OpenMHC: Accelerating the Science of Wearable Foundation Models

Narayan Schuetz[1], **Yuze Bai**[2], **Lianggang Pan**[2], **Edgar Eggert**[1,5], **Favour Nerrise**[1], **Juan Delgado-SanMartin**[2], **Max Rosenblattl**[1], **Milana Gurbanova**[1], **Mohammad Asadi**[1], **Anders Johnson**[1], **Paul Schmiedmayer**[1], **Dennis Wang**[2], **Allan Lawrie**[2], **Daniel Seung Kim**[3], **Xin Liu**[3,4], **Akshay Paruchuri**[1,4], **Ehsan Adeli**[1], **Euan Ashley**[*1], **Kelly W. Zhang**[*2]

[1]Stanford University, [2]Imperial College London, [3]University of Washington, [4]Google, [5]Charité – Universitätsmedizin Berlin

Mobile and wearable devices offer an unprecedented opportunity for continuous, passive health monitoring and active health coaching. However, the largest wearable datasets are not publicly available for research, and leading wearable foundation models trained on such datasets are rarely open-weight or come with reproducible training code. To accelerate open science in wearable health, we release **OpenMyHeartCounts (OpenMHC)**, the largest and most comprehensive open-access wearable health dataset to date, alongside open-source implementations of recent wearable foundation models. OpenMHC, derived from over a decade of data collected through the My Heart Counts study app, includes >60 million hours of wearable data across 19 sensor channels (e.g., step count, heart rate, sleep, workouts) and up to 169 linked variables, including health, lifestyle, mood, and behavior from 11,894 consenting participants. Furthermore, we introduce a unified, open benchmark that enables standardized comparison of wearable health models across three tracks: health and behavior downstream prediction, multivariate data imputation, and time-series forecasting. We benchmark classical methods alongside recent wearable and multivariate time series foundation models. By open-sourcing data, code, and model weights at this unprecedented scale, we aim to democratize wearable health AI research and enable the community to drive open progress in this domain.

Code: https://github.com/AshleyLab/myheartcounts-dataset
Website: https://myheartcounts.stanford.edu/benchmark

1. Introduction

Mobile and wearable devices have enabled the continuous, passive collection of longitudinal health and behavior data at unprecedented scale [Piwek et al., 2016]. The richness of these streams, spanning physical activity, cardio-respiratory fitness, sleep, and more, has fueled a growing range of applications and research around health and wellness, ranging from chronic disease management and early detection [Perez et al., 2019, Lubitz et al., 2022, Ajufo et al., 2025] to digital phenotyping [Matias et al., 2026, Shim et al., 2024], just-in-time adaptive interventions [Javed et al., 2023, Schmiedmayer et al., 2026, Klasnja et al., 2019, Liao et al., 2020], and personalized health coaching [Schmiedmayer et al., 2026, Jörke et al., 2026]. Machine learning is often critical for understanding, processing, and effectively utilizing mobile and wearable data for these health applications. Progress in machine learning application areas has historically been driven by large-scale, high-quality, open datasets and benchmarks (e.g., computer vision advancements were catalyzed by ImageNet [Deng et al., 2009]). Yet for wearable and mobile health data, one of the most promising emerging health data modalities,

Correspondence to: schuetzn@stanford.edu, kelly.zhang@imperial.ac.uk

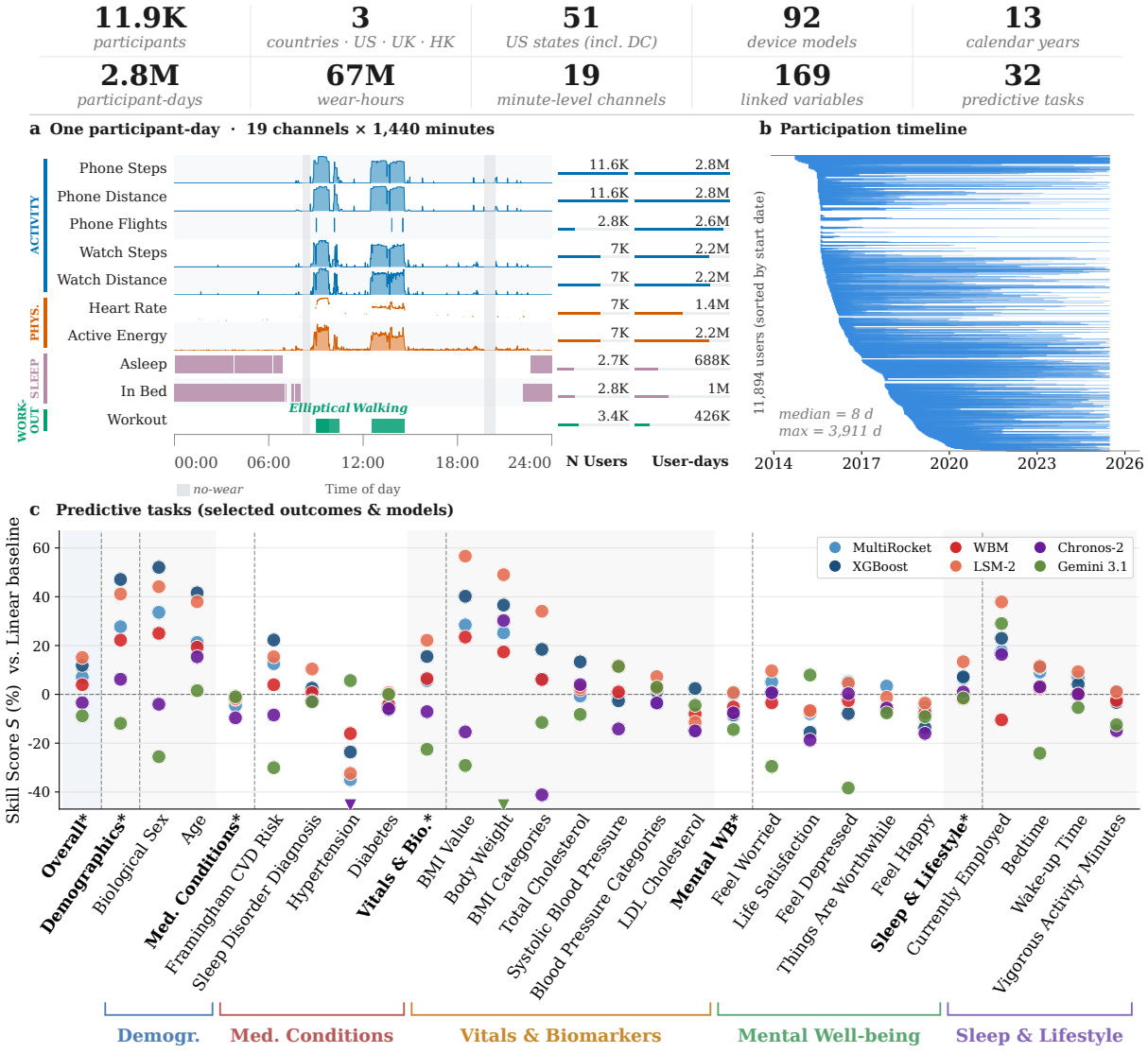

Figure 1 | **The OpenMHC dataset at a glance.** 11,894 participants across 3 countries (US, UK, HK), 51 US states, and 92 device models, contributing 2.82M participant-days (67M wear-hours) of minute-level sensor data over a 13-year calendar span, alongside 169 linked self-report and sparse HealthKit variables and 32 predictive tasks. **(a) Example of minute-level passive data participant-day** for one user as a 19×1,440 matrix spanning Activity, Watch-derived Physiology, Sleep, and 10 Workout types (compressed to a single channel for this visualization). Right columns report cohort-wide coverage: number of users contributing to the channel and total user-days. **(b) Participation timeline:** each line is one of the 11,894 shareable users, sorted by enrollment date, spanning their first to last contributed day. **(c) Skill scores (S, % vs. Linear baseline)** for selected predictive tasks, grouped into five health domains (Demographics, Medical Conditions, Vitals & Biomarkers, Mental Well-being, Sleep & Lifestyle), for six methods: MultiRocket, XGBoost, WBM, LSM-2, Chronos-2, and Gemini 3.1. Overall and per-domain summary columns are marked with $*$.

no such public foundational datasets exist. To date, larger-scale longitudinal wearable and mobile health datasets are private or strongly access-gated, limiting reproducibility and broader research participation [Truslow et al., 2024, Xu et al., 2025b, Narayanswamy et al., 2025]. Moreover, many

existing smaller datasets use specialized research-grade devices and sensors, making it harder to establish broadly applicable benchmarks.

This gap is particularly costly given recent evidence of the potential of wearable data: large-scale efforts from industry have demonstrated that modeling wearable and mobile data at scale yields impressive results, with scaling laws reminiscent of those observed in language and vision [Erturk et al., 2025, Narayanswamy et al., 2025, Xu et al., 2025b]. These results suggest that the field is not bottlenecked by algorithmic limitations, but by data access and sharing. Without open, large-scale datasets, progress risks foreclosing the kind of broad community participation that has propelled other fields. Our primary contributions are as follows:

1. **Curation and release of the largest and most comprehensive open access wearable health dataset.** We release OpenMHC, a dataset which provides longitudinal records collected over more than a decade from 11,894 consenting participants. The wearable dataset consists of 67 million hours of wearable data across 19 sensor channels—including step count, heart rate, sleep, and workouts—collected via Apple HealthKit from iPhones and Apple Watches, and other HealthKit-enabled wearable devices. The dataset also includes 169 self-reported and sparse HealthKit variables (diet, lifestyle, medical conditions, mood, etc.). This resource substantially surpasses all existing open-access wearable health datasets in the number of participants, duration of data collection, and comprehensiveness of linked health and lifestyle variables. The scale and longitudinal depth of this dataset enable, for the first time, large-scale pretraining and evaluation of foundation models for wearable health on real-world data, a regime previously inaccessible to the open science community.
2. **Establishing the first large-scale public benchmark for evaluating wearable health models.** To allow the community to better measure progress on this type of data, we develop a comprehensive set of evaluation tasks for wearable and mobile health models, covering three major tracks across downstream prediction and generative tasks. Our benchmark reflects the real-world complexity of wearable health data, including missingness, irregular sampling, and heterogeneity across individuals. (a) **Predictive:** We define supervised, predictive tasks for a wide range of self-reported health and behavior variables, including cardiovascular diseases, diabetes, and mental well-being. (b) **Generative (Imputation and Forecasting):** We conduct imputation on minute-level sensor data and 24-hour forecasting tasks for hourly data to address realistic missingness and future time series trajectory prediction over rolling windows.
3. **Open-source implementations of wearable foundation models and baselines.** While state-of-the-art wearable foundation models have been recently proposed, their implementations, model weights, and training data have often remained proprietary or inaccessible. We provide the *first, open-source implementations* of Apple's Wearable Health Behavior model (WBM) [Erturk et al., 2025] and Google's LSM-2 [Xu et al., 2025b], alongside a suite of classical ML and deep learning baselines. Paired with our OpenMHC dataset, this creates a fully reproducible ecosystem for developing and benchmarking wearable health models.

We release the OpenMHC public benchmark at `https://myheartcounts.stanford.edu/benchmark` and code to replicate our experiments and results at `https://github.com/AshleyLab/myheartcounts-dataset`.

2. Related Work

Wearable and Mobile Health Datasets. In Table 1, we provide an overview of existing wearable and mobile health datasets. We focus on health-related datasets, and not those for other tasks such as

activity recognition. All of Us [Singh et al., 2024, Bailey et al., 2025] is a large-scale US biobank that has high-quality health records and FitBit data from over 30k individuals; while this dataset is available via application, one must use the All of Us workbench to work with the data and be approved for access, which makes large-scale model training and development challenging for the broader community. Many large-scale biobank studies like the UK Biobank [Doherty et al., 2017] and NAKO Germany [Weber et al., 2024] only have accelerometer data from individuals for <10 days. Other key studies like the Apple Heart and Movement study [Truslow et al., 2024] and Google's Fitbit research dataset [Xu et al., 2025b, Narayanswamy et al., 2025] are not publicly available. There are a variety of smaller studies (≤ 500 individuals) [Jalin et al., 2026, Klasnja et al., 2019, Rossi et al., 2020, Xu et al., 2023]. An exception is HomeKit, which has 5k participants, but is limited to a monitoring period of at most four months per individual and has fewer sensor types.

Table 1 | **Wearable Health Datasets.** Modalities: ACC (Accelerometer), ST (Steps), HR (Heart rate), SL (Sleep), DT (Distance), FC (Floors Climbed), CB (Calories), WO (Workouts). N: number of subjects with wearable data. Open Access: ✓Yes, △ Partial, × No. When the exact number of wearable data hours was not explicitly reported, we calculated a conservative upper bound based on the number of participants and the study's overall duration.

Dataset	N	Hours	Source	Duration	Modalities	Open
China Kadoorie [Chen et al., 2023]	22k	3M	Axivity	7 days	ACC	△
NAKO Germany [Weber et al., 2024]	74k	10M	ActiGraph	7 days	ACC	△
UK Biobank [Doherty et al., 2017]	100k	17M	Axivity	7 days	ACC	△
NHANES [Aguiar et al., 2024]	4k	<868k	ActiGraph	9 days	ACC	✓
All of Us [Singh et al., 2024, Fulda et al., 2026]	30k	264M	Fitbit	5 years	ST,HR,SL	△
Google Fitbit [Narayanswamy et al., 2025]	165k	40M	Fitbit	2 years	ACC,ST,HR,PPG	×
Apple H&M [Truslow et al., 2024]	170k	2.5B	HealthKit	5 years	All*	×
HomeKit [Merrill et al., 2023]	5k	14M	Fitbit	4 mo.	ST,HR,SL	△
MHC Legacy [Hershman et al., 2019]	3.5k	<19M	HealthKit	8 mo.	ST,HR,SL,DT,FC,CB,WO	△
MMASH [Rossi et al., 2020]	22	528	BioBeats	1 day	ACC,ST,SL,HR	✓
HeartSteps V1 [Klasnja et al., 2019]	37	44k	Jawbone	6 wks	ST	✓
Roadmap HCT [Jalin et al., 2026]	332	<478k	Fitbit	4 mo.	ST,SL,HR	✓
GLOBEM [Xu et al., 2023]	500	<8M	Fitbit	2 years	ST,SL,DT	✓
OpenMHC (ours)	**12k**	**67M**	**HealthKit**	**10 years**	**ST,HR,SL,DT,FC,CB,WO**	✓

*All: ST, HR, SL, DT, FC, CB, WO, ACC, ECG, PPG

Foundation Models for Wearable Health Data. The predominant focus in foundation model (FM) research for wearable health data has been on learning representations from raw physiological signals, e.g., PPG [Saha et al., 2025, Pillai et al., 2025, Abbaspourazad et al., 2024b], ECG [Abbaspourazad et al., 2024a]. Exceptions include wearable accelerometer data [Xu et al., 2025a, Yuan et al., 2024]. These low-level sensor data are used for tasks such as activity recognition [Xu et al., 2025a, Yuan et al., 2024] and detecting health conditions like atrial fibrillation [Perez et al., 2019, Guo et al., 2019, Lubitz et al., 2021]. In contrast, far less work has explored foundation models operating at the level of behavioral and lifestyle measures, e.g., step counts, activity bouts, and sleep patterns, data which can actually be collected at scale by most major consumer devices (which stands in stark contrast to raw sensor data). The few wearable foundation models (WFMs) in this space include Apple's wearable health behavior foundation model (WBM) [Erturk et al., 2025], which was trained via contrastive learning to predict self-reported health outcomes and time-varying health detection tasks, and Google's Large Sensor Model (LSM) series [Narayanswamy et al., 2025, Xu et al., 2025b], which utilized a pretrained masked autoencoder for both predictive and generative tasks. However, these existing efforts [Erturk et al., 2025, Narayanswamy et al., 2025, Xu et al., 2025b] were developed and evaluated exclusively on large-scale proprietary datasets, and neither model weights nor training code

have been publicly released, making comparison, replication, or extension by the broader research community challenging. Our work aims to fill this gap by providing a first large-scale, open-source dataset for (pre-)training and evaluation, as well as public re-implementations of current wearable foundation models [Erturk et al., 2025, Xu et al., 2025b], to enable reproducible research.

3. OpenMHC-Dataset

Dataset Collection. OpenMHC is derived from the My Heart Counts (MHC) study, a large-scale, smartphone-based cardiovascular health study developed at Stanford University [McConnell et al., 2017, Hershman et al., 2019]. The study's iOS application was built using Apple's ResearchKit and launched on the US App Store in March 2015, open to English-speaking individuals aged ≥ 18 with a US-registered iPhone (iOS 8+). The study has since expanded to include a UK arm, was briefly available in Hong Kong, and will expand to an Android version in the near future [Schmiedmayer et al., 2026]. The MHC dataset includes de-identified data from its original release up until December 2025 (87% US, 13% UK participants). Participants enrolled digitally through an eConsent process and could designate their de-identified data for sharing either with Stanford only ("narrow") or with qualified researchers worldwide ("broad"), the latter of which makes up this dataset, while the former will serve as a private holdout set for future competitions. The app collects data both passively and actively. Passive streams include Apple HealthKit data sourced from built-in iPhone sensors as well as compatible wearable devices (e.g., Apple Watch, Withings, Peloton). Active data collection consists of health and lifestyle questionnaires (Appendix E.1.1). For a comprehensive description of the data collection protocol, study design, and available data types, we refer readers to an earlier, smaller-scale data release [Hershman et al., 2019]. Ethical oversight of the study was obtained from Stanford University's Research Compliance Office (Protocol #IRB-31409).

Dataset Characteristics. The total dataset contains data from 16,993 users (> 80M hours of passive Apple HealthKit data), of which we are allowed to publicly share data from 11,894 users (67M passive data hours), including users with >10 years of data history. Prevalent passive data are shared as minute-level daily matrices $d \in \mathbb{R}^{19\times1440}$, covering activity, physiology, sleep, and workouts (Figure 1a). Seven sparse HealthKit variables (e.g., VO2max) are shared separately, together with 162 self-reported variables related to demographics, disease status, lifestyle, diet, well-being, mindset, Covid, geographical location, and more (Figure 5). Participants come predominantly from the US, with around 1,145 from the UK and 80 from Hong Kong. US participants come from all over the US, with a focus in California (Figure 7). The median participant is 39 years old, male (77.1%), and white (82.4%). Due to high attrition, 50% of participants contributed for less than one week, while about 30% contributed for more than 1 month, enrollment/attrition behavior typical for first-generation mHealth apps [Amagai et al., 2022]. The cohort spans a wide range of consumer devices: 14 iPhone generations and 16 Apple Watch generations released between 2012 and 2025, covering roughly 40 to 50 overall phone and watch variants (see Table 5 for device types and Appendix C for additional dataset statistics).

Data Preprocessing and Release. All benchmark tasks use a shared data preprocessing pipeline, which includes removing anomalous or physiologically impossible values. Additionally, since HealthKit does not track missingness explicitly, it is not known whether an individual is inactive or simply not wearing/carrying the device. Thus, we developed heuristics for non-wear and missing periods (Appendix D; Figure 9) and report resulting coverage summaries in Figure 8. For reproducible research, we release official dataset splits at participant level (60% train, 10% val, 30% test), detailed participant and per-split statistics are found in Table 7. For development purposes, we also provide a tiny variant, MHC-XS, with 5% of the total users subsampled from the main split. The minute-level passive data are stored as daily matrices in a Huggingface dataset-readable format. Additional

data, like self-reported variables or sparse HealthKit metrics, are provided as json files. Data will be distributed through Harvard Dataverse with a data use agreement that prohibits re-identification and requires open-access publication of findings.

4. OpenMHC-Benchmark

Track 1: Predictive Tasks. Increasingly, wearable data is used to predict a variety of health outcomes, including cardiometabolic conditions [Guo et al., 2019, Lubitz et al., 2021, Metwally et al., 2026, Perez et al., 2019, Delgado-SanMartin et al., 2026], mental health outcomes [Abd-Alrazaq et al., 2023, Ahmed et al., 2023], and sleep-related outcomes [Walch et al., 2019, Retamales et al.]. We develop downstream prediction tasks that aim to probe a model's ability to predict a variety of health and behavior-related characteristics [Belinkov, 2022] based on **32** self-reported survey variables across five domains: Demographics (2), Medical conditions & risk (12), Vitals & blood biomarkers (8), Mental Well-being (5), Sleep & Lifestyle (5).

Note that the self-reported outcomes labels do not include diagnosis times, only the time at which the survey was conducted. Therefore, these prediction tasks should not be interpreted as early diagnostic prediction tasks. Instead, they evaluate whether models can predict health outcomes from all available wearable data collected, which may include both pre- and post-diagnosis data. We view this evaluation as a step toward future work on predicting early diagnoses from wearable data, as well as a useful way to assess the representations learned by wearable data foundation models. See Appendix E.2 for additional details on how we set up the prediction tasks. Note, the benchmark allows models to use different choices of time resolutions and filtering approaches for data quality, and will also allow for the incorporation of external data sources (e.g., historical weather information).

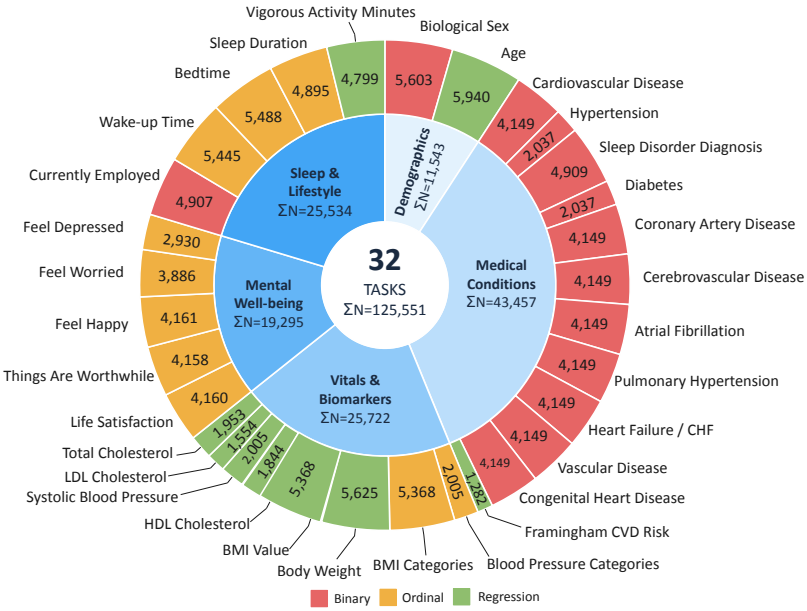

Figure 2 | Self-reported outcomes used for downstream prediction (**32 tasks** total), organized by domain. Numbers denote the number of participants per task, and sums represent the totals across all tasks in each domain.

