBMS.2019.00052. URL: https://ieeexplore.ieee.org/document/8787453/ (visited on 02/17/2025).

[66] Michael Mazzoleni et al. “A dynamical systems approach for the submaximal prediction of maximum heart rate and maximal oxygen uptake”. In: *Sports Engineering* 21 (Mar. 2018), pp. 31–41. DOI: 10.1007/s12283-017-0242-1.

[67] James Stirling et al. “A Model of Heart Rate Kinetics in Response to Exercise”. In: *Journal of Nonlinear Mathematical Physics Volume Supplement* 15 (Oct. 2008), pp. 426–436. DOI: 10.2991/jnmp.2008.15.s3.41.

[68] Maria S. Zakynthinaki. “Modelling Heart Rate Kinetics”. en. In: *PLOS ONE* 10.4 (Apr. 2015). Publisher: Public Library of Science, e0118263. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0118263. URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0118263 (visited on 12/14/2024).

[69] E. F. Coyle and J. González-Alonso. “Cardiovascular drift during prolonged exercise: new perspectives”. eng. In: *Exercise and Sport Sciences Reviews* 29.2 (Apr. 2001), pp. 88–92. ISSN: 0091-6331. DOI: 10.1097/00003677-200104000-00009.

[70] Jonathan E. Wingo et al. “Cardiovascular drift and Vo2max during cycling and walking in a temperate environment”. eng. In: *Aviation, Space, and Environmental Medicine* 83.7 (July 2012), pp. 660–666. ISSN: 0095-6562. DOI: 10.3357/asem.3246.2012.

[71] Julian D. Stevenson et al. “Prolonged cycling reduces power output at the moderate-to-heavy intensity transition”. eng. In: *European Journal of Applied Physiology* 122.12 (Dec. 2022), pp. 2673–2682. ISSN: 1439-6327. DOI: 10.1007/s00421-022-05036-9.

[72] Amine Souissi et al. “A new perspective on cardiovascular drift during prolonged exercise”. eng. In: *Life Sciences* 287 (Dec. 2021), p. 120109. ISSN: 1879-0631. DOI: 10.1016/j.lfs.2021.120109.

A

Data and Code Availability

The dataset used in this study is proprietary to Wahoo Fitness and cannot be publicly shared. Access to the dataset was granted under confidentiality agreements. The dataset and code were stored and processed on the High Performance Computing (HPC) cluster DAIC at TU Delft. In accordance with DAIC standard policy, all backups were permanently deleted two weeks after completion of the project. Throughout the project, access to the DAIC folder containing the data and code was restricted exclusively to the author.

The code developed for this study could be made available upon request, subject to prior approval from Wahoo Fitness due to proprietary dependencies. Additionally, all code and copies of the trained models are available on Wahoo Fitness' private GitHub repository, under access control.

B

Acknowledgements

This report was written for a computer science master thesis at TU Delft. As part of this thesis, a method for detecting anomalies in cycling data was developed. The project and data were provided by Wahoo Fitness under K. Hendrickx, with supervision from TU Delft under H. Hung and C. Raman. Major gratitude towards both parties, as the thesis would not have been possible without them. I would also like to express my deep gratitude to Wahoo Fitness for presenting the problem and offering valuable support throughout the project. In particular, I appreciate the day-to-day supervision from K. Hendrickx, and ideas and feedback from the rest of the team, with T. Camminady and M. Cassin in particular. T. Camminady's assistance with big data processing, and F. Van Nuland's, K. Hendrickx and B. Van Vliet contributions in providing anomaly examples. Additionally, special thanks to A. Nazaret and H. Kayange for small help with setting up their model and giving insights into design choices.

Research reported in this work was partially facilitated by computational resources and support of the Delft AI Cluster (DAIC) at TU Delft (RRID: SCR_025091), but remains the sole responsibility of the authors, not the DAIC team.

As a cyclist and computer scientist myself it was a very interesting project to combine both interests into one project that solved a real-world problem, rather than tackling a solely theoretical subject for my master thesis.

The report is designed to explain how the proposed methods work, their derivation, and the results they achieve. Readers of the report are expected to have a basic understanding of computer science.

This thesis has benefited from the use of ChatGPT, an AI language model developed by OpenAI, for assistance with writing and coding-related tasks. The AI was used as a tool to provide suggestions, clarify concepts, and improve the quality of the text and code. I acknowledge that while ChatGPT provided support, I retain full responsibility for the research, content, and conclusions presented in this work. All decisions regarding the structure, interpretation of data, and implementation of ideas were made independently, ensuring that the thesis reflects my own understanding and academic integrity.

C

Ethics Declaration

No authors were employed by Wahoo Fitness during this research. All research was done in a free collaboration between TU Delft and Wahoo Fitness. The author received small product samples from Wahoo Fitness in the context of this research. These products were provided without any obligation, financial compensation, or influence over the study design, analysis, or reporting. The research was conducted independently.