Track 2A: Generative (Imputation). Missingness in free-living wearable data is the norm rather than the exception, e.g., due to removal for showering or charging, sensor malfunctions, and users forgetting to wear their device. Because any model operating on such data must either tolerate or recover from these gaps, imputation quality directly affects the utility of wearable data for almost all applications. We define a *Daily Imputation* task to evaluate imputation methods across six masking approaches that reflect real-world failure modes in minute-level, single-day sensor data (Figure 12). In this setting, models observe 24 hours of multichannel sensor data with subsets of channels and time intervals masked, and are tasked with reconstructing the masked-out values. Masking approaches are organized into two tiers: *(1) Structural masks* simulate data-collection failures that randomly mask

out blocks of times and/or channels in a way that does not depend on the underlying signal values, similar to Xu et al. [2025b]. *(2) Semantic masks* simulate missingness driven by user activity (e.g., sleep and workouts) and require a model to learn relationships across channels. We further consider a *Long-Context Imputation* task, in which models may leverage historical user data beyond a single day, as well as additional contextual features (e.g., demographics), to improve performance on the same 24-hour multichannel imputation task. See Appendix F.1 for additional details.

Track 2B: Generative (Forecasting). Wearable data forecasting—such as predicting step count, heart rate trends, sleep duration, or activity levels—has the potential to improve the timeliness and relevance of health coaching and interventions [Park et al., 2023, Nahum-Shani et al., 2016, Lee et al., 2024, Schmiedmayer et al., 2026, Khasentino et al., 2025]. In this task, we evaluate the ability of models to forecast future physical activity and health behaviors derived from iPhone and Apple Watch data, including step count, flights climbed, heart rate, sleep, and workout. For each individual, we construct an hourly-resolution data trajectory for each channel. Models are trained to forecast the next 24 hours of wearable data at an hourly resolution, given all available historical data across all channels from that individual (subject to model capacity constraints). We use a rolling evaluation setup: for each trajectory, the model predicts the next 24 hours, after which the history is advanced by 24 hours, and the process is repeated. For more details on the forecasting setup, see Appendix G.1.1.

4.1. Evaluation Approach

Each of our three tasks comprises multiple sub-tasks, including predicting various health outcomes and performing imputation and forecasting across sensor channels. We first describe evaluation metrics for each sub-task, then present unified metrics that aggregate performance across sub-tasks.

- **Track 1: Predictive Tasks.** All outcomes are either binary, ordinal, or real-valued. For evaluation, we use Area Under the Precision-Recall Curve (AUPRC) for binary outcomes (since there is significant class imbalance, see Table E.1.1), Spearman ρ for ordinal outcomes, and Pearson's r for real-valued outcomes.
- **Track 2A: Generative (Imputation).** Raw metrics are reported as Mean Absolute Error (MAE) for continuous channels and ROC AUC for binary channels and summarized using skill score and average win-rate %.
- **Track 2B: Generative (Forecasting).** Raw metrics are reported as Mean Absolute Error (MAE) for continuous channels and AUROC for binary channels, and summarized using skill score and average win-rate %.

Unified Evaluation Metrics: Rank and Skill Score. For each of the tracks, to provide an aggregated model performance metric across sub-tasks, we report (i) the *average rank* of each model across sub-tasks (rank 1 is the best model), and (ii) the *skill score* [Hyndman and Athanasopoulos, 2018, Shchur et al., 2025] of each model, which measures relative error reduction with respect to a fixed reference model, aggregated across sub-tasks. Specifically, the skill score for model j is computed as the following geometric mean across tasks:

$$S_j = 1 - \text{GeometricMean}\left(\text{clip}\left(\frac{E_{r,j}}{E_{r,b}}, \ell, u\right)\right), \tag{1}$$

where for task r, $E_{r,j}$ is the error of model j and $E_{r,b}$ is the error of the reference model b. Error ratios are clipped for numerical stability; use $\ell = 0.01$ and $u = 100$ (see Appendix B for more details). A skill score of 0 indicates performance on par with the reference, while a score of 0.2 indicates a 20% geometric-mean reduction in error. Negative scores indicate performance worse

than the reference. The geometric mean is the standard aggregation choice for error ratios to ensure proportional improvements and degradations are treated consistently [Hyndman and Athanasopoulos, 2018, Shchur et al., 2025].

Unified Fairness Evaluation. We report a *fairness* skill score that penalizes disparate performance across sensitive demographic subgroups (sex and age bracket). Our formulation builds upon the foundational principles of algorithmic fairness [Mitchell et al., 2021], specifically targeting *performance parity* across historical or demographic subgroups to prevent disproportionate error distributions [Rajkomar et al., 2018, Chen et al., 2024]. For each sensitive attribute $\mathcal{G}$, we compute the average *disparity* $D^{(\mathcal{G})}$, i.e., the absolute value of difference in error metric between all the unique pairs of subgroups in $\mathcal{G}$, averaged across all unique pairs. We then compute the fairness skill score for attribute $\mathcal{G}$ using

$$S_{\text{fair}}^{(\mathcal{G})} = 1 - \text{GeometricMean}\left(\text{clip}\left(\frac{D_{r,j}^{(\mathcal{G})}}{D_{r,b}^{(\mathcal{G})}}, \ell, u\right)\right)$$

where for task r, $D_{r,j}^{(\mathcal{G})}$ is the disparity for model j and $D_{r,b}^{(\mathcal{G})}$ is the disparity for the baseline model. The fairness skill score S_{fair} is the average of $S_{\text{fair}}^{(\mathcal{G})}$ across sensitive attributes $\mathcal{G}$ (Appendix B.2). For the generative tracks (imputation and forecasting), the per-task error E above—and hence the disparity $D^{(\mathcal{G})}$—is aggregated *per participant* (one value per participant per task). We then average these scores across demographic categories to ensure each group is weighted equally, rather than taking a simple, unweighted average of all tasks; Appendices F.4 and G.1.4 give the exact per-track aggregation.

5. Experiments and Results

We train and evaluate a range of models across our benchmark tracks on the full OpenMHC dataset, using the shared *train/test/val* splits. While we reasonably optimize each model given a limited compute budget, our goal is not to provide perfectly optimized models but rather to establish reproducible baselines for training and evaluation on OpenMHC as well as providing points of comparison and rough ballpark numbers across different model types for future work.

5.1. Track 1: Predictive Tasks

Models. Following Erturk et al. [2025], our first baseline is a (generalized) Linear model that takes age, sex, BMI, and summary statistics (i.e., mean and standard deviation) computed from hourly wearable data as input. Specifically, Linear uses ordinary least squares for continuous outcomes and logistic regression for binary outcomes. An commonly missing part in evaluations of wearable (foundation) models is comparing to domain-expertise guided feature engineering methods. To this end, we implemented one of the most prevalent gradient boosting models XGBoost [Chen and Guestrin, 2016], leveraging 495 hand-crafted features extracted from minute-level wearable data, spanning physical activity, sleep, circadian rhythm, as well as time and frequency domain constructs. We also compare to MultiRocket [Tan et al., 2022], which automatically extracts features using ensembles for time series classification, as well as GRU-D [Che et al., 2022], a supervised neural approach applied to hourly-level wearable data.

Additionally, we compare to probes trained on two pre-trained self-supervised learning foundation models for wearable data: WBM, a Mamba2-based contrastive encoder operating on weekly wearable tensors, our reimplementation of Apple's wearable behavior foundation model [Erturk et al., 2025], and Google's LSM-2, a ViT-style masked autoencoder operating on minute-level daily segments, our

reimplementation of LSM-2 [Xu et al., 2025b]. The probes use the last hidden layer and freeze the pre-trained models. Additionally, we compare to probes trained on representations from time series foundation models that support multivariate inputs fine-tuned on forecasting (CHRONOS-2 [Ansari et al., 2025] and TOTO 1.0 [Cohen et al., 2024]). See Appendix E.2 for exact details on the models and setup. We additionally evaluate Gemini-family LLMs on zero and few-shot prediction on these tasks in Appendix E.4, but they generally underperform baselines.

Results. Table 2 shows that no single model consistently performs the best across predicting all 32 outcomes or across all five health domains. LSM-2 performs best overall, achieving the best average rank ($R = 2.20^{+0.60}_{-0.11}$), and Skill Score ($S = +15.1^{+1.9}_{-2.3}$). XGBOOST is the closest competitor ($R = 3.52^{+0.43}_{-0.39}$, $S = +11.6^{+2.2}_{-2.3}$), performing best on *Demographics*. LSM-2 also leads on *Vitals & Blood Biomarkers*, *Mental Well-Being*, and *Sleep & Lifestyle*. On *Medical Conditions & Risk*, no model surpasses the LINEAR reference, though XGBOOST, LSM-2, and WBM come closest. Overall, LSM-2 is the most consistent method, but XGBOOST remains highly competitive. All models' fairness skill scores (discrepancies in performance across sensitive subgroups) are not statistically significantly different from the baseline; however, the greatest improvements over the baseline are from GRU-D, MULTIROCKET, TOTO, and CHRONOS-2.

Table 2 | **Prediction Tasks.** We report Average Rank R, Aggregate Skill Score S (in %; 0 =LINEAR is the reference), Fairness Skill Score S_{fair}, and category-specific Skill Scores across the five outcome categories: *Demographics, Medical Conditions & Risk, Vitals & Blood Biomarkers, Mental Well-Being, Sleep & Lifestyle*. Below, FT denotes fine-tuned; * denotes reimplementations of the original paper. Methods are grouped by class and ordered by Average Rank within each group. Values are point estimates on the held-out test split; subscripts and superscripts indicate the 95% bootstrap confidence interval (1000 resamples): the percentile interval for every column except S_{fair}, which uses the bias-corrected and accelerated (BCa) interval. Additional experimental details are in Appendix E.2.

Method	$R \downarrow$	$S \uparrow$	$S_{\text{fair}} \uparrow$	Demographics ↑	Medical Conditions & Risk ↑	Vitals & Blood Biomarkers ↑	Mental Well-Being ↑	Sleep & Lifestyle ↑
Statistical Models								
XGBOOST [Chen and Guestrin, 2016]	$3.52^{+0.43}_{-0.39}$	$+11.6^{+2.2}_{-2.3}$	$-3.5^{+17.2}_{-30.3}$	$\mathbf{+46.9}^{+5.4}_{-6.0}$	$-1.4^{+5.1}_{-5.7}$	$+13.6^{+4.1}_{-4.6}$	$-8.5^{+4.6}_{-4.7}$	$+7.3^{+4.5}_{-4.8}$
MULTIROCKET [Tan et al., 2022]	$3.71^{+0.59}_{-0.23}$	$+7.1^{+2.1}_{-2.1}$	$+11.2^{+18.5}_{-16.0}$	$+27.7^{+5.5}_{-5.2}$	$-4.5^{+5.8}_{-4.9}$	$+5.6^{+3.7}_{-4.5}$	$+0.2^{+4.3}_{-4.9}$	$+6.6^{+4.4}_{-4.3}$
LINEAR *(reference)*	$4.43^{+0.58}_{-0.20}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Supervised Neural Models								
GRU-D [Che et al., 2022]	$4.92^{+0.46}_{-0.44}$	$+3.1^{+2.5}_{-2.6}$	$\mathbf{+13.5}^{+18.2}_{-11.2}$	$+18.2^{+8.2}_{-8.0}$	$-6.2^{+3.7}_{-4.4}$	$+10.5^{+3.9}_{-4.5}$	$-3.9^{+4.5}_{-4.8}$	$-3.3^{+5.4}_{-6.4}$
Time-Series Foundation Models								
CHRONOS-2 (FT) [Ansari et al., 2025]	$5.98^{+0.32}_{-0.56}$	$-3.4^{+2.3}_{-2.9}$	$+7.4^{+20.3}_{-19.4}$	$+6.2^{+8.7}_{-9.0}$	$-9.6^{+2.9}_{-5.2}$	$-7.1^{+4.6}_{-6.4}$	$-7.6^{+4.5}_{-5.4}$	$+0.9^{+4.5}_{-5.4}$
TOTO (FT) [Cohen et al., 2024]	$6.47^{+0.03}_{-0.79}$	$-5.0^{+2.3}_{-2.9}$	$+8.2^{+20.4}_{-16.6}$	$-0.9^{+8.9}_{-9.4}$	$-7.6^{+2.9}_{-4.7}$	$-4.5^{+4.6}_{-6.1}$	$-9.2^{+4.3}_{-4.9}$	$-2.6^{+5.3}_{-5.5}$
Wearable Foundation Models								
LSM-2* [Xu et al., 2025b]	$\mathbf{2.20}^{+0.60}_{-0.11}$	$\mathbf{+15.1}^{+1.9}_{-2.3}$	$+2.3^{+20.7}_{-21.2}$	$+41.1^{+5.6}_{-6.4}$	$-1.8^{+4.0}_{-4.4}$	$\mathbf{+22.2}^{+3.4}_{-4.2}$	$\mathbf{+0.7}^{+4.0}_{-4.2}$	$\mathbf{+13.4}^{+4.0}_{-4.2}$
WBM* [Erturk et al., 2025]	$4.76^{+0.31}_{-0.54}$	$+3.3^{+1.4}_{-1.5}$	$-5.3^{+15.1}_{-25.8}$	$+22.2^{+4.3}_{-4.6}$	$-2.0^{+2.3}_{-2.1}$	$+6.6^{+2.3}_{-2.6}$	$-7.5^{+3.6}_{-3.5}$	$-2.8^{+3.1}_{-3.4}$

5.2. Track 2A: Imputation (Generative)

Methods. For the *single-day imputation* task, we evaluate eleven imputation methods spanning statistical baselines (MEAN, MODE, LINEAR INTERPOLATION, LAST OBSERVED CARRY FORWARD/LOCF, and baseline methods based on diurnal temporal statistics), existing neural imputation models (BRITS [Cao et al., 2018], DLINEAR [Zeng et al., 2023], FEDFORMER [Zhou et al., 2022], TIMESNET [Wu et al., 2023]), and LSM-2, our reimplementation of Google's LSM-2 masked autoencoder [Xu et al., 2025b]. BRITS, DLINEAR, FEDFORMER, and TIMESNET we use implementations from the PyPOTS library [Du, 2023]. For *long-context imputation*, we evaluate methods that can use up to seven days of a user's historical data to improve imputation. Specifically, we use baseline methods based on personalized statistics over 7 days, a 7-DAY DLINEAR, and LSM-2-SPARSE, which pairs the frozen daily encoder with a sparse cross-day decoder (Appendix F.2.2). Methods are described

Table 3 | **Imputation Results.** We report Average Rank R, Aggregate Skill Score S (in %; 0 = LOCF reference), Fairness Skill Score S_{fair}, and Channel-Specific Skill Scores for the following channels: *Activity, Physiology, Sleep, Workout*. Finally, we also report performance on all *Semantic* masking approaches (see Appendix F). Single-day imputation method results are in the upper section of the table; long-context imputation method results ($\geq 7 \times 1440$ time steps) are below. Values are point estimates on the held-out test split; sub/superscripts give the 95% bootstrap confidence interval (1000 resamples): the percentile interval for every column except S_{fair}, which uses the bias-corrected and accelerated (BCa) interval.

Method	$R \downarrow$	$S \uparrow$	$S_{\text{fair}} \uparrow$	Activity ↑	Physio. ↑	Sleep ↑	Workout ↑	Semantic ↑
Single-day imputation								
Statistical Models								
Linear	$6.4^{+0.1}_{-0.1}$	$+21.5^{+0.7}_{-1.2}$	$+34.7^{+11.6}_{-6.5}$	$+4.5^{+0.5}_{-0.5}$	$+9.8^{+0.3}_{-0.4}$	$+62.6^{+0.4}_{-1.9}$	$+56.5^{+2.2}_{-3.0}$	$-0.8^{+1.9}_{-1.9}$
LOCF *(reference)*	$7.7^{+0.1}_{-0.1}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Temporal mode	$9.3^{+0.1}_{-0.1}$	$-6.2^{+2.4}_{-2.3}$	$+55.9^{+11.2}_{-12.7}$	$\mathbf{+46.5}^{+0.6}_{-0.6}$	$-0.7^{+1.1}_{-1.1}$	$-13.9^{+5.8}_{-5.0}$	$-69.4^{+7.3}_{-7.6}$	$-11.8^{+4.2}_{-4.3}$
Mode	$9.9^{+0.0}_{-0.1}$	$-27.3^{+2.7}_{-2.5}$	$+91.2^{+0.7}_{-0.8}$	$\mathbf{+46.5}^{+0.6}_{-0.6}$	$-0.8^{+1.1}_{-1.0}$	$-380.7^{+19.7}_{-18.1}$	$-69.4^{+7.3}_{-7.6}$	$-12.0^{+4.2}_{-4.2}$
Temporal mean	$9.5^{+0.1}_{-0.1}$	$-31.2^{+3.1}_{-3.2}$	$-28.9^{+44.9}_{-45.3}$	$-30.7^{+2.1}_{-2.4}$	$-18.4^{+1.3}_{-1.4}$	$+59.9^{+2.9}_{-3.0}$	$-2.1^{+6.1}_{-6.0}$	$-93.0^{+6.7}_{-7.2}$
Mean	$12.4^{+0.0}_{-0.0}$	$-119.7^{+4.7}_{-4.4}$	$\mathbf{+92.2}^{+0.5}_{-0.7}$	$-36.3^{+2.2}_{-2.4}$	$-25.5^{+1.3}_{-1.4}$	$-380.7^{+19.7}_{-18.1}$	$-69.4^{+7.3}_{-7.6}$	$-149.8^{+8.5}_{-9.4}$
Neural Models								
LSM-2 [Xu et al., 2025b]	$\mathbf{3.6}^{+0.1}_{-0.0}$	$\mathbf{+61.4}^{+0.5}_{-1.2}$	$+57.6^{+9.6}_{-8.2}$	$+40.0^{+0.7}_{-0.8}$	$\mathbf{+31.4}^{+0.5}_{-0.5}$	$\mathbf{+90.0}^{+0.2}_{-0.8}$	$\mathbf{+94.9}^{+0.2}_{-0.6}$	$\mathbf{+30.2}^{+2.2}_{-2.4}$
BRITS [Cao et al., 2018]	$7.1^{+0.1}_{-0.1}$	$+6.8^{+1.8}_{-1.9}$	$-30.3^{+30.4}_{-30.0}$	$+18.8^{+1.1}_{-1.1}$	$-28.5^{+1.7}_{-1.7}$	$+39.0^{+2.1}_{-2.7}$	$+28.0^{+4.8}_{-5.1}$	$-5.7^{+3.3}_{-3.6}$
DLinear [Zeng et al., 2023]	$7.4^{+0.1}_{-0.1}$	$-5.7^{+2.1}_{-2.1}$	$+30.1^{+12.9}_{-6.4}$	$+29.3^{+0.7}_{-0.8}$	$-5.1^{+1.0}_{-1.0}$	$-11.1^{+4.0}_{-3.7}$	$+58.2^{+3.0}_{-3.4}$	$-45.9^{+4.9}_{-5.0}$
FEDformer [Zhou et al., 2022]	$10.4^{+0.1}_{-0.1}$	$-53.7^{+3.3}_{-3.0}$	$+35.4^{+20.1}_{-9.4}$	$+28.9^{+0.7}_{-0.8}$	$-14.6^{+1.1}_{-1.1}$	$-214.6^{+13.2}_{-12.2}$	$-53.7^{+7.6}_{-7.7}$	$-67.7^{+5.8}_{-6.3}$
TimesNet [Wu et al., 2023]	$10.0^{+0.1}_{-0.1}$	$-66.0^{+3.5}_{-3.5}$	$+6.2^{+27.3}_{-17.4}$	$+9.6^{+1.2}_{-1.4}$	$-18.6^{+1.3}_{-1.3}$	$-216.2^{+14.0}_{-13.0}$	$+0.4^{+7.2}_{-7.3}$	$-103.2^{+6.7}_{-7.9}$
Long-context imputation ($\geq 7 \times 1440$ time steps)								
Statistical Models								
Personalized temp. mean	$8.2^{+0.1}_{-0.1}$	$-7.7^{+2.8}_{-2.8}$	$-50.7^{+35.7}_{-67.6}$	$+1.1^{+1.5}_{-1.6}$	$-5.9^{+1.1}_{-1.2}$	$+58.9^{+3.3}_{-3.6}$	$+15.7^{+6.3}_{-6.5}$	$-49.5^{+5.3}_{-5.4}$
Personalized mode	$9.8^{+0.1}_{-0.1}$	$-26.1^{+2.6}_{-2.4}$	$\mathbf{+76.4}^{+4.7}_{-5.5}$	$\mathbf{+46.6}^{+0.6}_{-0.6}$	$+1.8^{+1.0}_{-1.0}$	$-383.1^{+19.7}_{-18.7}$	$-69.4^{+7.3}_{-7.6}$	$-10.5^{+4.1}_{-4.1}$
Personalized mean	$12.4^{+0.1}_{-0.1}$	$-114.1^{+4.4}_{-4.3}$	$-26.7^{+37.2}_{-26.2}$	$-4.1^{+1.6}_{-1.7}$	$-12.6^{+1.2}_{-1.2}$	$-437.7^{+20.9}_{-19.3}$	$-140.0^{+10.4}_{-11.2}$	$-132.5^{+8.2}_{-9.2}$
Neural Models								
LSM-2-Sparse (7-day)	$\mathbf{3.2}^{+0.1}_{-0.1}$	$\mathbf{+64.7}^{+0.4}_{-1.2}$	$+68.2^{+6.0}_{-1.7}$	$+41.0^{+0.7}_{-0.7}$	$\mathbf{+34.6}^{+0.5}_{-0.5}$	$\mathbf{+92.2}^{+0.0}_{-0.7}$	$\mathbf{+95.7}^{+0.1}_{-0.5}$	$\mathbf{+34.6}^{+2.1}_{-2.5}$
DLinear (7-day) [Zeng et al., 2023]	$8.6^{+0.1}_{-0.1}$	$-28.3^{+2.5}_{-2.6}$	$+10.2^{+25.0}_{-20.4}$	$+19.9^{+0.9}_{-1.0}$	$-2.7^{+0.9}_{-0.9}$	$-40.0^{+5.3}_{-4.9}$	$+22.9^{+4.5}_{-5.6}$	$-69.5^{+5.8}_{-6.0}$

in detail in Appendix F.2; hyperparameter search spaces and selected configurations are given in Appendix F.6.

Results. Table 3 summarises aggregate performance across all masking scenarios. LSM-2-Sparse (7-day) attains the best Average Rank and aggregate Skill Score, followed by the daily LSM-2; the two also lead the Physiology, Sleep, and Workout channels and the Semantic scenarios. The main exception is the Activity channels, where the constant mode-based baselines score highest because some activity channels are mostly zero. Among the remaining methods, only Linear interpolation exceeds the LOCF reference (S=0); all other statistical baselines fall below it. The PyPOTS-trained neural baselines (BRITS, TimesNet, FEDformer) cluster around or below LOCF and far below LSM-2, likely because they are trained with random masking objectives (masked imputation training and observed reconstruction) on small missing patches, and thus fail to extrapolate over the long, structured gaps in our scenarios, whereas the wearable-tailored LSM-2 / LSM-2-Sparse masking explicitly trains on realistic missingness patterns Xu et al. [2025b]. Full results across masking scenarios are in Appendix F.5.

5.3. Track 2B: Forecasting (Generative)

Models. We evaluate a set of statistical baselines (Seasonal Naive, AutoARIMA, AutoETS [Hyndman and Athanasopoulos, 2018]), which are applied independently to each channel and fitted

Table 4 | **Forecasting Results.** We report Average Rank R, Aggregate Skill Score S (in %; 0 = Seasonal Naive reference), Fairness-adjusted Skill Score S_{fair}, and category-specific Skill Scores for *Activity*, *Physiology*, *Sleep*, and *Workout*. FT denotes fine-tuned. Values are point estimates on the held-out test split; subscripts and superscripts indicate the 95% bootstrap confidence interval (1000 resamples): the percentile interval for every column except S_{fair}, which uses the bias-corrected and accelerated (BCa) interval.

Method	$R \downarrow$	$S \uparrow$	$S_{\text{fair}} \uparrow$	Activity ↑	Physio. ↑	Sleep ↑	Workout ↑
Statistical Methods							
Seasonal Naive	$7.72^{+0.08}_{-0.08}$	0.0	0.0	0.0	0.0	0.0	0.0
AutoARIMA	$7.64^{+0.07}_{-0.07}$	$+5.9^{+2.5}_{-2.6}$	$-29.3^{+13.7}_{-37.0}$	$-1.8^{+1.0}_{-1.2}$	$-9.0^{+1.7}_{-1.6}$	$+7.0^{+5.1}_{-5.3}$	$+24.0^{+6.7}_{-6.7}$
AutoETS	$7.07^{+0.08}_{-0.08}$	$+14.3^{+2.5}_{-2.1}$	-308.4^{+28}_{-74}	$+0.6^{+1.0}_{-1.0}$	$-26.8^{+2.3}_{-2.8}$	$+37.6^{+3.3}_{-3.3}$	$\mathbf{+31.4}^{+5.7}_{-5.6}$
Neural Models							
MixLinear [Ma et al., 2024]	$5.68^{+0.09}_{-0.10}$	$+29.2^{+1.8}_{-1.7}$	$+9.3^{+16.4}_{-11.4}$	$+23.4^{+1.1}_{-1.1}$	$+13.4^{+1.8}_{-1.9}$	$+64.6^{+1.8}_{-1.7}$	$-7.2^{+9.2}_{-9.3}$
DLinear [Zeng et al., 2023]	$4.63^{+0.10}_{-0.10}$	$+35.9^{+1.8}_{-1.9}$	$\mathbf{+11.9}^{+17.3}_{-8.8}$	$+25.0^{+0.9}_{-1.0}$	$+16.5^{+1.7}_{-1.7}$	$\mathbf{+71.5}^{+1.6}_{-1.6}$	$+5.5^{+9.4}_{-10.7}$
SegRNN [Lin et al., 2025]	$4.35^{+0.09}_{-0.08}$	$+34.6^{+1.4}_{-1.6}$	$+7.8^{+11.0}_{-21.0}$	$+25.4^{+1.0}_{-1.1}$	$+20.8^{+1.4}_{-1.4}$	$+68.2^{+1.8}_{-1.9}$	$+2.5^{+6.5}_{-8.7}$
Time-Series Foundation Models							
Toto [Cohen et al., 2024]	$5.49^{+0.09}_{-0.10}$	$+26.8^{+1.7}_{-1.7}$	$-2.1^{+15.2}_{-23.8}$	$+29.2^{+1.1}_{-1.2}$	$+6.8^{+1.9}_{-1.7}$	$+50.5^{+2.5}_{-2.4}$	$+11.9^{+6.1}_{-5.9}$
Toto (FT)	$4.68^{+0.09}_{-0.09}$	$+30.9^{+2.8}_{-2.2}$	$-4.7^{+5.6}_{-26.3}$	$+29.5^{+1.0}_{-1.1}$	$+26.1^{+0.8}_{-0.8}$	$+46.1^{+2.4}_{-2.5}$	$+18.8^{+11.5}_{-9.8}$
Chronos-2 [Ansari et al., 2025]	$4.17^{+0.09}_{-0.09}$	$+36.4^{+2.0}_{-1.8}$	$+0.7^{+10.3}_{-23.8}$	$+30.5^{+1.0}_{-1.1}$	$+26.5^{+0.8}_{-0.8}$	$+62.3^{+2.2}_{-2.1}$	$+14.8^{+8.0}_{-8.1}$
Chronos-2 (FT)	$\mathbf{3.56}^{+0.08}_{-0.09}$	$\mathbf{+37.6}^{+2.1}_{-1.9}$	$-1.4^{+8.9}_{-25.4}$	$\mathbf{+30.7}^{+1.0}_{-1.1}$	$\mathbf{+26.9}^{+0.8}_{-0.8}$	$+63.9^{+2.1}_{-2.0}$	$+17.0^{+8.1}_{-8.7}$

separately for each participant using only their observed trajectory at test time (i.e., without access to training data). We also evaluate several deep learning sequence models trained from scratch (DLinear [Zeng et al., 2023], MixLinear [Ma et al., 2024], and SegRNN [Lin et al., 2025]), which are trained as global models across participants in the training set using multivariate inputs from all channels. Finally, we evaluate two time-series foundation models Chronos-2 [Ansari et al., 2025] and Toto 1.0 [Cohen et al., 2024], which we chose due to their ability to incorporate multi-channel inputs. For the foundation models, we evaluate both their zero-shot performance and their performance after fine-tuning on the participants in the training set (Appendix G.2).

Results. Table 4 shows that the fine-tuned Chronos-2 achieves the strongest overall performance, obtaining the best average rank ($R = 3.55$) and aggregate Skill Score ($S = +37.6$). The non-fine-tuned Chronos-2 is the closest competitor, ranking second in both average rank ($R = 4.17$) and aggregate Skill Score ($S = +36.4$), with the from-scratch DLinear close behind ($S = +35.9$, the strongest model trained from scratch). The fairness-adjusted Skill Score tells a different story: DLinear attains the highest $S_{\text{fair}} = +11.9$, while the foundation models—including fine-tuned Chronos-2 ($S_{\text{fair}} = -1.4$)—sit near or below the Seasonal Naive reference, indicating that their accuracy gains do not translate into more equitable performance across demographic subgroups. We also report skill scores across 4 sensor categories: *Activity, Physiology, Sleep, Workout*. At the category level, Chronos-2 (FT) performs best on Activity and Physiology and DLinear on Sleep, while the highest Workout skill comes from the statistical AutoETS baseline (+31.6), albeit with poor overall and fairness scores. See Appendix G for further details.

6. Discussion

OpenMHC introduces, to our knowledge, the first open, AI-ready wearable health dataset at a scale sufficient to support and democratize the development of foundation models on real-world consumer

wearable device data. Through the development of this contribution, we have several key findings:

1. Self-supervised pretrained can improve performance, but its effectiveness depends critically on the choice of training objective. We compare three classes of foundation models trained on OpenMHC: WBM [Erturk et al., 2025], trained with a contrastive objective over dense weekly segments; LSM-2 [Xu et al., 2025b], trained with a reconstruction objective; and time-series FMs trained with next-token prediction objectives. We observe substantial variation in downstream performance across these approaches (Table 2). In particular, LSM-2 consistently outperforms alternatives on prediction and imputation tasks, suggesting that its wearable data tailored masked reconstruction is well-suited to the sparse and irregular nature of this data. In contrast, WBM underperforms in our setting, partly due to its reliance on high-quality, contiguous weekly segments, which reduces the amount of usable training data. These results indicate that pretraining objectives and flexible architectures that explicitly accommodate missingness and partial observations are better aligned with real-world wearable data.

2. Well-crafted simple models remain highly competitive with, and often outperform, more complex architectures. A well-tuned XGBoost model achieves the second-best overall performance across prediction tasks, outperforming many neural models, trailing only LSM-2. This highlights the importance of including strong simple baselines when evaluating models on wearable health tasks but also highlights that some of the downstream medical and mental health conditions are hard to improve upon beyond simple baselines.

3. Leveraging longitudinal data is a promising direction to improve performance. We find that incorporating longitudinal context provides a clear benefit: extending LSM-2 with a 7-day sparse cross-day decoder yields a substantial improvement in imputation performance. This highlights that leveraging an extended personal history is a promising frontier for future wearable ML research.

Looking ahead, OpenMHC enables several concrete research directions. First, its scale makes it possible to study scaling laws for wearable foundation models, which remain largely unexplored. Second, our results highlight the need to understand cross-device and cross-cohort transfer, particularly given differences in data quality and missingness patterns (e.g., extending to Fitbit datasets). Third, while the Gemini-family LLM and agentic probes evaluated here perform poorly (Appendix E.4), the benchmark provides a controlled setting for developing more effective interfaces between large language models and longitudinal health data. Finally, future releases of My Heart Counts, including a planned Android cohort, will further expand the dataset and broaden its applicability.

7. Conclusion

We present OpenMHC, the first large-scale, open, AI-ready consumer wearable dataset and benchmark, comprising over 60M hours of real-world consumer wearable data. Our evaluation reveals several insights that we believe will shape the development of wearable foundation models beyond current data silos: pretraining objectives and architecture matter for WFMs, revealing stark differences between current approaches; strong tree-boosting baselines remain surprisingly competitive, cautioning against complexity for its own sake; and longitudinal context offers a promising but underexplored avenue for improving model performance.

Acknowledgements

We would like to thank everyone who was involved in My Heart Counts to make this project possible, especially Steve Hershman, Anna Sherbina, and the team at Sage Bionetworks, as well as all the My Heart Counts participants who contributed their data. This work is supported by the Imperial BHF

Research Excellence Award (4) (RE/24/130023) and NIHR Imperial Biomedical Research Centre. N.S. was supported by the Wu-Tsai Human Performance Alliance as a Postdoctoral Fellow and by Swiss National Science Foundation under Postdoc Mobility fellowship 210803. D.S.K. was supported by the Wu-Tsai Human Performance Alliance as a Clinician-Scientist Fellow, the Stanford Center for Digital Health as a Digital Health Scholar, the Pilot Grant from the Stanford Center for Digital Health, and NIH 1L30HL170306. D.S.K. is presently supported by NIH 9L30DK144879-02, the Robert A. Winn Excellence in Clinical Trials Career Development Award, the American Heart Association (AHA) Career Development Award (AHA 25CDA1436622), and the American Diabetes Association (ADA) Pathway to Stop Diabetes Initiator Award (7-25-INI-11). This project was also supported by HAI Google Cloud Credits and directly by Google's GCP research credits program.

Data Release

We release the extra small version of the dataset through Harvard Dataverse here: https://doi.org/10.7910/DVN/ZYMJF6. The full dataset will be released upon publication of the manuscript. We release the OpenMHC public benchmark at https://myheartcounts.stanford.edu/openmhc and code to replicate our experiments and results at https://github.com/AshleyLab/myheartcounts-dataset.

References

Salar Abbaspourazad, Oussama Elachqar, Andrew Miller, Saba Emrani, Udhyakumar Nallasamy, and Ian Shapiro. Large-scale training of foundation models for wearable biosignals. In *The Twelfth International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=pC3WJHf51j.

Salar Abbaspourazad, Anshuman Mishra, Joseph Futoma, Andrew C Miller, and Ian Shapiro. Wearable accelerometer foundation models for health via knowledge distillation. *arXiv preprint arXiv:2412.11276*, 2024b.

Alaa Abd-Alrazaq, Rawan AlSaad, Farag Shuweihdi, Arfan Ahmed, Sarah Aziz, and Javaid Sheikh. Systematic review and meta-analysis of performance of wearable artificial intelligence in detecting and predicting depression. 6(1):84, 2023. ISSN 2398-6352. doi: 10.1038/s41746-023-00828-5. URL https://doi.org/10.1038/s41746-023-00828-5.

Elroy J Aguiar, Dusty T Turner, James D Pleuss, Peixuan Zheng, Cristal J Benitez, and Scott W Ducharme. Daily and peak monitor independent movement summary (mims) values associated with metabolic syndrome: Nhanes 2011–12 and 2013–14. *Scandinavian Journal of Medicine & Science in Sports*, 34(11):e14762, 2024.

Arfan Ahmed, Sarah Aziz, Mahmood Alzubaidi, Jens Schneider, Sara Irshaidat, Hashem Abu Serhan, Alaa A. Abd-alrazaq, Barry Solaiman, and Mowafa Househ. Wearable devices for anxiety & depression: A scoping review. 3:100095, 2023. ISSN 2666-9900. doi: https://doi.org/10.1016/j.cmpbup.2023.100095. URL https://www.sciencedirect.com/science/article/pii/S2666990023000046.

Ezimamaka Ajufo, Shinwan Kany, Joel T Rämö, Timothy W Churchill, J Sawalla Guseh, Krishna G Aragam, Patrick T Ellinor, and Shaan Khurshid. Accelerometer-measured sedentary behavior and risk of future cardiovascular disease. *Journal of the American College of Cardiology*, 85(5):473–486, 2025.

Saki Amagai, Sarah Pila, Aaron J Kaat, Cindy J Nowinski, and Richard C Gershon. Challenges in participant engagement and retention using mobile health apps: literature review. *Journal of medical Internet research*, 24(4):e35120, 2022.

Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, et al. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv:2510.15821*, 2025.

Melissa J Azur, Elizabeth A Stuart, Constantine Frangakis, and Philip J Leaf. Multiple imputation by chained equations: what is it and how does it work? *International journal of methods in psychiatric research*, 20(1):40–49, 2011.

Caitlin P Bailey, Kevin W Dodd, James J McClain, Isabell Seo, William Wheeler, and Dana L Wolff-Hughes. Fitbit physical activity and sleep data in the all of us research program: data exploration and processing considerations for research. *Medicine and science in sports and exercise*, pages 10–1249, 2025.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. BRITS: Bidirectional recurrent imputation for time series. In *Advances in Neural Information Processing Systems*, volume 31, pages 6775–6785, 2018.

Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. 8(1):6085, 2022. ISSN 2045-2322. doi: 10.1038/s41598-018-24271-9. URL https://doi.org/10.1038/s41598-018-24271-9.

Jane Chen, Peter O'Reilly, and Marzyeh Ghassemi. Navigating fairness aspects of clinical prediction models. *PMC Clinical AI / Nature Medicine and Healthcare Review*, 12:104–115, 2024.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939785. URL https://doi.org/10.1145/2939672.2939785.

Yuanyuan Chen, Shing Chan, Derrick Bennett, Xiaofang Chen, Xianping Wu, Yalei Ke, Jun Lv, Dianjianyi Sun, Lang Pan, Pei Pei, et al. Device-measured movement behaviours in over 20,000 china kadoorie biobank participants. *International Journal of Behavioral Nutrition and Physical Activity*, 20(1):138, 2023.

Ben Cohen, Emaad Khwaja, Kan Wang, Charles Masson, Elise Ramé, Youssef Doubli, and Othmane Abou-Amal. Toto: Time series optimized transformer for observability. *arXiv preprint arXiv:2407.07874*, 2024.

Ben Cohen, Emaad Khwaja, Youssef Doubli, Salahidine Lemaachi, Chris Lettieri, Charles Masson, Hugo Miccinilli, Elise Ramé, Qiqi Ren, Afshin Rostamizadeh, et al. This time is different: An observability perspective on time series foundation models. *arXiv preprint arXiv:2505.14766*, 2025.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Justin Cosentino, Anastasiya Belyaeva, Xin Liu, Nicholas A Furlotte, Zhun Yang, Chace Lee, Erik Schenck, Yojan Patel, Jian Cui, Logan Douglas Schneider, et al. Towards a personal health large language model. *arXiv preprint arXiv:2406.06474*, 2024.

Juan A Delgado-SanMartin, Merve Keles, Niamh Errington, Narayan Schuetz, Anders Johnson, Varsha Gupta, Steve Hershman, Mark Toshner, Martin R Wilkins, David G Kiely, et al. Assessing the feasibility of using smartphone data to identify risk of idiopathic pulmonary arterial hypertension. *npj Cardiovascular Health*, 3(1):16, 2026.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

Thomas J DiCiccio and Bradley Efron. Bootstrap confidence intervals. *Statistical science*, 11(3): 189–228, 1996.

Aiden Doherty, Dan Jackson, Nils Hammerla, Thomas Plötz, Patrick Olivier, Malcolm H Granat, Tom White, Vincent T Van Hees, Michael I Trenell, Christoper G Owen, et al. Large scale population assessment of physical activity using wrist worn accelerometers: the uk biobank study. *PloS one*, 12 (2):e0169649, 2017.

Wenjie Du. PyPOTS: A Python Toolkit for Data Mining on Partially-Observed Time Series. *KDD 2023 MiLeTS*, 2023.

Ralph B D'Agostino Sr, Ramachandran S Vasan, Michael J Pencina, Philip A Wolf, Mark Cobain, Joseph M Massaro, and William B Kannel. General cardiovascular risk profile for use in primary care: the framingham heart study. *Circulation*, 117(6):743–753, 2008.

Eray Erturk, Fahad Kamran, Salar Abbaspourazad, Sean Jewell, Harsh Sharma, Yujie Li, Sinead Williamson, Nicholas J Foti, and Joseph Futoma. Beyond sensor data: Foundation models of behavioral data from wearables improve health predictions. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=DtVVltU1ak.

Eibe Frank and Mark Hall. A simple approach to ordinal classification. In Luc De Raedt and Peter Flach, editors, *Machine Learning: ECML 2001*, pages 145–156, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg. ISBN 978-3-540-44795-5.

Evelynne S Fulda, Bennett J Waxse, Slavina B Goleva, Tam C Tran, Henry J Taylor, Caitlin P Bailey, Dana L Wolff-Hughes, Huan Mo, Chenjie Zeng, Jacob M Keaton, et al. 11 million days of longitudinal wearable data reveal novel future health insights. *medRxiv*, 2026.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? *Advances in neural information processing systems*, 35: 507–520, 2022.

Yutao Guo, Hao Wang, Hui Zhang, Tong Liu, Zhaoguang Liang, Yunlong Xia, Li Yan, Yunli Xing, Haili Shi, Shuyan Li, et al. Mobile photoplethysmographic technology to detect atrial fibrillation. *Journal of the American College of Cardiology*, 74(19):2365–2375, 2019.

Zelin He, Sarah Alnegheimish, and Matthew Reimherr. Harnessing vision-language models for time series anomaly detection, 2025. URL https://arxiv.org/abs/2506.06836.

Steven G Hershman, Brian M Bot, Anna Shcherbina, Megan Doerr, Yasbanoo Moayedi, Aleksandra Pavlovic, Daryl Waggott, Mildred K Cho, Mary E Rosenberger, William L Haskell, et al. Physical

activity, sleep and cardiovascular health data for 50,000 individuals from the myheart counts study. *Scientific data*, 6(1):24, 2019.

A Ali Heydari, Ken Gu, Vidya Srinivas, Hong Yu, Zhihan Zhang, Yuwei Zhang, Akshay Paruchuri, Qian He, Hamid Palangi, Nova Hammerquist, et al. The anatomy of a personal health agent. *arXiv preprint arXiv:2508.20148*, 2025.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

Aditya Jalin, Nawat Swatthong, Michelle Rozwadowski, Rajnish Kumar, Tom Braun, Noelle Carlozzi, David A Hanauer, Afton Hassett, Muneesh Tewari, and Sung Won Choi. A digital biomarker dataset from hematopoietic cell transplant caregivers and patients. *Scientific Data*, 2026.

Ali Javed, Daniel Seung Kim, Steven G Hershman, Anna Shcherbina, Anders Johnson, Alexander Tolas, Jack W O'Sullivan, Michael V McConnell, Laura Lazzeroni, Abby C King, et al. Personalized digital behaviour interventions increase short-term physical activity: a randomized control crossover trial substudy of the myheart counts cardiovascular health study. *European Heart Journal-Digital Health*, 4(5):411–419, 2023.

Matthew Jörke, Defne Genç, Valentin Teutschbein, Shardul Sapkota, Sarah Chung, Paul Schmiedmayer, Maria Ines Campero, Abby C. King, Emma Brunskill, and James A. Landay. Bloom: Designing for llm-augmented behavior change interactions. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems (CHI '26)*, New York, NY, USA, April 2026. ACM. doi: 10.1145/3772318.3790506. In Press, CHI 2026 Best Paper Award.