This research involved the analysis of anonymized cycling sensor data collected by Wahoo Fitness devices. No personal identifiable information (PII) was used or processed. Ethical approval was not required as the study only involved secondary analysis of pre-existing, anonymized sensor data without any interaction with human subjects. According to TU Delft's ethics guidelines for human-related research, studies that exclusively use fully anonymized, non-sensitive data without direct involvement of participants are exempt from mandatory ethical review. This study complied with all applicable TU Delft policies regarding research ethics and data protection.

D

Wilcoxon Signed-Rank Test Results

Model A	Model B	Median Diff (A - B)	p-value	Significant?	Better Model
Linear Regression	ODE	2.553	2.11e-122	Yes	ODE
Linear Regression	DBN LSTM	-1.300	1.25e-32	Yes	Linear Regression
Linear Regression	ODE Parameter Reduced	3.496	4.90e-275	Yes	ODE Parameter Reduced
ODE	DBN LSTM	-3.936	3.07e-228	Yes	ODE
ODE	ODE Parameter Reduced	0.711	5.68e-113	Yes	ODE Parameter Reduced
DBN LSTM	ODE Parameter Reduced	5.076	0.00	Yes	ODE Parameter Reduced

Table D.1: Statistical Comparison MAE from models using Wilcoxon Signed-Rank Test

Model A	Model B	Median Diff (A - B)	p-value	Significant?	Better Model
Linear Regression	ODE	-0.097	3.34e-145	Yes	ODE
Linear Regression	DBN LSTM	0.366	0.00	Yes	Linear Regression
Linear Regression	ODE Parameter Reduced	-0.157	0.00	Yes	ODE Parameter Reduced
ODE	DBN LSTM	0.457	0.00	Yes	ODE
ODE	ODE Parameter Reduced	-0.034	6.48e-227	Yes	ODE Parameter Reduced
DBN LSTM	ODE Parameter Reduced	-0.542	0.00	Yes	ODE Parameter Reduced

Table D.2: Statistical Comparison Correlation from models using Wilcoxon Signed-Rank Test

E

Lessons Learned

Throughout the thesis project, I encountered several technical, methodological, and personal lessons that significantly shaped both the process and the outcome. I have decided to write them down to help potential future master students in guiding them through their first big research project.

One of the most important lessons was the value of first understanding the problem space deeply before jumping into implementation. That said, there is a tension here: trying things quickly, failing fast, often leads to insights you would not get from just thinking. Balancing structured understanding with experimental iteration proved to be crucial. Both approaches are valid. The challenge is knowing when to switch between them. In hindsight, I jumped into implementation too quickly, wasting time that would have been better spent understanding the problem first. A bit of upfront analysis would have made it obvious that my initial approach didn't align with the problem's characteristics. That said, many of the more valuable lessons I learned during this thesis came from building something, seeing it fail, and then working backwards to understand why it failed.

Discussing the project with people who understood the domain very well, like my company supervisor K. Hendrickx, helped guiding the research. They didn't steer me in any particular direction. Instead, simply talking about the project helped me uncover new insights I had not considered on my own. The conversations themselves sparked ideas I wouldn't have reached in isolation. Surprisingly, speaking with those unfamiliar with the topic was also valuable, they asked basic but revealing questions that exposed assumptions or weaknesses in my understanding. At times, these discussions snapped me out of tunnel vision, challenging assumptions I had made or revealing parts of the thesis that were still unclear or poorly explained.

Time spent trying to "rediscover the wheel" can often be saved by locating relevant literature early. Learning how to quickly assess what is and is not useful was essential. Papers not only inform your direction, they shape how you frame your own work. In the beginning I tried creating a reconstruction framework myself, which took me a lot of time, while heart rate prediction models, like discussed in the literature review, already existed.

Regular check-ins with supervisors helped prevent misalignment. Often you start to set your own goals and requirements, which do not always align with the expectations from your supervisors and university. More importantly take the time to understand their feedback, rather than just implementing their feedback, try to understand were it is coming from and what the underlying lesson is. They have a lot of experience in research, and aim to guide you on developing that skill set.

A thesis needs narrative clarity. It's tempting to include technically interesting but tangential ideas. I learned to push such content into a background or appendix section and keep the main storyline focused. The same applies to experiments. They should be designed around the central hypothesis, not just as stand-alone experiments. Reading how other authors structured and justified their experiments helped sharpen mine. How results are presented affects how they're interpreted. Simply running experiments isn't enough, making their purpose, design, and meaning clear is just as important.

An important realization was that the problem was more physiological in nature than it was a classic anomaly detection task. Misframing it early led to a lot of effort in trying to make classical anomaly detection work. Being willing to adjust the framing, based on data and insight was essential.

Working with the DAIC supercomputer demands careful planning. Environment setup was slow, and run times on large datasets were long, making it especially frustrating when a bug appeared at the end of a multi-day job, producing no usable results. Testing locally on a small dataset is essential, and although I did this, some errors still slipped through due to differences between my local environment and DAIC. To avoid this, always run an example on a small dataset directly on DAIC before scaling up. When running large jobs, set checkpoints so you can recover from errors without starting over. The DAIC beginner course was also extremely valuable and something I highly recommend. It teaches you all the basics and how to estimate resource usage, which is critical. Finally, always build buffer time into your planning, technical delays are inevitable.