Justin Khasentino, Anastasiya Belyaeva, Xin Liu, Zhun Yang, Nicholas A Furlotte, Chace Lee, Erik Schenck, Yojan Patel, Jian Cui, Logan Douglas Schneider, et al. A personal health large language model for sleep and fitness coaching. *Nature Medicine*, 31(10):3394–3403, 2025.

Predrag Klasnja, Shawna Smith, Nicholas J Seewald, Andy Lee, Kelly Hall, Brook Luers, Eric B Hekler, and Susan A Murphy. Efficacy of contextually tailored suggestions for physical activity: a micro-randomized optimization trial of heartsteps. *Annals of Behavioral Medicine*, 53(6):573–582, 2019.

Patrick Langer, Thomas Kaar, Max Rosenblattl, Maxwell A Xu, Winnie Chow, Martin Maritsch, Robert Jakob, Ning Wang, Juncheng Liu, Aradhana Verma, et al. Opentslm: Time-series language models for reasoning over multivariate medical text-and time-series data. *arXiv preprint arXiv:2510.02410*, 2025.

Edmund WJ Lee, Huanyu Bao, Yongda S Wu, Man Ping Wang, Yi Jie Wong, and K Viswanath. Examining health apps and wearable use in improving physical and mental well-being across us, china, and singapore. *Scientific reports*, 14(1):10779, 2024.

Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, 18(185):1–52, 2018. URL https://jmlr.org/papers/v18/16-558.html.

Peng Liao, Kristjan Greenewald, Predrag Klasnja, and Susan Murphy. Personalized heartsteps: A reinforcement learning algorithm for optimizing physical activity. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, 4(1):1–22, 2020.

Shengsheng Lin, Weiwei Lin, Wentai Wu, Feiyu Zhao, Ruichao Mo, and Haotong Zhang. Segrnn: Segment recurrent neural network for long-term time series forecasting. *IEEE Internet of Things Journal*, 2025.

Qinghua Liu, Sam Heshmati, Zheda Mai, Zubin Abraham, John Paparrizos, and Liu Ren. Mllm4ts: Leveraging vision and multimodal language models for general time-series analysis, 2025. URL https://arxiv.org/abs/2510.07513.

Markus Löning, Anthony Bagnall, Sajaysurya Ganesh, Viktor Kazakov, Jason Lines, and Franz Király. sktime: A unified interface for machine learning with time series. 2019.

Steven A Lubitz, Anthony Z Faranesh, Steven J Atlas, David D McManus, Daniel E Singer, Sherry Pagoto, Alexandros Pantelopoulos, and Andrea S Foulkes. Rationale and design of a large population study to validate software for the assessment of atrial fibrillation from data acquired by a consumer tracker or smartwatch: the fitbit heart study. *American Heart Journal*, 238:16–26, 2021.

Steven A Lubitz, Anthony Z Faranesh, Caitlin Selvaggi, Steven J Atlas, David D McManus, Daniel E Singer, Sherry Pagoto, Michael V McConnell, Alexandros Pantelopoulos, and Andrea S Foulkes. Detection of atrial fibrillation in a large population using wearable devices: the fitbit heart study. *Circulation*, 146(19):1415–1424, 2022.

Aitian Ma, Dongsheng Luo, and Mo Sha. Mixlinear: Extreme low resource multivariate time series forecasting with 0.1 k parameters. *arXiv preprint arXiv:2410.02081*, 2024.

Igor Matias, Maximilian Haas, Eric J Daza, Matthias Kliegel, and Katarzyna Wac. Digital biomarkers for brain health: passive and continuous assessment from wearable sensors. *npj Digital Medicine*, 2026.

Michael V McConnell, Anna Shcherbina, Aleksandra Pavlovic, Julian R Homburger, Rachel L Goldfeder, Daryl Waggot, Mildred K Cho, Mary E Rosenberger, William L Haskell, Jonathan Myers, et al. Feasibility of obtaining measures of lifestyle from a smartphone app: the myheart counts cardiovascular health study. *JAMA cardiology*, 2(1):67–76, 2017.

Mike A Merrill, Esteban Safranchik, Arinbjörn Kolbeinsson, Piyusha Gade, Ernesto Ramirez, Ludwig Schmidt, Luca Foschini, and Tim Althoff. Homekit2020: A benchmark for time series classification on a large mobile sensing dataset with laboratory tested ground truth of influenza infections. In Bobak J. Mortazavi, Tasmie Sarker, Andrew Beam, and Joyce C. Ho, editors, *Proceedings of the Conference on Health, Inference, and Learning*, volume 209 of *Proceedings of Machine Learning Research*, pages 207–228. PMLR, 22 Jun–24 Jun 2023. URL https://proceedings.mlr.press/v209/merrill23b.html.

Mike A Merrill, Akshay Paruchuri, Naghmeh Rezaei, Geza Kovacs, Javier Perez, Yun Liu, Erik Schenck, Nova Hammerquist, Jake Sunshine, Shyam Tailor, et al. Transforming wearable data into personal health insights using large language model agents. *Nature Communications*, 2026.

Ahmed A Metwally, A Ali Heydari, Daniel McDuff, Alexandru Solot, Zeinab Esmaeilpour, Anthony Z Faranesh, Menglian Zhou, Girish Narayanswamy, Maxwell A Xu, Xin Liu, et al. Insulin resistance prediction from wearables and routine blood biomarkers. *Nature*, pages 1–11, 2026.

Shira Mitchell, Eric Potash, Solon Barocas, Alexander D'Amour, and Kristian Lum. Algorithmic fairness: Choices, assumptions, and definitions. *Annual Review of Statistics and Its Application*, 8:141–163, 2021.

Inbal Nahum-Shani, Shawna N Smith, Bonnie J Spring, Linda M Collins, Katie Witkiewitz, Ambuj Tewari, and Susan A Murphy. Just-in-time adaptive interventions (jitais) in mobile health: key components and design principles for ongoing health behavior support. *Annals of behavioral medicine*, pages 1–17, 2016.

Girish Narayanswamy, Xin Liu, Kumar Ayush, Yuzhe Yang, Xuhai Xu, shun liao, Jake Garrison, Shyam A. Tailor, Jacob Sunshine, Yun Liu, Tim Althoff, Shrikanth Narayanan, Pushmeet Kohli, Jiening Zhan, Mark Malhotra, Shwetak Patel, Samy Abdel-Ghaffar, and Daniel McDuff. Scaling wearable foundation models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=yb4QE6b22f.

Junghwan Park, Gregory J Norman, Predrag Klasnja, Daniel E Rivera, and Eric Hekler. Development and validation of multivariable prediction algorithms to estimate future walking behavior in adults: retrospective cohort study. *JMIR mHealth and uHealth*, 11(1):e44296, 2023.

Marco V Perez, Kenneth W Mahaffey, Haley Hedlin, John S Rumsfeld, Ariadna Garcia, Todd Ferris, Vidhya Balasubramanian, Andrea M Russo, Amol Rajmane, Lauren Cheung, et al. Large-scale assessment of a smartwatch to identify atrial fibrillation. *New England Journal of Medicine*, 381 (20):1909–1917, 2019.

Arvind Pillai, Dimitris Spathis, Fahim Kawsar, and Mohammad Malekzadeh. Papagei: Open foundation models for optical physiological signals. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=kYwTmlq6Vn.

Lukasz Piwek, David A Ellis, Sally Andrews, and Adam Joinson. The rise of consumer health wearables: promises and barriers. *PLoS medicine*, 13(2):e1001953, 2016.

Alvin Rajkomar, Michaela Hardt, Michael D Howell, Greg Corrado, and Marshall H Chin. Ensuring fairness in machine learning to advance health equity. *Annals of Internal Medicine*, 169(12):866–872, 2018.

Gabriela Retamales, Marino E. Gavidia, Ben Bausch, Arthur N. Montanari, Andreas Husch, and Jorge Goncalves. Towards automatic home-based sleep apnea estimation using deep learning. 7(1): 144. ISSN 2398-6352. doi: 10.1038/s41746-024-01139-z. URL https://doi.org/10.1038/s41746-024-01139-z.

Alessio Rossi, Eleonora Da Pozzo, Dario Menicagli, Chiara Tremolanti, Corrado Priami, Alina Sirbu, David Clifton, Claudia Martini, and David Morelli. Multilevel monitoring of activity and sleep in healthy people. *PhysioNet*, 2020.

Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Spreading vectors for similarity search, 2019. URL https://arxiv.org/abs/1806.03198.

Mithun Saha, Maxwell A Xu, Wanting Mao, Sameer Neupane, James M Rehg, and Santosh Kumar. Pulse-ppg: An open-source field-trained ppg foundation model for wearable applications across lab and field settings. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(3):1–35, 2025.

Paul Schmiedmayer, Anders Johnson, Narayan Schuetz, Lukas Kollmer, Paul Goldschmidt, Juan Delgado-SanMartin, Kelly W Zhang, Sriya D. Mantena, Alexander Tolas, Samuel Montalvo, Mariana Ramirez-Posada, Jack W. O'Sullivan, Marily Oppezzo, Abby C King, Fatima Rodriguez, Euan Ashley, Allan Lawrie, and Daniel Seung Kim. Design and rationale of the my heart counts cardiovascular health study: a large-scale, fully digital biobank, and randomized trial of large language model-driven coaching of physical activity. *American Journal of Preventive Cardiology*, page 101565, 2026.

ISSN 2666-6677. doi: 10.1016/j.ajpc.2026.101565. URL https://www.sciencedirect.com/science/article/pii/S2666667726001595.

Oleksandr Shchur et al. fev-bench: A realistic benchmark for time series forecasting. *arXiv preprint arXiv:2509.26468*, 2025.

Jinjoo Shim, Elgar Fleisch, and Filipe Barata. Circadian rhythm analysis using wearable-based accelerometry as a digital biomarker of aging and healthspan. *NPJ Digital Medicine*, 7(1):146, 2024.

Ravid Shwartz-Ziv and Amitai Armon. Tabular data: Deep learning is not all you need. *Information fusion*, 81:84–90, 2022.

Rujul Singh, Macy K Tetrick, James L Fisher, Peter Washington, Jane Yu, Electra D Paskett, Frank J Penedo, Steven K Clinton, and Roberto M Benzo. Analysis of physical activity using wearable health technology in us adults enrolled in the all of us research program: multiyear observational study. *Journal of medical Internet research*, 26:e65095, 2024.

Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical bayesian optimization of machine learning algorithms. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'12, page 2951–2959, Red Hook, NY, USA, 2012. Curran Associates Inc.

Chang Wei Tan, Angus Dempster, Christoph Bergmeir, and Geoffrey I. Webb. MultiRocket: multiple pooling operators and transformations for fast and effective time series classification. 36(5):1623–1646, 2022. ISSN 1573-756X. doi: 10.1007/s10618-022-00844-1. URL https://doi.org/10.1007/s10618-022-00844-1.

Mingtian Tan, Mike A Merrill, Vinayak Gupta, Tim Althoff, and Thomas Hartvigsen. Are language models actually useful for time series forecasting? *Advances in Neural Information Processing Systems*, 37:60162–60191, 2024.

Adedolapo Aishat Toye, Asuman Celik, and Samantha Kleinberg. Benchmarking missing data imputation methods for time series using real-world test cases. *Proceedings of machine learning research*, 287:480, 2025.

James Truslow, Angela Spillane, Huiming Lin, Katherine Cyr, Adeeti Ullal, Edith Arnold, Ron Huang, Laura Rhodes, Jennifer Block, Jamie Stark, et al. Understanding activity and physiology at scale: the apple heart & movement study. *npj Digital Medicine*, 7(1):242, 2024.

U.S. Food and Drug Administration. Clinical decision support software: Guidance for industry and food and drug administration staff. https://www.fda.gov/regulatory-information/search-fda-guidance-documents/clinical-decision-support-software, January 2026a. Final Guidance, issued January 6, 2026 (re-issued January 29, 2026); supersedes the September 28, 2022 version. Docket FDA-2017-D-6569.

U.S. Food and Drug Administration. General wellness: Policy for low risk devices. https://www.fda.gov/regulatory-information/search-fda-guidance-documents/general-wellness-policy-low-risk-devices, January 2026b. Revised Final Guidance, reissued January 6, 2026; supersedes the September 27, 2019 version.

Olivia J. Walch, Yitong Huang, Daniel B. Forger, and Cathy A Goldstein. Sleep stage prediction with raw acceleration and photoplethysmography heart rate data derived from a consumer wearable device. *Sleep*, 42, 2019. URL https://api.semanticscholar.org/CorpusID:203653773.

Andrea Weber, Vincent T van Hees, Michael J Stein, Sylvia Gastell, Karen Steindorf, Florian Herbolsheimer, Stefan Ostrzinski, Tobias Pischon, Mirko Brandes, Lilian Krist, et al. Large-scale assessment of physical activity in a population using high-resolution hip-worn accelerometry: the german national cohort (nako). *Scientific Reports*, 14(1):7927, 2024.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. TimesNet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

Maxwell A Xu, Jaya Narain, Gregory Darnell, Haraldur T Hallgrimsson, Hyewon Jeong, Darren Forde, Richard Andres Fineman, Karthik Jayaraman Raghuram, James Matthew Rehg, and Shirley You Ren. Relcon: Relative contrastive learning for a motion foundation model for wearable data. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=k2uUeLCrQq.

Maxwell A Xu, Girish Narayanswamy, Kumar Ayush, Dimitris Spathis, Shun Liao, Shyam A Tailor, Ahmed Metwally, A Ali Heydari, Yuwei Zhang, Jake Garrison, et al. Lsm-2: Learning from incomplete wearable sensor data. *arXiv preprint arXiv:2506.05321*, 2025b.

Xuhai Xu, Xin Liu, Han Zhang, Weichen Wang, Subigya Nepal, Yasaman Sefidgar, Woosuk Seo, Kevin S Kuehn, Jeremy F Huckins, Margaret E Morris, et al. Globem: Cross-dataset generalization of longitudinal human behavior modeling. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 6(4):1–34, 2023.

Hang Yuan, Shing Chan, Andrew P Creagh, Catherine Tong, Aidan Acquah, David A Clifton, and Aiden Doherty. Self-supervised learning for human activity recognition using 700,000 person-days of wearable data. *NPJ digital medicine*, 7(1):91, 2024.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 11121–11128, 2023. doi: 10.1609/aaai.v37i9.26317.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International Conference on Machine Learning*, pages 27268–27286. PMLR, 2022.

Contents

A. Research Implementation and Oversight

A.1. Limitations

A major limitation of our work is that despite the size and diversity, the demographics of our dataset and benchmark reflect the study populations that choose to use such digital health apps, and are thus skewed towards white, male, based around US metropolitan areas, and in their late thirties, which may limit generalizability to the broader public. While fairness-adjusted evaluations partially mitigate this, they cannot fully address it. Moreover, since this dataset comes from a fully digital study, all variables are self-reported, introducing a degree of label noise, which is likely at least part of the reason why we could not improve performance on some downstream targets beyond what a linear baseline achieved. The dataset reflects real-world conditions and evolved over time, which is a strength and limitation; the data are not perfect, and despite our efforts to clean them and include quality indicators such as coverage metrics, substantial data are missing for some users, and not all variables and channels are available uniformly. It should be noted that our sensitivity analyses (Appendix D.4), indicate that, e.g., data quality filtering introduces no meaningful bias. Finally, the models presented in our benchmark tasks are best-effort implementations that may not always be optimal due to resource constraints and should be viewed as a reference point, not a reflection of what will ultimately be possible.

On a related note, OpenMHC does not support adjudicated incident-event prediction, such as one-year incident cardiovascular disease (see Apple Heart Study [Perez et al., 2019] and Fitbit Heart Study [Lubitz et al., 2022]), two-year incident diabetes (see WEAR-ME [Metwally et al., 2026]), or six-month depression-symptom worsening, because the underlying study lacks linked clinical-event adjudication, future-glucose endpoints, and validated longitudinal mental-health instruments. Tasks here are therefore detection or characterization at the time of survey rather than prospective risk forecasting; Appendix A.6 discusses the regulatory tier framing for each outcome group.

A.2. Compute Resources

Experiments were run on a mix of academic HPC clusters (Imperial College London, Stanford) and commercial cloud infrastructure, using a heterogeneous mix of NVIDIA GPUs (including A100, H100, L40S, and L4). Single-run evaluation wall-clock ranges from minutes for statistical baselines and linear probes on precomputed features, to up to roughly two days for slower neural method in imputation and forecasting evaluation. Cumulative GPU wall-clock for all reported experiments on academic clusters is approximately 400 GPU-hours, with foundation-model fine-tuning for the forecasting track contributing up to a thousand further GPU-hours. The large language model evaluation track was performed via Google's Gemini API and consumed no local GPU compute; the at-list-price equivalent on Vertex AI is approximately $3,400 (actual cost zero through a courtesy research allocation). Hyperparameter sweeps and exploratory or preliminary experiments that did not contribute to reported results required approximately 2,200 additional GPU-hours on academic clusters.

A.3. Ethical Oversight and IRB Documentation

This research was conducted in accordance with the Declaration of Helsinki. Ethical approval was granted and renewed in September 2025 by the Institutional Review Board (IRB) at Stanford University under Protocol ID: **#31409**, (My Heart Counts: Stanford Mobile Cardiovascular Health Study).

A.4. Data Collection Application

The data for this study were collected via a custom-built mobile application called My Heart Counts (https://myheartcounts.stanford.edu/) built on Apple ResearchKit and released as one of the first flagship apps in collaboration with Apple. The application was designed to ensure user consent, data integrity, and user privacy. Informed consent was obtained digitally from all participants prior to data collection. Participants may withdraw at any point during the study without justification. Participants also have granular control on what individual metrics they are comfortable sharing (e.g., share step count but not heart rate) [McConnell et al., 2017].

A.5. App Screenshots

Below you can find the actual consent screens as they appeared in the app.

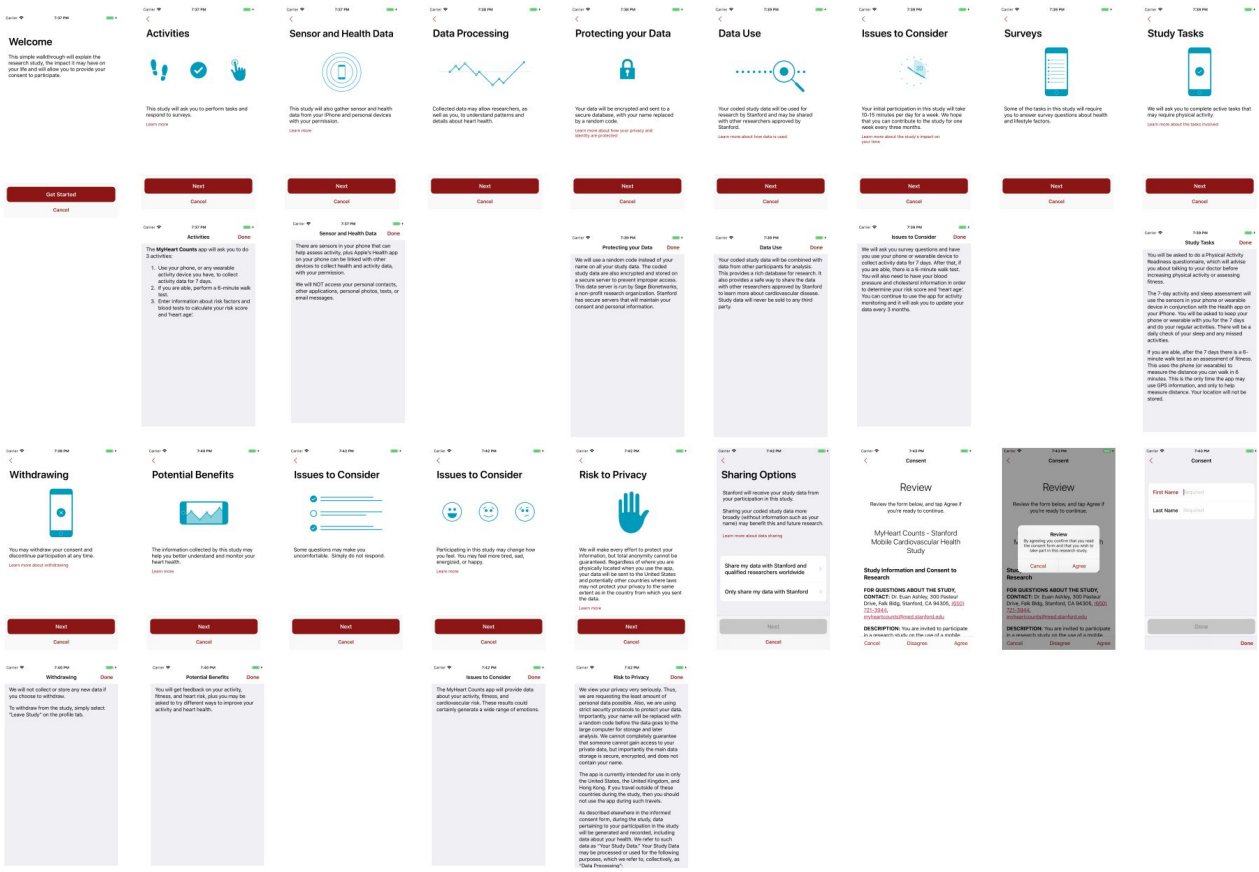

Figure 3 | Exemplar screenshots of consent screens as they appear in the US / HK version of the app.

A.6. Regulatory Context

OpenMHC's task definitions and reported outputs are designed to align with the FDA's January 2026 reissue of the Clinical Decision Support (CDS) and General Wellness guidances [U.S. Food and Drug Administration, 2026a,b], which clarified that episodic wearable summaries typically fall outside device-tier regulation, extended enforcement discretion to single-recommendation CDS, and broadened the General Wellness envelope for non-invasive physiologic summaries (e.g., blood pressure, glucose) reported as ranges, trends, or category bands. We operationalize this through three choices:

outputs are summarized as skill-scored aggregates and category bands rather than patient-facing probabilities; tasks are framed as detection or characterization at the time of survey, not prospective forecasting of clinical events; and we publish per-task fairness-adjusted skill scores so subgroup performance is transparent (Appendix B.2). Regulatory classification is ultimately a determination by the relevant authority and not one we assert.

What OpenMHC does *not* do. OpenMHC does not support adjudicated incident-event prediction. Specifically, these include:

- **One-year incident cardiovascular disease** (see Apple Heart Study [Perez et al., 2019], Fitbit Heart Study [Lubitz et al., 2022], sedentary-time associations [Ajufo et al., 2025]) requires linked EHR or insurance-claim adjudication of MACE outcomes that the MHC study does not collect.
- **Two-year incident diabetes** (see WEAR-ME [Metwally et al., 2026]) requires future glucose endpoints, which MHC does not link.
- **Six-month depression-symptom worsening** would require validated longitudinal mental-health instruments (e.g., PHQ-9 repeats); MHC's mental-well-being tasks are single-time-point Likert items.

Only 13.1% of MHC participants have ≥ 1 year of data (Figure 8; max 10.7 yr), which further limits the cohort even for tasks that could be reframed as longitudinal.

Outcome tier framing. Most OpenMHC outcomes fall under the General Wellness envelope: demographics, sleep behavior, subjective well-being, Apple-Watch digital biomarkers (resting heart rate, VO_2max, HRV-SDNN, etc.), and lifestyle indicators are reported as continuous values or ordinal bands. Cardiometabolic biomarkers (HDL, LDL, blood pressure) sit at a wellness/device-tier boundary depending on output framing, which the benchmark sidesteps by reporting only aggregate skill scores rather than user-facing predictions. Self-reported chronic-disease history (e.g., Diabetes, Hypertension, AFib) is retrospective and is not framed as a diagnosis. The Sleep Diagnosis label's status (self-reported clinical history vs. derived screening flag) is pending verification against the original MHC survey schema; we default to a conservative clinician-gated framing.

B. Evaluation Metrics

Evaluating models across diverse tasks in wearable and mobile health research presents a significant challenge due to the varied nature of the data and prediction targets. For instance, forecasting and imputation tasks span multiple sensor channels (e.g., heart rate, step count) with vastly different units, scales, and variances. Similarly, downstream prediction tasks encompass binary classification, ordinal classification, and regression, each evaluated using different metrics (e.g., AUROC, PRAUC, Pearson R). Averaging raw metrics across these tasks is mathematically unsound.

To address this, we adopt a unified evaluation methodology inspired by large-scale time series benchmarks [Hyndman and Athanasopoulos, 2018, Shchur et al., 2025], utilizing a **Skill Score** and an **Average Rank** metric. These approaches aggregate performance reliably across heterogeneous tasks.

B.1. Skill Score

Recall from (1) that the skill score quantifies the average relative improvement of a model over a fixed reference or baseline model. We now formally define the clipping function:

$$\text{clip}\left(\frac{E_{r,j}}{E_{r,b}}, \ell, u\right) = \min\left(\max\left(\frac{E_{r,j}}{E_{r,b}}, \ell\right), u\right)$$

Specifically, we use lower clipping value $\ell = 0.01$ and an upper clipping value of $u = 100$.

We adapt the definition of the "error" term E depending on the domain:

- **Forecasting and Imputation:** For tasks predicting continuous sensor values, E is a standard error metric such as Mean Absolute Error (MAE) or Mean Squared Error (MSE). The baseline β is a simple heuristic, such as a Seasonal Naive forecaster (for forecasting) or LOCF (for imputation).
- **Prediction Tasks:** For our prediction tasks we are using metrics that better models maximize: AUPRC for binary outcomes, Spearman's ρ for ordinal outcomes, and Pearson's r for continuous outcomes. Each has a maximum attainable value of 1. To apply the skill score, we convert each metric into an error-transformed score measuring distance from the optimum:

  $$E = 1 - \text{Metric}. \tag{2}$$

  For example, an ROC AUC of 0.85 corresponds to an error-transformed score of 0.15. For health outcome prediction, the baseline β is the LINEAR model.

Uncertainty quantification. We report uncertainty using 95% bootstrap confidence intervals (CIs). The reported value for every metric is its point estimate, computed on the original held-out test split. CIs are estimated by participant-level bootstrap resampling of the test split with 1,000 replicates: participants are sampled with replacement using a single shared, seeded draw matrix per split, and each metric (skill score, average rank, fairness skill score) is recomputed on every replicate. For the skill score and average rank we report the percentile interval—the 2.5th and 97.5th percentiles of the bootstrap distribution; the fairness skill score uses the bias-corrected and accelerated (BCa) interval anchored at the point estimate (Appendix B.2).

B.2. Fairness Skill Score

To evaluate the performance of models across demographic subgroups, we benchmark performance across two sensitive attributes:

- **age_group**: 18-29, 30-39, 40-49, 50-59, 60+, unknown.
- **sex**: male, female, unknown.

The `unknown` bucket preserves participants with missing demographics rather than dropping them, ensuring the union of subgroup test sets structurally equals the global test set. We limit the sensitive categories to these two variables because the available sample size is significantly lower for variables like ethnicity, which many users opted not to disclose.

Let $\mathcal{G}$ denote a specific sensitive attribute (e.g., $\mathcal{G}$ = sex), and let $g \in \mathcal{G}$ represent a specific mutually exclusive subgroup within that attribute (e.g., g = female). We define the raw performance disparity of model j for attribute $\mathcal{G}$ using an average over all distinct pairs $g, g' \in \mathcal{G}$:

$$D_j^{(\mathcal{G})} = \frac{1}{|\mathcal{G}|(|\mathcal{G}|-1)} \sum_{g,g' \in \mathcal{G}, g \neq g'} \left| E_j^{(g)} - E_j^{(g')} \right|$$

where $E_j^{(g)}$ is the error metric (as defined in Appendix B.1) achieved by model j on subgroup g. We define the *Fairness Skill Score* ($S_{\text{fair}}^{(\mathcal{G})}$) for a given sensitive attribute as the relative reduction in the demographic performance gap over the baseline model b:

$$S_{\text{fair}}^{(\mathcal{G})} = 1 - \text{GeometricMean}\left(\text{clip}\left(\frac{D_j^{(\mathcal{G})}}{D_b^{(\mathcal{G})}}, \ell, u \right) \right) \tag{3}$$

We apply the identical clipping boundaries ($\ell = 0.01$, $u = 100$) detailed in Appendix B.1. Under this formulation, a positive score ($S_{\text{fair}}^{(\mathcal{G})} > 0$) implies the model successfully contracted the baseline's disparity gap, a score of 0 indicates parity with the baseline's inequity, and negative scores flag models that actively exacerbated group disparities.

To aggregate this property across our multi-task benchmark, the global fairness score for a given model execution is defined as the macro-average across all evaluated sensitive attribute dimensions:

$$S_{\text{fair}} = \frac{1}{|\mathbf{A}|} \sum_{\mathcal{G} \in \mathbf{A}} S_{\text{fair}}^{(\mathcal{G})} \tag{4}$$

where $\mathbf{A} = \{\text{age\_group}, \text{sex}\}$.

Uncertainty quantification. We report bias-corrected 95% bootstrap confidence interval for S_{fair}. Let S_{fair} denote the observed fairness skill score and $S^*_{\text{fair},1}, \ldots, S^*_{\text{fair},B}$ denote the B bootstrap samples of the fairness skill score (where we sample with replacement participants). Specifically, the bias-correction we use re-centers the confidence interval by adjusting the percentiles of the empirical distribution of $S^*_{\text{fair},1}, \ldots, S^*_{\text{fair},B}$ that we use (e.g., instead of always using the 0.025 and 0.975 percentiles of the empirical distribution for a 95% confidence interval, we may use 0.01 and 0.94).

Following DiCiccio and Efron [1996], we compute

$$z_0 = \Phi^{-1}\left(\frac{1}{B} \sum_{b=1}^{B} \mathbf{1}\left\{ S^*_{\text{fair},b} < S_{\text{fair}} \right\} \right),$$

where Φ is the CDF of a standard Gaussian distribution. We also compute the acceleration term using leave-one-participant-out jackknife recomputes. Let $S_{\text{fair},(i)}$ denote the fairness score recomputed after leaving out participant (i), let $\bar{S}_{\text{fair},(\cdot)} = \frac{1}{n} \sum_{i=1}^{n} S_{\text{fair},(i)}$, and define $d_i := \bar{S}_{\text{fair},(\cdot)} - S_{\text{fair},(i)}$. The acceleration term is computed as follows:

$$A := \frac{\sum_{i=1}^{n} d_i^3}{6 \left(\sum_{i=1}^{n} d_i^2 \right)^{3/2}}.$$

This term adjusts the interval for asymmetry in the sampling distribution of the score. For a nominal $1 - \alpha$ interval, the adjusted percentile levels are

$$\alpha_{\text{lo}} = \Phi\left(z_0 + \frac{z_0 + z_{\alpha/2}}{1 - A\left(z_0 + z_{\alpha/2}\right)}\right), \qquad \alpha_{\text{hi}} = \Phi\left(z_0 + \frac{z_0 + z_{1-\alpha/2}}{1 - A\left(z_0 + z_{1-\alpha/2}\right)}\right),$$

where $z_q = \Phi^{-1}(q)$. Our confidence interval reports the α_{lo} and α_{hi} quantiles of the empirical distribution of $S^*_{\text{fair},1}, \ldots, S^*_{\text{fair},B}$.

B.3. Average Rank

While the skill score measures the magnitude of improvement, the **Average Rank** measures consistency across tasks. For each task r, all evaluated models are ranked from 1 (best) to N (worst) based on their respective metrics. The average rank for model j is the arithmetic mean of its ranks across all tasks. The average rank is entirely invariant to the scale of the metric and robust to outliers, providing a reliable secondary measure to confirm that a model's high skill score is due to consistent performance across the board rather than massive gains on a small subset of tasks.

Uncertainty quantification. We report a 95% confidence interval constructed using same bootstrap approach used for the Skill Score (Appendix B.1).

C. Detailed Dataset Characteristics

This appendix provides additional descriptive statistics for the OpenMHC participant cohort, linked variables, device coverage, geography, wearable-channel coverage, and official dataset splits. Data preprocessing is described separately in Appendix D.

The OpenMHC dataset is built from 11,894 sharable participants in the My Heart Counts study. The wearable data include four channel groups: phone-derived activity, watch-derived activity and physiology, sleep, and workouts. Figure 4 describes participant flow for the wearable cohort, Figure 5 summarizes linked variable coverage and participant-level profiles, and Figure 7 visualizes the released geographic distribution. Finally, Table 7 details participant demographics across dataset splits. Continuous variables are reported as mean ± SD and median [IQR]; categorical variables as count (% of covered participants).

Linked variables. The labels API (`src/labels/`) exposes 169 per-participant variables in total, split between 7 longitudinal HealthKit-derived metrics extracted from raw Apple Watch records and 162 self-reported survey variables collected through the MHC app's questionnaires. Appendix H provides a detailed enumeration of all variables.

Table 6 reports participant-level observation coverage for each wearable channel. Coverage is highest for phone-derived activity channels: step count and walking/running distance are observed for 11,631 and 11,630 participants, respectively. Watch-derived activity and physiology channels are observed for approximately 7,000 participants, while sleep channels are observed for fewer than 2,800 participants. Workout channels have the sparsest coverage, ranging from 176 participants for Mixed Metabolic Cardio to 2,607 participants for Walking. These differences reflect device ownership and logging behavior rather than task-specific inclusion criteria.

Device coverage. Table 5 summarises the iPhone and Apple Watch generations present in the sharable cohort. Each user is counted once per generation family they ever owned, with within-generation variants (Plus, Pro, Pro Max, mini, case size, GPS versus GPS+Cellular) folded into the parent row, since sensor hardware is largely shared within a generation. The column sums therefore exceed the number of unique users, because a user who upgrades across generations contributes to multiple rows. The *Unknown* rows reflect a HealthKit metadata artifact: when a user renames their device, the original model identifier is overwritten, so the generation is not recovered from metadata alone. Across the cohort, 18 % of phone records and 40 % of watch records carry no clear model identifier. Share of devices over time is visualized in Figure 6. Figure 9 displays samples across low, medium, and high-coverage.

Geographic aggregation. We summarize participant geography using the released context variables `field_country` and `field_zip`. Country-level assignment prioritizes `field_country`; when it is missing, we infer the country bucket from the anonymized ZIP token. Numeric 1–3 digit ZIP prefixes are treated as US, while UK and Hong Kong postcodes are released only as the coarse `UK` and `HK` tokens. For US zip-codes we remove the zip code of participants from regions with populations of less than 20k for privacy reasons. For the US state map, we map each released numeric ZIP prefix to candidate states using a ZIP-to-state crosswalk. Prefixes that span multiple states are split fractionally according to the number of full ZIP codes in each state, yielding an estimated state-level distribution rather than exact residence locations. The resulting country and US state distributions are shown in Figure 7.

(a) iPhone families

Family	Users
iPhone 4	1
iPhone 5 / 5s	906
iPhone 6 / 6s (+ Plus)	4,855
iPhone 7 (+ Plus)	872
iPhone 8 (+ Plus)	400
iPhone X / XR / XS / XS Max	1,234
iPhone SE (1st–3rd gen)	172
iPhone 11 (+ Pro / Pro Max)	682
iPhone 12 (+ mini / Pro / Pro Max)	428
iPhone 13 (+ mini / Pro / Pro Max)	346
iPhone 14 (+ Plus / Pro / Pro Max)	207
iPhone 15 (+ Plus / Pro / Pro Max)	130
iPhone 16 (+ Pro / Pro Max)	79
iPhone Unknown	6,499
Any phone (unique users)	11,642

(b) Apple Watch families

Family	Users
Apple Watch (1st gen)	304
Series 1	61
Series 2	224
Series 3	474
Series 4	710
Series 5	382
Series 6	357
Series 7	176
Series 8	93
Series 9	53
Series 10	23
SE (1st gen)	56
SE (2nd gen)	15
Ultra	52
Ultra 2	42
Apple Watch Unknown	6,485
Any watch (unique users)	7,338

Table 5 | Device generations represented in the sharable cohort (n = 11,894). Each user is counted once per generation family they ever owned. Within-generation variants (Plus, Pro, Pro Max, mini, case size, GPS versus GPS+Cellular) are folded into the parent row. *Unknown* entries arise when a user renames their device in iOS Settings, which overwrites the model identifier exposed via HealthKit and the MHC app did not store the device version specifically otherwise. Assuming that device naming is likely unrelated to device type, the distribution of known names should, however, be somewhat representative of the total distribution.

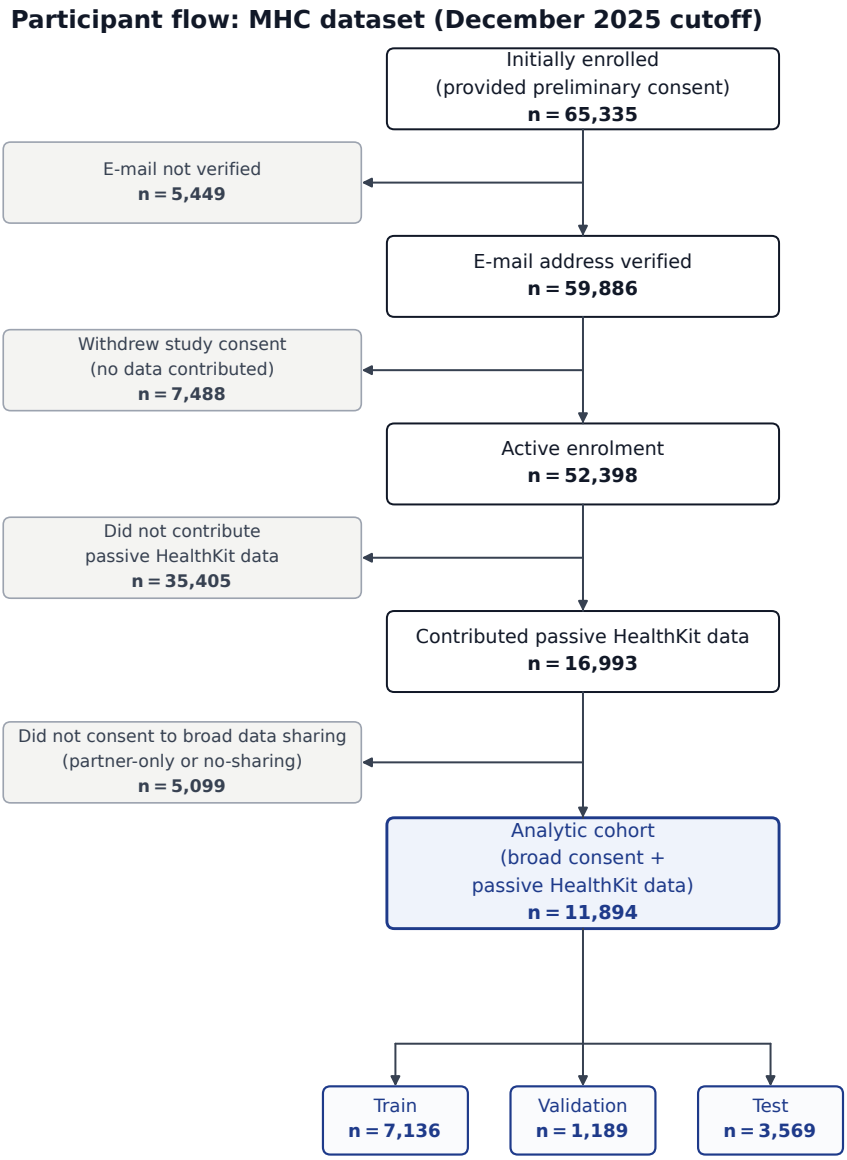

Figure 4 | Participant flow diagram for the MHC wearable benchmark cohort. Counts n denote the number of unique participants at each cohort-construction step. The figure reports cohort construction and wearable-data availability. Task-specific downstream prediction participant counts are reported separately in Table 13.

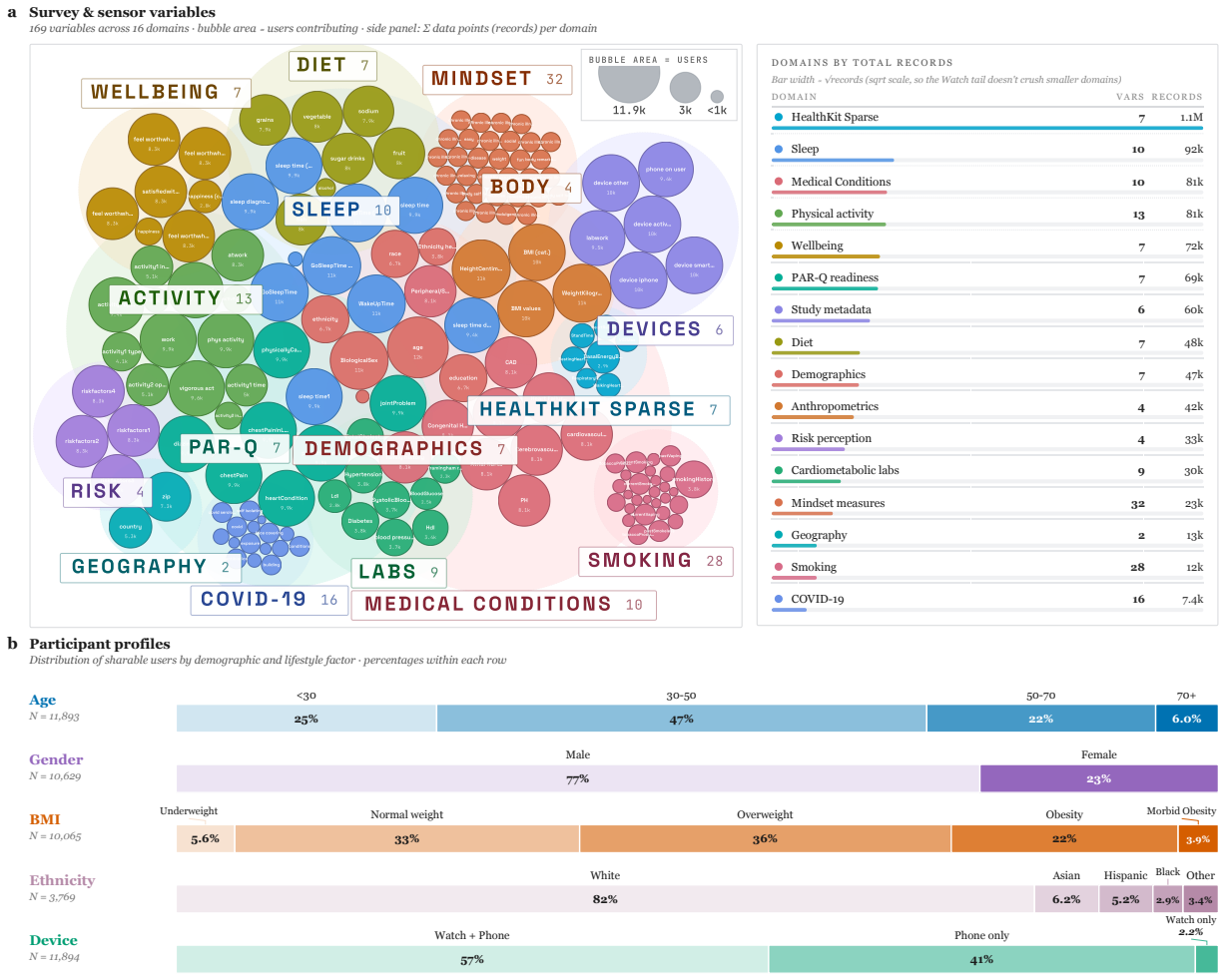

Figure 5 | **Linked variables and participant profiles. (a) Linked variables:** coverage of 169 linked variables across 16 source domains. Each circle denotes one variable, and circle area is proportional to the number of participants with at least one observation for that variable. **(b) Participant profiles:** distributions of age (N = 11,893), biological sex (N = 10,629), BMI (N = 10,065), ethnicity (N = 3,769), and device availability (N = 11,894).

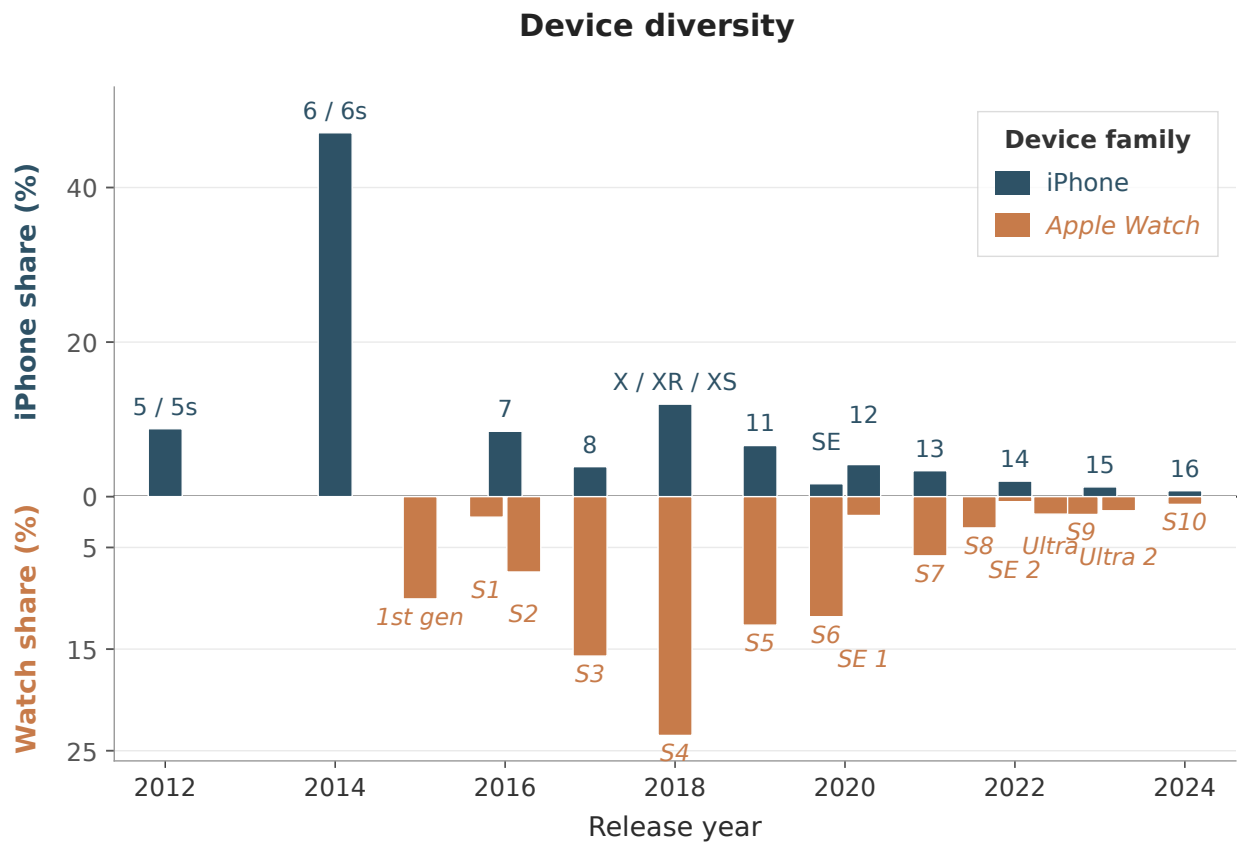

Figure 6 | **Device generation distribution in the sharable cohort**. Bars show each generation's share of known device entries within its category (iPhone top, Watch bottom; bars sum to 100% per panel). Users are counted once per generation owned. Unknown entries and iPhone 4 (n = 1) are excluded; SE generations are aggregated.

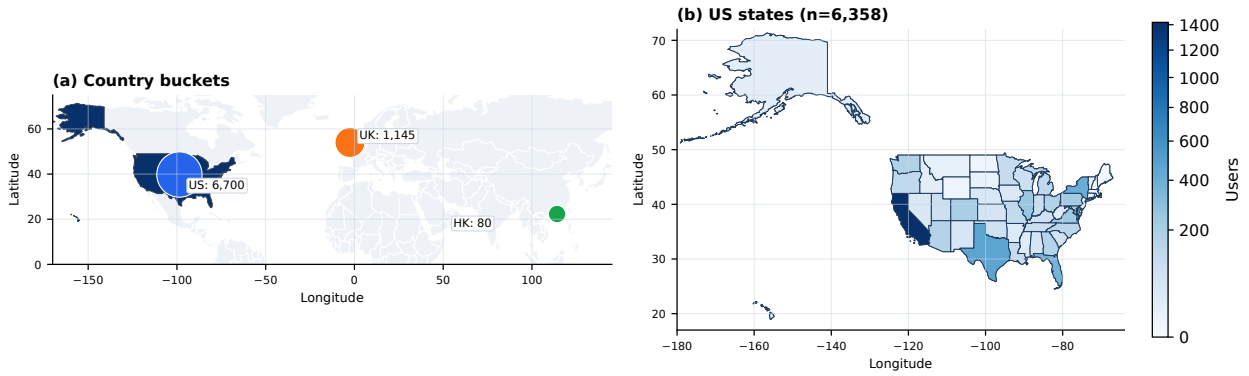

Figure 7 | **Released geographic distribution. (a) Country buckets:** participant counts resolved to US, UK, and HK from the geography context labels. **(b) US states:** estimated state-level distribution among the 6,372 participants with numeric US ZIP-prefix labels; 6,358 are allocated to states and 14 have prefixes not represented in the crosswalk.

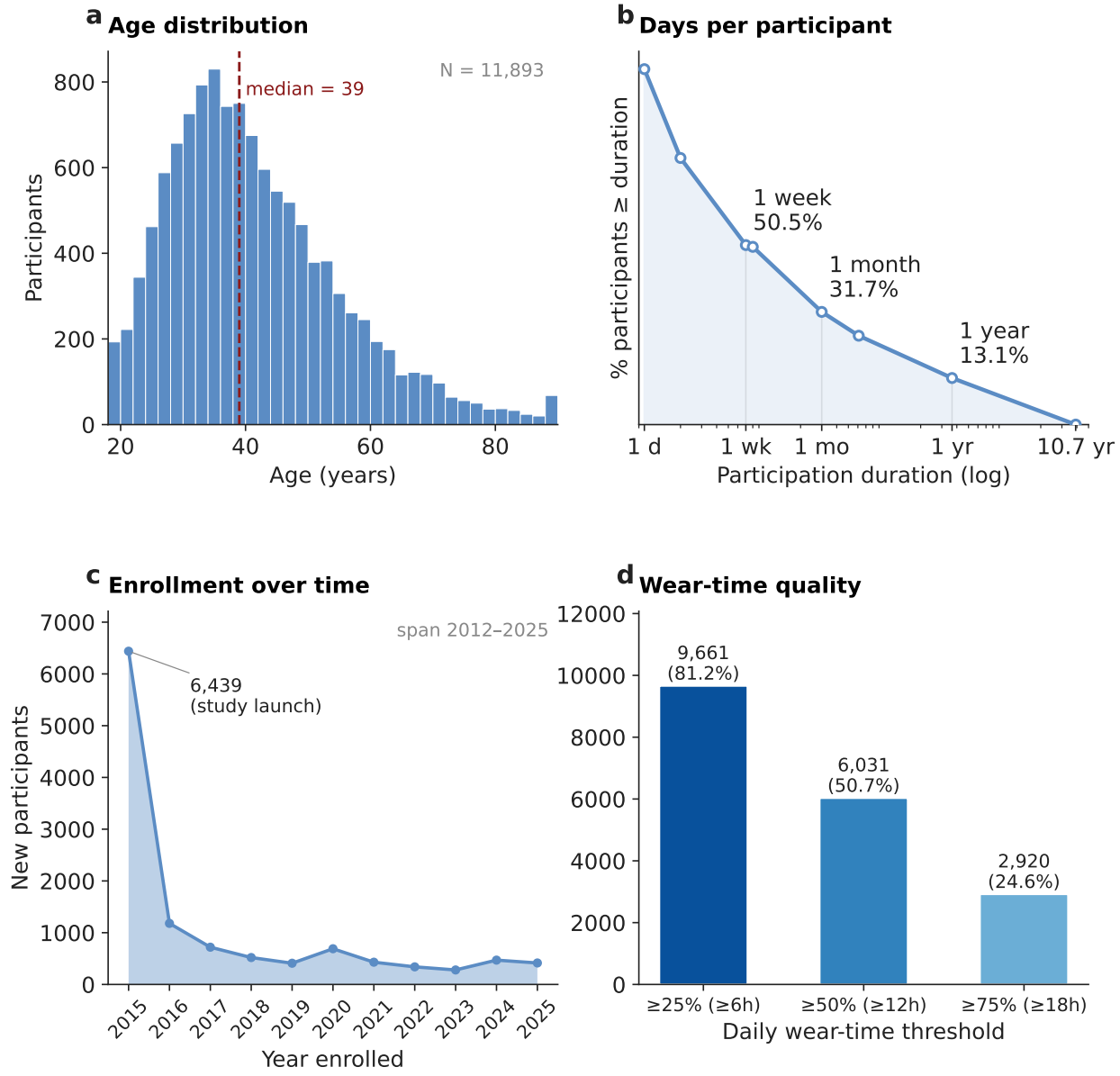

Figure 8 | **Overview of the My Heart Counts (MHC) dataset distributions. (a)** Age distribution of all reporting participants (N = 11,893, median 39, IQR [30, 52]). **(b)** Survival curve of participation duration on a log x-axis; 50.5% of participants contributed ≥ 1 week, 31.7% ≥ 1 month and 13.1% ≥ 1 year (max 10.7 yr). **(c)** New participants by year of enrollment: the MHC study launched in 2015 and drew 6,439 of its 11,894 lifetime participants in that first year alone, with a long tail of continued recruitment through 2025. **(d)** Wear-time quality: number of participants with at least one day meeting each daily wear-time threshold (≥ 6 h, ≥ 12 h, ≥ 18 h).

Table 6 | Summary of the 19 channels in the daily matrix $d \in \mathbb{R}^{19\times1440}$, computed over all 11,894 sharable participants prior to any quality filtering. **n**: number of participants with ≥1 day of data for that channel. **Avg. Days**: mean number of days with data per participant. **Span**: calendar-day range from first to last observation across all participants.

Category	Channel	Unit	n	Avg. Days	Span
Phone	StepCount	steps/min	11,631	234.4	4,730
	DistanceWalkingRunning	meter/min	11,630	234.6	4,730
	FlightsClimbed	count/min	2,752	430.6	4,730
Watch	StepCount	steps/min	7,010	202.2	4,307
	DistanceWalkingRunning	meter/min	7,006	200.6	4,307
	HeartRate	count/s	7,020	202.0	4,307
	ActiveEnergyBurned	cal/min	6,993	201.4	4,307
Sleep	Asleep	binary	2,704	254.3	4,730
	InBed	binary	2,784	361.7	4,730
Workout	Walking	binary	2,607	90.3	3,909
	Cycling	binary	1,325	52.9	3,912
	Running	binary	1,544	39.5	3,911
	Other	binary	1,164	43.4	3,871
	Mixed Metabolic Cardio	binary	176	20.5	3,240
	Strength Training	binary	557	52.9	3,827
	Elliptical	binary	677	35.1	3,868
	HIIT	binary	379	43.1	3,579
	Functional Strength	binary	493	36.7	3,858
	Yoga	binary	498	32.1	3,833

Table 7 | Participant demographics by dataset split. Continuous variables: mean ± SD, median [P25–P75]. Categorical variables: n (% of covered). Coverage: fraction of split with a non-null value.

Characteristic	**Train** (n=7,136)	**Validation** (n=1,189)	**Test** (n=3,569)	**Overall** (N=11,894)
Age (years)				
Coverage	7,136 (100%)	1,189 (100%)	3,568 (100%)	11,893 (100%)
Mean ± SD	42.2 ± 15.5	41.1 ± 14.7	41.6 ± 15.4	41.9 ± 15.4
Median [IQR]	39 [30–53]	39 [30–50]	39 [30–52]	39 [30–52]
Biological Sex				
Coverage	6,395 (89.6%)	1,064 (89.5%)	3,170 (88.8%)	10,629 (89.4%)
Male	4,945 (77.3%)	813 (76.4%)	2,439 (76.9%)	8,197 (77.1%)
Female	1,450 (22.7%)	251 (23.6%)	731 (23.1%)	2,432 (22.9%)
Height (cm)				
Coverage	6,447 (90.3%)	1,072 (90.2%)	3,192 (89.4%)	10,711 (90.1%)
Mean ± SD	168.8 ± 34.9	169.6 ± 33.0	167.7 ± 37.2	168.6 ± 35.5
Median [IQR]	175.3 [167.6–182.9]	175.3 [167.6–182.9]	175.3 [167.6–182.9]	175.3 [167.6–182.9]
Weight (kg)				
Coverage	6,447 (90.3%)	1,072 (90.2%)	3,192 (89.4%)	10,711 (90.1%)
Mean ± SD	81.3 ± 24.6	82.0 ± 23.2	81.2 ± 24.4	81.3 ± 24.4
Median [IQR]	79.8 [68.0–93.9]	81.2 [69.4–93.4]	80.3 [68.0–93.6]	80.3 [68.0–93.9]
BMI (kg/m^2)				
Coverage	6,063 (85.0%)	1,017 (85.5%)	2,985 (83.6%)	10,065 (84.6%)
Mean ± SD	27.3 ± 6.5	27.4 ± 6.1	27.4 ± 6.2	27.3 ± 6.4
Median [IQR]	26.1 [23.3–30.1]	26.2 [23.6–30.1]	26.2 [23.3–30.0]	26.1 [23.3–30.1]
Ethnicity (self-reported)				
Coverage	2,277 (31.9%)	375 (31.5%)	1,117 (31.3%)	3,769 (31.7%)
White	1,872 (82.2%)	294 (78.4%)	938 (84.0%)	3,104 (82.4%)
Asian	138 (6.1%)	31 (8.3%)	64 (5.7%)	233 (6.2%)
Hispanic	119 (5.2%)	23 (6.1%)	53 (4.7%)	195 (5.2%)
Black	69 (3.0%)	11 (2.9%)	29 (2.6%)	109 (2.9%)
Other	53 (2.3%)	11 (2.9%)	19 (1.7%)	83 (2.2%)
Prefer not to say	11 (0.5%)	2 (0.5%)	6 (0.5%)	19 (0.5%)
American Indian	8 (0.4%)	2 (0.5%)	6 (0.5%)	16 (0.4%)
Pacific Islander	6 (0.3%)	1 (0.3%)	2 (0.2%)	9 (0.2%)
Alaska Native	1 (0.0%)	0 (0.0%)	0 (0.0%)	1 (0.0%)

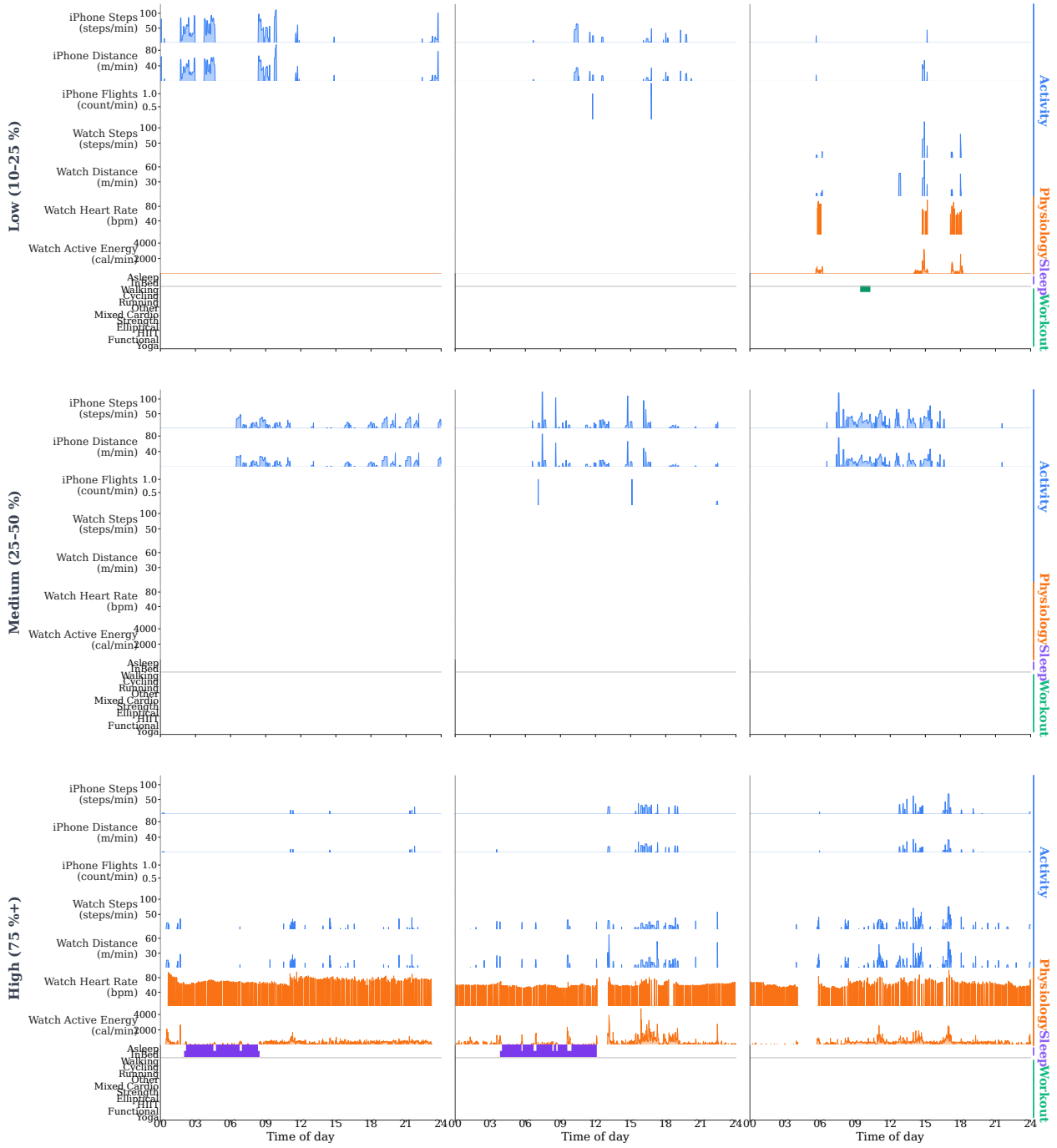

Figure 9 | **Daily wearable samples at different wear-time coverage levels.** Nine randomly selected daily samples from the released dataset, arranged in a 3×3 grid by wear-time coverage tier: low (10–25 %), medium (25–50 %), and high (≥75 %). Each cell shows all 19 channels at minute resolution (1,440 time steps): seven continuous channels (phone and watch activity, heart rate, active energy) as sparkline traces and twelve binary channels (sleep, workouts) as heatmap lanes. Gaps in the traces correspond to periods with no recorded data. The figure illustrates the heterogeneity of data completeness across real-world participants.

D. Data Preprocessing

This appendix describes how raw My Heart Counts app collected HealthKit records are turned into activity, physiology, sleep, and workout records, which are released and used throughout the benchmark. We first define the daily minute-level matrix construction pipeline, then describe the additional filtering and aggregation steps used to construct daily, hourly, and weekly model inputs.

D.1. Basic Data Cleaning: Construction of Daily Matrices

This section describes how raw wearable data from the MHC study are converted into daily minute-level matrices. The raw data are heterogeneous: participants use different devices and iOS versions, records arrive with free-text source identifiers rather than structured device metadata, and measurement coverage varies substantially across participants and time periods as the app evolved together with HealthKit and included more and more HealthKit types as time went on. Initial database files are stored as CSV records on a daily level, leading to TBs of raw data. In a first step, we extract and transform these raw files with evolving schemas, remove corrupted entries, and de-duplicate rows to create an initial data lake version of the raw HealthKit data in parquet format (for full transparency, this process is documented here: https://github.com/NarayanSchuetz/myheart-counts-client).

We transform per-participant Parquet exports of HealthKit, sleep, and workout records into per-participant HDF5 files, each containing daily matrices $d \in \mathbb{R}^{19\times1440}$ (19 channels at minute resolution). The pipeline proceeds through device type inference, dominant source selection, record-level cleaning, temporal alignment, data stream availability detection, daily matrix generation, and hourly aggregation.

Device Type Inference. From the raw HealthKit data, it is often unclear what type of device each individual has (e.g., iPhone, Apple Watch, Garmin, etc.). Therefore, we first assign each raw source to a device class: iPhone, Apple Watch, or ambiguous. In the raw HealthKit records, each source is represented by a free-text source name (e.g., "iPhone 14 Pro", "Apple Watch Series 8") rather than structured device type identifiers. We apply the staged heuristic in Algorithm 1 to classify each record's source. Let $\mathcal{T}_{\text{watch}} = \{\text{StepCount}, \text{DistanceWalkingRunning}, \text{ActiveEnergyBurned}, \text{HeartRate}\}$ and $\mathcal{T}_{\text{phone}} = \{\text{StepCount}, \text{DistanceWalkingRunning}, \text{FlightsClimbed}\}$ denote the expected type signatures for each device class. Ambiguous devices are filtered out as we found their data to be too inconsistent, while likely iPhone and likely Apple Watch sources are propagated to their respective Apple device class. This decision was made because manual inspection showed that the vast majority are most likely Apple devices, and retaining them maximizes data availability.

Dominant Source Selection. Another issue we encountered is that some participants have multiple devices of the same type contributing data simultaneously, for instance, when upgrading to a new Apple Watch while the previous device continues to sync. To avoid double-counting, we select a single *dominant source* per device type (iPhone, Apple Watch) per calendar day. A source is considered valid for a given day only if it reported all required HealthKit types for its device class: StepCount and DistanceWalkingRunning for iPhones; those two plus HeartRate and ActiveEnergyBurned for Apple Watches. Among valid sources, the one with the highest record count is selected as dominant; ties result in no dominant source for that device type on that day. Only records from the dominant iPhone and dominant Apple Watch sources are retained for later matrix construction.

Record-level cleaning. Before the temporal aggregation, we remove records with inconsistent timestamps, invalid durations, or implausible measurement values. Three universal checks are applied to all record types: records with mismatched timezone offsets between start and end times (indicating timezone transitions during recording, creating potential artifacts), records with negative

Algorithm 1: Device Type Inference

Input: Source name n, bundle identifier b, set of reported HealthKit types $\mathcal{T}$
Output: Device type label $\ell \in$ {iPhone, AppleWatch, ambiguous}

```
// Stage 1: Name-based hints
if n contains "phone" (case-insensitive) then
    ℓ ← likely_iPhone
else if n contains "watch" (case-insensitive) then
    ℓ ← likely_AppleWatch
else
    ℓ ← unassigned

// Stage 2: Apple identifier confirmation
if b starts with com.apple then
    if ℓ = likely_iPhone then ℓ ← iPhone;
    if ℓ = likely_AppleWatch then ℓ ← AppleWatch;

// Stage 3: Type signature classification (unresolved sources only)
if ℓ ∉ {iPhone, AppleWatch} then
    if T_watch ⊆ T then
        ℓ ← likely_AppleWatch
    else if T_phone ⊆ T and {ActiveEnergyBurned, HeartRate} ∩ T = ∅ then
        ℓ ← likely_iPhone
    // Repeat Apple identifier confirmation for newly labeled sources
    if b starts with com.apple then
        if ℓ = likely_iPhone then ℓ ← iPhone;
        if ℓ = likely_AppleWatch then ℓ ← AppleWatch;

// Stage 4: Fallback
if ℓ ∉ {iPhone, AppleWatch, likely_iPhone, likely_AppleWatch} then ℓ ← ambiguous;
return ℓ
```

duration, and records exceeding 24 hours are removed (which we found to mostly be anomalies). We additionally apply type-specific rate and range thresholds to remove physiologically implausible values. Active energy values are converted from kilocalories to calories where applicable. Specific threshold settings are defined in Table 8.

Temporal Alignment (Midnight Splitting). To provide daily matrices, records spanning midnight are split at the day boundary so that each day receives its respective part of the record. For count-based quantities (steps, distance, energy, flights climbed, stand time), the value is allocated proportionally to the fraction of the record's duration falling on each side of midnight. For rate quantities (heart rate) the value is preserved unchanged on both sides. Sleep and workout intervals are similarly split at midnight with adjusted durations. Note, Apple HealthKit data is recorded in the user's local time, and while it logs timezones, those are unreliable and only valid for live records since the recorded timezone is based on the timezone at the time the data is extracted from HealthKit; to avoid confusion, we do not adjust timezones and simply provide data in the user's local time.

Daily Matrix Generation. Since HealthKit data is recorded as intervals and values for many types, and we require minute-level data, records were first spread across an 86,400-element second-level array (one value per second), then resampled to 1,440 minute-level bins: count-based channels (steps, distance, energy, flights, stand time) are summed per minute, while heart rate is averaged.

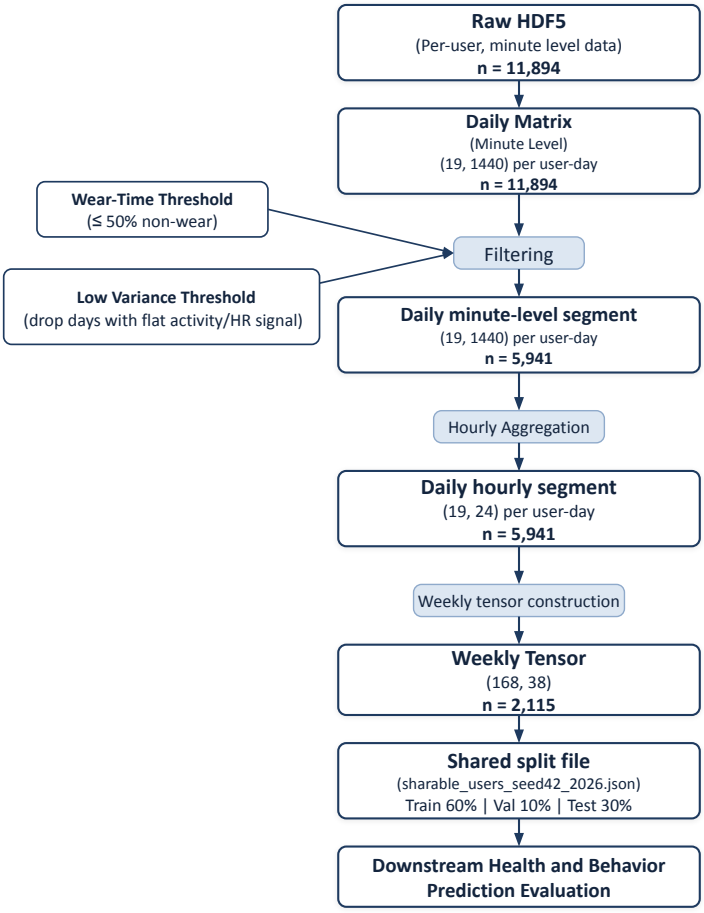

Figure 10 | Data preprocessing flow for the downstream health and behavior prediction evaluation. The initial participant filtering process is shown in Figure 4, and daily matrix construction is described in Section D.1.

Table 8 | Per-type quality filter thresholds. Records exceeding the maximum rate or falling outside the valid range are removed.

HealthKit Type	Filter	Threshold
StepCount	Max rate	5 steps/s
DistanceWalkingRunning	Max rate	10 m/s
ActiveEnergyBurned	Max rate	250 cal/s
DistanceCycling	Max rate	25 m/s
AppleStandTime	Max rate	1 min/min
HeartRate	Valid range	40–200 bpm

Table 9 | Value semantics in the daily matrix.

Channel Group	Presence Criterion	Zero	NaN
Continuous HK (0–6)	≥10% of days	No activity	Stream unavailable
Sleep (7–8)	Any occurrence	Not asleep	No sleep data
Workouts (9–18)	Any occurrence	Not exercising	Never logged

Overlapping records at the same second bin are resolved via `nanmean`. Sleep and workout intervals produce binary minute-level indicator arrays (1 during the interval, 0 otherwise; overlapping intervals use OR semantics). The 7 continuous, 2 sleep, and 10 workout channels are stacked to form the daily matrix $d \in \mathbb{R}^{19\times1440}$.

Data Stream Detection and Non-Wear Time Heuristic. Not all participants have the same wearable channels: many lack an Apple Watch entirely, and workout logging varies widely. To distinguish *true inactivity* (zero) from *structural absence* (NaN), we determine per-participant data stream availability before generating daily matrices. For each participant and continuous HealthKit channel, we compute the fraction of the participant's observed calendar days on which that channel reports any data. If this fraction is below 10%, the channel is treated as unavailable for that participant and filled with NaN on all days. Otherwise, the channel is treated as available, and missing values on individual days are represented through the standard NaN/mask convention. For sleep and workout channels, any occurrence across the participant's history is sufficient to mark the stream as available. Table 9 summarizes the resulting value semantics.

We try to estimate non-wear periods by identifying consecutive time intervals where all 7 continuous HealthKit channels (indices 0–6) simultaneously record zero or NaN. Consecutive runs exceeding 30 minutes are flagged as non-wear. Sleep and workout channels (indices 7–18) are excluded from this detection, as they represent behavioral signals rather than passive sensor readings (for instance sleep can be non-wear). The resulting binary non-wear vector is used to compute per-day wear-time statistics.

Per-Channel Missing Data Detection. Beyond general non-wear periods that affect all channels, we independently for each of the 19 channels, identify consecutive runs of ≥120 minutes of zero or NaN values as missing intervals. These per-channel missing intervals are stored as metadata alongside each daily matrix in a HDF5 file, providing a compact summary of long gaps in each data stream.

Output Format. The pipeline produces one gzip-compressed HDF5 file per participant, with one dataset per calendar day. Each dataset stores the 19×1440 daily matrix together with metadata attributes including the non-wear vector, per-channel missing intervals, and total wear-time minutes.

D.2. Self-Reported Variable Sanitation

Self-reported outcomes were cleaned to ensure physiological validity. For instance, to address implausible outliers in height and weight, specific thresholds were implemented based on clinical guidelines and physiological limits. Table 10 summarizes the inclusion/filtering criteria and processing steps applied to the self-reported variables exposed through the labels API in the code base.

Table 10 | Assumptions and filtering of final self-reported outcome labels

Metrics	Variable Names	Considerations
Blood Pressure	SystolicBloodPressure, DiastolicBloodPressure	Inverted if Systolic BP was lower than Diastolic BP
BMI	bmi	Height 1.4–2.1m and weight ≥ 40kg
Wake & Sleep time	wake_time, sleep_time	Local time registered by device converted to hours
Psychological	feel_worthwhile, satisfied_life, happiness	Thresholds based on distributions
Happiness (long)	happiness (static)	Filtered to ≥ 3 instances per patient
Activity	vigorous_activity, physical_activity	Range [0, ∞); Categories based on AHA recommendations

D.3. Wearable Data Preprocessing for the Benchmark

After constructing the daily wearable matrices, we apply additional preprocessing to define the wearable data used by the models we train for the benchmark. Note that other dataset preprocessing approaches could be used at this step. To standardize the evaluation, we apply these filtering steps and require users who want to submit their models to our leaderboard to evaluate on these samples (it is up to the submitter whether they use the same filtering for their training dataset).

D.3.1. Data Filtering

Wear-Time Filtering. For all evaluation tasks, we retain only calendar days whose non-wear vector contains at most 720 non-wear minutes, equivalently to at least 12 hours of estimated wear time ($\leq$ 50% non-wear). The 12-hour cutoff was chosen to balance day-level signal quality, cohort size, and cohort representativeness; Appendix D.4 reports a sensitivity analysis supporting this choice.

Low-Variance Filtering. As a second day-level removal criterion applied to the retained days, we remove days with near-constant sensor traces while being above zero (likely sensor artifacts/glitches) by requiring minimum within-day variance on the monitored continuous channels. The thresholds are 1.0 for iPhone steps, iPhone distance, Watch steps, Watch distance, and Watch active energy, and 10^{-4} for Watch heart rate; the flights-climbed channel is excluded. Channels with undefined variance because of insufficient observed values are not removed by this rule. We refer to days that pass both the wear-time and low-variance filters as *retained days*.

Representation-level NaN handling. On the days that survive the wear-time and low-variance filters, we apply a value-level zero-to-NaN transform that re-labels missingness without removing any further days. It reinterprets values that are semantically implausible or unreliable as missing:

heart-rate values equal to zero are set to NaN; for the step, distance, and active-energy channels, the entire day of that channel is set to NaN if it is all zero throughout the day; and for the two sleep channels, zero-valued minutes are set to NaN when the total detected sleep on that day is less than 3 hours. Flights climbed and workout channels are left unchanged. These filtering and re-labeling steps are applied when constructing the hourly and weekly benchmark representations and related feature-extraction pipelines.

Weekly Data Filtering. We define a *retained week* as a candidate 7-day window containing at least five retained days. Candidate windows with fewer than five retained days are discarded and are not used for weekly-tensor model inputs.

D.3.2. Data Aggregation

Daily minute-level segment. A daily minute-level segment corresponds to one filtered participant-day at minute resolution, represented as

$$X_{i,d}^{\text{min}} \in \mathbb{R}^{19\times 1440}, \tag{5}$$

where i denotes the participant and d the calendar day. This segment preserves within-day minute resolution and is used by methods that operate directly on minute-level daily records.

Hourly Aggregation. To transform a daily matrix from minute-level to hourly level, it is aggregated from minute resolution to one value per hour. Count-like continuous channels are summed (NaNs removed), whereas heart rate is averaged (NaNs removed). Sleep and workout channels are binary and use OR semantics: an hourly value is 1 if any minute in that hour is 1, and 0 if all observed minutes are 0. If all minute-level values for a channel within an hour are NaN, the hourly entry is treated as missing and is stored as a zero-filled value together with a binary indicator of missingness.

Daily hourly segment. The hourly segment for a retained day is stored as a pair of arrays

$$V_{i,d}^{\text{hr}} \in \mathbb{R}^{19\times 24}, \qquad M_{i,d}^{\text{hr}} \in \{0,1\}^{19\times 24}, \tag{6}$$

where $V_{i,d}^{\text{hr}}$ contains hourly aggregated values and $M_{i,d}^{\text{hr}}$ is the corresponding binary missingness indicator. For methods that require a time-major segment, the hourly values and missingness indicators are transposed to shape $(24, 19)$ and concatenated to form a $(24, 38)$ tensor.

Weekly Tensors. A weekly tensor corresponds to a 7-day sequence of hourly aggregated wearable data. We use rolling 7-day windows to find each available week window. For each participant, we generate candidate 7-day windows starting from the participant's first available wearable date and advance the window one calendar day at a time. Each candidate window is represented as

$$X_{i,s}^{\text{week}} \in \mathbb{R}^{168\times 38}, \tag{7}$$

where s indexes the candidate 7-day window and the 168 time steps correspond to 7 consecutive days at hourly resolution. Each hourly step contains 19 aggregated wearable values and 19 binary missingness indicators.

D.4. Sensitivity Analysis of Wear-Time Filtering

Two design choices control which participant days and which label instances enter evaluation: the daily wear-time threshold that defines a *retained day* (Appendix D.3.1) and the category-specific eligibility window that determines when a label instance has sufficient temporally proximate wearable data (Appendix E.1.2). We ablate both choices on the full cohort used throughout the article (11,894 participants) and find that the benchmark defaults do not produce a large or systematic demographic

shift relative to the full population, despite a few statistically significant differences at individual thresholds.

Daily Wear-Time Filter. We tested four wear-time thresholds: a day is retained if its estimated wear time is at least 18, 12, 6, or 0 hours per day, equivalently at most 25%, 50%, 75%, or 100% non-wear. The 12-hour threshold (≤ 50% non-wear) is the benchmark default (Appendix D.3.1). The XGBoost pipeline (Appendix E.2.2) was re-run end-to-end at each setting with bootstrap evaluation (n=1,000 resamples). Table 11 summarizes the resulting cohort demographics.

Table 11 | Cohort size and demographic composition across non-wear thresholds. Values marked * are statistically significantly different ($p < 0.05$) from the unfiltered baseline (KS test for continuous variables, χ^2 test for categorical variables).

	≤ 25% ≥ 18 h wear	≤ 50% ≥ 12 h wear	≤ 75% ≥ 6 h wear	No filter (all days)
Participants Retained	2,920	6,031	9,661	11,894
Age (mean ± SD)	46.1 ± 15.4*	44.8 ± 14.9*	44.6 ± 15.2	44.4 ± 15.1
BMI (mean ± SD)	27.8 ± 6.5	27.6 ± 6.1*	27.4 ± 6.4	27.3 ± 6.4
Hypertension %	29.4*	26.8	26.2	25.9
Diabetes %	7.2	6.0	5.8	5.8
CVD %	14.1*	11.4	10.9	10.9
Female %	75.5	78.9*	78.2	77.1

E. Prediction Tasks

This appendix section specifies the prediction tasks and their evaluation protocol. The benchmark contains 32 static prediction tasks constructed from MHC source variables, including survey fields, HealthKit measurements, and derived quantities.

E.1. Prediction Task Definitions

E.1.1. Outcome Types and Construction

Table 12 | 32 prediction tasks from self-reported health outcomes and HealthKit measurements.

Outcome	Self-report question/source	Type	Construction details
Demographics (2 outcomes)			
Age	`enrollment_info.json` birthdate and `last_labels.json` last-survey timestamp	Continuous	Whole years from birthdate to last-survey timestamp; participants with computed age outside [0, 100] are dropped.
Biological Sex	"Gender", recorded by Apple's HealthKit as one of `Female` / `Male` / `Other` (HealthKit field`HKBiologicalSex`)	Binary	Male is encoded as 1 and Female as 0; "Other" option absent in observed data.
Medical Conditions & Risk (12 outcomes)			
Atrial Fibrillation	`heart_disease`: "Have you been diagnosed with any of the below diseases?"	Binary	1 if the participant selected "Atrial fibrillation (Afib)".
Cardiovascular Disease	`heart_disease`: "Have you been diagnosed with any of the below diseases?"; `vascular`: "Which vascular disease diagnosis have you received?"	Binary	1 if the participant selected any non-"None of the above" option in either multi-select question.
Cerebrovascular Disease	`vascular`: "Which vascular disease diagnosis have you received?"	Binary	1 if the participant selected Stroke, Transient Ischemic Attack, Carotid Artery Blockage/Stenosis, or Carotid Artery Surgery/Stent.
Congenital Heart Disease	`heart_disease`: "Have you been diagnosed with any of the below diseases?"	Binary	1 if the participant selected "Congenital Heart Defect".
Coronary Artery Disease	`heart_disease`: "Have you been diagnosed with any of the below diseases?"	Binary	1 if the participant selected any of Heart Attack/Myocardial Infarction, Heart Bypass Surgery, Coronary Blockage/Stenosis, Coronary Stent/Angioplasty, Angina, or High Coronary Calcium Score.
Diabetes	"Do you have Diabetes?"	Binary	Yes/no response.

Outcome	Self-report question/source	Type	Construction details
Framingham CVD Risk	Age, Biological Sex, Total Cholesterol, HDL Cholesterol, Systolic Blood Pressure, hypertension-treatment status, smoking, and Diabetes	Continuous, [0, 1]	D'Agostino Sr et al. [2008] ASCVD 10-year risk; values are clipped to [0, 1]; cohort restricted to age 30–79, with participants with prior CVD or diabetes excluded.
Heart Failure / CHF	`heart_disease`: "Have you been diagnosed with any of the below diseases?"	Binary	1 if the participant selected "Heart Failure or CHF".
Hypertension	"Are you being treated for Hypertension (High Blood Pressure)?"	Binary	Yes/no response; denotes treatment status, not measured blood-pressure elevation.
Pulmonary Hypertension	`heart_disease`: "Have you been diagnosed with any of the below diseases?"; `vascular`: "Which vascular disease diagnosis have you received?"	Binary	1 if the participant selected "Pulmonary Hypertension" in `heart_disease` or "Pulmonary Arterial Hypertension" in `vascular`.
Sleep Disorder Diagnosis	"Have you ever been told by a doctor or other health professional that you have a sleep disorder?"	Binary	Yes/no response.
Vascular Disease	`vascular`: "Which vascular disease diagnosis have you received?"	Binary	1 if the participant selected Peripheral Vascular Disease or Abdominal Aortic Aneurysm.
Vitals & Blood Biomarkers (8 outcomes)			
BMI Categories	BMI Value	Ordinal (K=5)	BMI binned using cut points 19.9, 24.9, 29.9, and 39.9 into Underweight, Normal weight, Overweight, Obesity, and Morbid Obesity.
BMI Value	HealthKit `BodyMass` and HealthKit `Height`	Continuous	Computed as $\text{weight}_{\text{kg}}/\text{height}^2_{\text{m}}$.
Blood Pressure Categories	Systolic Blood Pressure and Diastolic Blood Pressure	Ordinal (K=4)	Compound rule: Normal if SBP< 120 and DBP< 80; Elevated if SBP 120–129 and DBP< 80; Hypertension Stage 1 if SBP 130–139 or DBP 80–89; Hypertension Stage 2 if SBP≥ 140 or DBP≥ 90. Swapped if DBP > SBP
Body Weight	HealthKit `BodyMass` passive measurement	Continuous	Numeric value in kilograms.
HDL Cholesterol	"HDL Cholesterol"	Continuous	Numeric response, range 10–140; units are locale-dependent.
LDL Cholesterol	"LDL Cholesterol"	Continuous	Numeric response, range 0–1000; units are locale-dependent.
Systolic Blood Pressure	"Systolic Blood Pressure"	Continuous	Numeric response, range 90–200 mmHg.

Outcome	Self-report question/source	Type	Construction details
Total Cholesterol	"Total Cholesterol"	Continuous	Numeric response, range 80–400; units are locale-dependent.
Mental Well-Being (5 outcomes)			
Feel Depressed	"How about depressed?"	Ordinal (K=4)	0–10 Likert response binned using cut points 1, 3, and 5 into Low, Medium, High, and Very High; higher classes correspond to higher reported distress.
Feel Happy	"How about happy?"	Ordinal (K=4)	0–10 Likert response binned using cut points 4, 6, and 8 into Low, Medium, High, and Very High.
Feel Worried	"How about worried?"	Ordinal (K=4)	0–10 Likert response binned using cut points 4, 6, and 8 into Low, Medium, High, and Very High.
Life Satisfaction	"Overall, how satisfied are you with life as a whole these days?"	Ordinal (K=4)	0–10 Likert response binned using cut points 4, 6, and 8 into Low, Medium, High, and Very High.
Things Are Worthwhile	"Overall, to what extent do you feel the things you do in your life are worthwhile?"	Ordinal (K=4)	0–10 Likert response binned using cut points 4, 6, and 8 into Low, Medium, High, and Very High.
Sleep & Lifestyle (5 outcomes)			
Bedtime	AppCore profile `GoSleepTime`, represented as 24-hour decimal time	Ordinal	Binned using cut points 1, 7, 19, 21, and 23; Late Sleeper appears on both sides of midnight.
Currently Employed	"Do you do regular work?"	Binary	Yes/no response.
Sleep Duration	"How much sleep do you think you need every night to be rested?"	Ordinal (K=4)	Continuous hours binned using cut points 6, 7, and 9 into Insufficient, Short, Normal, and Too Long.
Vigorous Activity Minutes	"Overall, how many minutes of vigorous activity do you get in a week?"	Continuous	Numeric slider, 0–2000 minutes/week.
Wake-up Time	AppCore profile `WakeUpTime`, represented as 24-hour decimal time	Ordinal (K=4)	Binned using cut points 5, 7, and 9 into Early Riser, Normal Riser, Late Riser, and Very Late Riser.

E.1.2. Inclusion Criteria

In order to ensure that participants have enough wearable data for us to consider doing prediction, we define a formal inclusion criteria that determines this. For each prediction task τ and label date t associated with a participant's non-missing label for that task, we define a task-specific window

$$W_\tau(t) = [t - \Delta_\tau^-,\ t + \Delta_\tau^+], \tag{8}$$

where $\Delta_\tau^-, \Delta_\tau^+ \geq 0$ are the task-specific offsets specified in days. Table 13 reports the window used for each task. A retained day is a calendar day that satisfies both the wear-time and the low-variance filters defined in the Appendix D.3.1. A participant is included if (i) the participant's label is not missing and (ii) the task-specific window contains at least one retained day.

Table 13 | **Overview of the 32 prediction tasks.** Each row corresponds to one prediction task defined from a health or behavior outcome, with outcomes grouped by domain. For each task, we report the task-specific offsets $\Delta^-\tau$ and $\Delta^+\tau$, in days, that define the inclusion window W_τ. The number of participants meeting the inclusion criteria (Appendix E.1.2) out of the total number of participants with at least one retained day after data filtering (see Figure 10). Label summaries: mean ± SD for continuous (regression) outcomes, positive-class prevalence for binary outcomes, and the category distribution for ordinal outcomes.

Outcome	Source	Outcome Type	Included Participants	Label Summary
Demographics				
Age ($\Delta_\tau^- = \Delta_\tau^+$ =1095 days)	Enrollment	Continuous	5,767 (97.1%)	43.0 ± 15.1 years
Biological Sex	HealthKit	Binary	5,603 (94.3%)	Male: 79.0%
Medical Conditions & Risk ($\Delta_\tau^- = \Delta_\tau^+$ =365 days)				
Atrial Fibrillation	Survey	Binary	3,661 (61.6%)	Prevalence 2.0%
Cardiovascular Disease	Survey	Binary	3,661 (61.6%)	Prevalence 11.2%
Cerebrovascular Disease	Survey	Binary	3,661 (61.6%)	Prevalence 1.7%
Congenital Heart Disease	Survey	Binary	3,661 (61.6%)	Prevalence 1.0%
Coronary Artery Disease	Survey	Binary	3,661 (61.6%)	Prevalence 4.2%
Diabetes	Survey	Binary	1,750 (29.5%)	Prevalence 6.4%
Framingham CVD Risk	Derived	Continuous	952 (16.0%)	0.1 ± 0.1 (10-yr prob.)
Heart Failure / CHF	Survey	Binary	3,661 (61.6%)	Prevalence 0.7%
Hypertension	Survey	Binary	1,750 (29.5%)	Prevalence 27.0%
Pulmonary Hypertension	Survey	Binary	3,661 (61.6%)	Prevalence 0.9%
Sleep Disorder Diagnosis	Survey	Binary	4,364 (73.5%)	Prevalence 15.7%
Vascular Disease	Survey	Binary	3,661 (61.6%)	Prevalence 0.7%
Vitals & Blood Biomarkers ($\Delta_\tau^- = \Delta_\tau^+$ =91 days)				
BMI Categories	Derived	Ordinal	4,477 (75.4%)	K=5
BMI Value	Derived	Continuous	4,477 (75.4%)	27.6 ± 6.2 kg/m^2
Blood Pressure Categories	Survey	Ordinal	1,507 (25.4%)	K=4
Body Weight	HealthKit	Continuous	4,687 (78.9%)	83.3 ± 23.5 kg
HDL Cholesterol	Survey	Continuous	1,371 (23.1%)	2.9 ± 0.9 mmol/L
LDL Cholesterol	Survey	Continuous	1,151 (19.4%)	5.6 ± 1.9 mmol/L
Systolic Blood Pressure	Survey	Continuous	1,507 (25.4%)	120.7 ± 13.3 mmHg
Total Cholesterol	Survey	Continuous	1,456 (24.5%)	9.4 ± 2.5 mmol/L
Mental Well-being ($\Delta_\tau^- = \Delta_\tau^+$ =14 days)				
Feel Depressed	Survey	Ordinal	2,005 (33.7%)	K=4
Feel Happy	Survey	Ordinal	2,829 (47.6%)	K=4
Feel Worried	Survey	Ordinal	2,660 (44.8%)	K=4
Life Satisfaction	Survey	Ordinal	2,822 (47.5%)	K=4
Things Are Worthwhile	Survey	Ordinal	2,822 (47.5%)	K=4
Sleep & Lifestyle ($\Delta_\tau^- = \Delta_\tau^+$ =91 days)				
Bedtime	Profile	Ordinal	4,568 (76.9%)	K=5
Currently Employed	Survey	Binary	3,909 (65.8%)	Prevalence 80.9%

Continued on next page

Table 13 continued

Outcome	Source	Outcome Type	Included	Label Summary
Sleep Duration	Survey	Ordinal	3,903 (65.7%)	K=4
Vigorous Activity Minutes	Survey	Continuous	3,815 (64.2%)	72.1 ± 124.8 min/week
Wake-up Time	Profile	Ordinal	4,546 (76.5%)	K=4

Enrollment = participant enrollment metadata. Profile = participant-entered app profile field. Derived = computed from one or more source variables.

E.2. Prediction Task Modeling

Evaluation. We use the same 60/10/30 train/validation/test split across participants across all tasks. For each task, predictors are fit on the training split and selected using validation performance only. The test split is held out for final reporting and is not used for model or hyperparameter selection.

Model input context. For each prediction task, models are evaluated on participants that meet the Inclusion Criteria (defined in Appendix (E.1.2). For each included participant, the models are allowed to use the complete history of the wearable data as input.

Dimensionality reduction. When PCA is used for dimensionality reduction, it is fitted on the training split only and then applied unchanged to validation and test participants.

Auxiliary Covariates. For LINEAR, we include age, biological sex, and BMI value as input. These covariates are excluded for tasks where they correspond to, or directly determine, the prediction target: Age, Biological Sex, BMI Value, BMI Category, Body Weight, and Framingham CVD Risk.

Outcome-specific prediction heads. For binary outcomes, all models except XGBOOST and GRU-D use a logistic regression head. Similarly, for ordinal outcomes, all models use the Frank–Hall method [Frank and Hall, 2001]. Specifically, for an outcome with ordered classes $\{0, 1, \ldots, K-1\}$ and ground-truth label $y \in \{0, 1, \ldots, K-1\}$, we train $K-1$ binary threshold predictors to estimate whether the label exceeds threshold k, for $k = 0, \ldots, K-2$. For continuous outcomes, all models except XGBOOST and GRU-D use ordinary least-squares regression. XGBOOST and GRU-D use model-specific prediction heads and losses, described in their respective sections.

Handling missing or non-finite predictions. When a model cannot produce a prediction for an eligible outcome instance (e.g. no eligible input segment) or emits a non-finite (`NaN` or `Inf`) value, the harness substitutes the canonical Track-1 baseline (LINEAR) prediction for that instance before scoring and reports the substitution (fallback) rate — the same model-agnostic contract used by the imputation and forecasting tracks. This generalizes the WBM weekly-tensor routing (the WBM model in Appendix E.2, where participants with no valid weekly tensor fall back to LINEAR) to all Track-1 models.

E.2.1. Linear

LINEAR uses summary-statistic features computed from the retained daily hourly segments (defined in Appendix D.3.2). For each retained daily hourly segment, we compute the mean and standard deviation of each of the 19 sensor channels (ignoring NaNs), yielding a 38-dimensional segment-level feature vector [Erturk et al., 2025]. The segment-level feature vectors are averaged across retained daily segments in the model input context to obtain a single 38-dimensional representation.

E.2.2. XGBoost

XGBoost uses hand-crafted features [Shwartz-Ziv and Armon, 2022, Grinsztajn et al., 2022] from daily minute-level segments. Features are organized in three types:

(i) **Daily descriptors** (177 features): for each retained day we extract activity totals, heart-rate summaries (resting, daytime, nighttime), physical-activity intensity zones, sleep duration and timing, workout statistics, and non-parametric circadian rhythm indices (interdaily stability, intradaily variability, L5/M10 relative amplitude); these are then aggregated across the participant's full available wearable history using robust summary statistics (P5, median, P95, IQR).

(ii) **Day-to-day dynamics** (266 features): 14 daily metrics spanning steps, distance, flights, heart-rate percentiles, energy expenditure, sleep and in-bed minutes, active minutes, and workout time are treated as longitudinal time series over each participant's observation period, from which we extract distributional statistics, ARIMA(2,1,0) parameters (fit only on participants with at least 15 retained days), and pairwise cross-correlations at lags 0–2 over an 8-metric subset (28 pairs $\times$ 3 lags).

(iii) **Curve analysis** (52 features): each participant's minute-level daily curves are averaged across days, yielding mean 24-hour profiles on which we compute functional PCA scores (10 components $\times$ 4 channels), single-component cosinor parameters for heart rate (MESOR, amplitude, acrophase, fit R^2, p-value, amplitude-to-MESOR ratio, and fitted-minute count; 7 features), and 5 heart-rate-over-steps (HROS) profile statistics (mean, standard deviation, daytime mean, nighttime mean, and day-to-night ratio).

A separate XGBoost model is fitted for each prediction task using the task-specific prediction heads. We use a hyperparameter configuration selected using validation-set performance (1000 estimators, tree depth 2, learning rate 0.05, row subsampling 0.8, column subsampling 0.3, $\ell_1 = 0.1$, $\ell_2 = 1.0$). NaN features are passed to XGBoost natively without imputation.

E.2.3. MultiRocket

MultiRocket is a deterministic convolutional transform for multivariate time series [Tan et al., 2022]. We apply it on hourly data derived according to Appendix D.3.2, treating each retained daily segment as a multivariate sequence over 19 sensor channels. We z-score normalize the hourly values and zero-fill missing entries before applying the transform. The resulting 19-channel tensor is then passed to MultiRocket. We apply 6,216 kernels combined with four pooling operators to both the hourly value sequence and its first differences, yielding a 49,728-dimensional segment-level representation. Segment-level MultiRocket features are averaged across retained daily segments in the model input context to obtain a 49,728-dimensional participant-level representation. This representation is reduced to 50 dimensions using PCA fitted on the training split.

E.2.4. GRU-D

GRU-D is a recurrent neural network designed for multivariate time series with informative missingness [Che et al., 2022]. GRU-D is trained end-to-end on the prediction tasks and on day-level hourly segments (Appendix D.3.2). We train a shared GRU-D model end-to-end with task-specific linear prediction heads. Hidden states from a participant's retained daily segments are mean-pooled to obtain one 64-dimensional representation. Binary outcomes use classification heads trained with cross-entropy loss, ordinal outcomes use $K-1$ sigmoid threshold heads trained with binary cross-entropy loss, and continuous outcomes use regression heads trained with mean-squared-error loss.

E.2.5. WBM

We follow the Mamba-2-based WBM architecture of Erturk et al. [Erturk et al., 2025] with a bidirectional Mamba-2 encoder over 168-step weekly wearable tensors trained with a contrastive objective. After pretraining, the encoder is frozen. For each participant, each week with sufficient wear time (defined in Appendix D.3.2) is fed into the model to produce a 256-dimensional embedding; weekly embeddings are averaged. This representation is reduced to 50 dimensions using PCA fitted on the training split and then passed to the task-specific linear prediction heads. If a participant does not have any weeks that meet the weekly wear time criteria, we use the LINEAR fallback predictor.

Architecture.

- **Input representation.** The model receives weekly tensors $\mathbf{X} \in \mathbb{R}^{168\times 38}$ consisting of 168 hourly time steps with 19 sensor channels and 19 binary missingness-mask channels (Appendix D.3.2). The hourly values are z-score normalized using training-set hourly statistics, and missing entries are zero-filled before encoder input. The missingness is conveyed explicitly through the corresponding binary mask channels.
- **Hour patch embedding.** Each hourly input vector $\mathbf{x}_t \in \mathbb{R}^{38}$ is independently projected to a d-dimensional token embedding using a two-layer feedforward network:

$$\mathbf{e}_t = W_2\,\mathrm{GELU}(W_1\mathbf{x}_t + b_1) + b_2, \quad \mathbf{e}_t \in \mathbb{R}^d,$$

  where $W_1 \in \mathbb{R}^{h\times 38}$, $W_2 \in \mathbb{R}^{d\times h}$, with hidden dimension h=64 and output dimension d=256. This produces a token sequence $(\mathbf{e}_1, \ldots, \mathbf{e}_{168}) \in \mathbb{R}^{168\times 256}$.
- **Bidirectional Mamba2 encoder.** The token sequence is processed by a stack of L=4 bidirectional Mamba2 blocks. Each block applies a forward Mamba2 pass and a backward (time-reversed) Mamba2 pass in parallel, concatenates their outputs, and projects back to the model dimension:

$$\mathbf{z}_t = W_{\text{proj}}\left[\text{Mamba2}_{\text{fwd}}(\mathbf{e})_t \,\|\, \text{Mamba2}_{\text{bwd}}(\mathbf{e})_t\right] + b_{\text{proj}},$$

  where $W_{\text{proj}} \in \mathbb{R}^{d\times 2d}$ and $\|$ denotes concatenation. This is followed by LayerNorm, a residual connection, and a feedforward network (FFN) with expansion factor 4:

$$\text{FFN}(\mathbf{z}) = W_4\,\mathrm{GELU}(W_3\mathbf{z} + b_3) + b_4, \quad W_3 \in \mathbb{R}^{4d\times d},\ W_4 \in \mathbb{R}^{d\times 4d},$$

  with a second LayerNorm and residual connection. Dropout is applied within the FFN.
- **Pooling and projection.** The encoder output is aggregated via masked mean pooling over the time dimension. During pretraining, a binary keep-mask $\mathbf{m} \in \{0,1\}^{168}$ (from time-step dropout augmentation) determines which tokens contribute to the pool:

$$\mathbf{r} = \frac{\sum_{t=1}^{168} m_t \cdot \mathbf{z}_t}{\sum_{t=1}^{168} m_t}, \quad \mathbf{r} \in \mathbb{R}^{256}.$$

  The representation $\mathbf{r}$ is used for downstream tasks. For the contrastive loss, a linear projection head maps $\mathbf{r}$ to a lower-dimensional embedding that is L_2-normalized:

$$\mathbf{h} = \frac{W_p\mathbf{r} + b_p}{\|W_p\mathbf{r} + b_p\|_2}, \quad \mathbf{h} \in \mathbb{R}^{128}.$$

Pretraining Objective.

- **Contrastive views.** Each training batch samples B participants, drawing one random week per participant. Two augmented views of each weekly tensor are created using time-step dropout: each hourly token is independently dropped with probability p_{drop}=0.223, and the surviving tokens are pooled as in Eq. (E.2.5). This yields two normalized embeddings $\mathbf{h}_i^{(1)}, \mathbf{h}_i^{(2)}$ per participant i.

- **Symmetric InfoNCE.** We use a symmetric InfoNCE loss with temperature τ=0.2. Only diagonal pairs (same participant, different views) serve as positives; off-diagonal pairs from the same participant are masked from the denominator to prevent trivial shortcuts:

$$\mathcal{L}_{\text{InfoNCE}} = -\frac{1}{2B}\sum_{i=1}^{B}\left[\log\frac{\exp(\mathbf{h}_i^{(1)\top}\mathbf{h}_i^{(2)}/\tau)}{\sum_{j\in\mathcal{N}_i}\exp(\mathbf{h}_i^{(1)\top}\mathbf{h}_j^{(2)}/\tau)} + \log\frac{\exp(\mathbf{h}_i^{(2)\top}\mathbf{h}_i^{(1)}/\tau)}{\sum_{j\in\mathcal{N}_i}\exp(\mathbf{h}_i^{(2)\top}\mathbf{h}_j^{(1)}/\tau)}\right],$$

  where $\mathcal{N}_i$ excludes other samples from the same participant in the batch.
- **KoLeo regularization.** To encourage uniform coverage of the embedding space, we add KoLeo regularization [Sablayrolles et al., 2019], which maximizes the log nearest-neighbour distance:

$$\mathcal{L}_{\text{KoLeo}} = -\frac{1}{2B}\sum_{v\in\{1,2\}}\sum_{i=1}^{B}\log\left(\min_{j\neq i}\|\mathbf{h}_i^{(v)} - \mathbf{h}_j^{(v)}\|_2^2 + \epsilon\right).$$

- **Total loss.** The final pretraining objective is:

$$\mathcal{L} = \mathcal{L}_{\text{InfoNCE}} + \lambda_{\text{KoLeo}}\,\mathcal{L}_{\text{KoLeo}},$$

  with λ_{KoLeo}=0.689 selected by hyperparameter optimization.

Training Configuration. Table 14 summarizes the final WBM pretraining configuration used for the selected checkpoint. Architecture, batching, normalization, and optimizer settings are fixed across pretraining runs, while the token dropout probability, KoLeo weight, and learning rate are selected through the hyperparameter search described below.

Hyperparameter Selection. WBM requires a separate pretraining stage before health outcome evaluation. We select its pretraining configuration using the self-supervised validation loss. After pretraining, the selected encoder is frozen and reused across health outcome tasks.

Because exhaustive pretraining sweeps are computationally expensive, we use a two-stage selection procedure. The first stage identifies which pretraining hyperparameters most affect validation InfoNCE loss; the second stage performs a targeted Bayesian sweep over this reduced set of hyperparameters.

- **Phase 1: Sensitivity screening.** We first run a one-factor-at-a-time sensitivity screen on a proxy subset of 1,783 participants. In each run, one candidate hyperparameter is varied while the remaining hyperparameters are held fixed at their reference values. Runs are compared using validation InfoNCE loss on held-out weekly tensors. This phase is used only to reduce the search space, not to select the final pretrained encoder. Based on this screen, we retain three hyperparameters for the final sweep: token drop probability (`drop_prob`), KoLeo regularization weight (`lambda_koleo`), and learning rate (`lr`).
- **Phase 2: Bayesian optimization.** After Phase 1, we run targeted Bayesian optimization using Weights & Biases sweeps [Snoek et al., 2012]. For WBM, the search is restricted to the three hyperparameters retained from sensitivity screening: token drop probability (`drop_prob`), KoLeo regularization weight (`lambda_koleo`), and learning rate (`lr`). Each trial consists of pretraining WBM from scratch, extracting frozen representations, and evaluating validation InfoNCE loss. To limit compute cost, the sweep is capped at 15 trials and uses Hyperband early termination [Li et al., 2018] with minimum iterations = 5 and reduction factor η = 3.

The final search space and selected WBM hyperparameters are reported in Table 15. All hyperparameters not listed are fixed at their baseline values throughout Phase 2.

Table 14 | WBM pretraining configuration. All hyperparameters not in Table 15 are fixed at the values below.

Parameter	**Value**
Architecture	
Encoder type	Bidirectional Mamba2
Encoder layers	4
Embedding dimension (d)	256
Tokenizer hidden dimension (h)	64
FFN expansion factor	4
Mamba2 internal heads	8
Projection head	Linear
Projection dimension	128
Loss	
Temperature (τ)	0.2
Loss variant	Masked (diagonal positives only)
KoLeo weight (λ_{KoLeo})	0.689
Optimization	
Optimizer	AdamW
Learning rate	1.3×10^{-5}
Weight decay	0.01
Scheduler	Cosine ($\eta_{\min}$=10^{-6})
Warmup ratio	0.09
Gradient clipping	1.0
Precision	bf16-mixed
Epochs	20
Data and batching	
Segment type	Weekly (168 hours)
Input dimension	38 (19 sensors + 19 masks)
Normalization	Hourly z-score (training-set stats)
Minimum valid days per week	5
Participants per batch	128
Weeks per participants per batch	1
Maximum weeks per participants (cap)	50
Augmentation	
Time-step dropout (p_{drop})	0.223

E.2.6. LSM-2.

We additionally evaluate LSM-2 [Xu et al., 2025b], a ViT-1D encoder pretrained on minute-level daily wearable segments (see Appendix F.6.1 for details on the pretraining approach). LSM-2 uses minute-level daily inputs (as defined in Appendix D.3.2). Minute-level values are z-score normalized using training-split statistics, and missing entries are zero-filled before encoder input. Representations from the frozen model at non-masked positions are mean-pooled to a single 384-dimensional day-level vector, and these day-level vectors are averaged across all retained days in the model input context to obtain one participant-level representation. This representation is reduced to 50 dimensions using

Table 15 | Phase 2 search space and selected WBM pretraining configuration. Selected values correspond to the final pretrained checkpoint used for downstream evaluation.

Hyperparameter	Search space	Selected value
Token drop probability (`drop_prob`)	Uniform $[0.05, 0.25]$	0.223
KoLeo regularization weight (`lambda_koleo`)	Uniform $[0.0, 1.0]$	0.689
Learning rate (`lr`)	Log-uniform $[10^{-5}, 5 \times 10^{-4}]$	1.3×10^{-5}

PCA fitted on the training split and then passed to the task-specific prediction heads.

E.2.7. Toto (Time-series foundation model)

We use the Toto [Cohen et al., 2025] model fine-tuned on forecasting to extract representations for prediction (see Appendix G.2.3 for fine-tuning details). Toto operates on the hourly wearable representation defined in Appendix D.3.2. Specifically, we extract last-layer latent representations, yielding channel-wise embeddings of shape $(19, 768)$, with one 768-dimensional vector per wearable channel. These channel-wise embeddings are mean-pooled across the 19 channels to obtain one 768-dimensional participant-level representation. This representation is reduced to 50 dimensions using PCA fitted on the training split and then passed to the task-specific linear prediction heads.

E.2.8. Chronos-2 (Time-series foundation model)

We use a Chronos-2 model [Ansari et al., 2025] fine-tuned on forecasting to extract representations for prediction (see Appendix G.2.4 for fine-tuning details). Chronos-2 operates on the same hourly wearable representation used by Toto, as defined in Appendix D.3.2. We extract last-layer latent representations, yielding channel-wise embedding matrix of dimension 19×768, with one 768-dimensional vector per wearable channel. As with Toto, these channel-wise embeddings are mean-pooled across the 19 channels to obtain one 768-dimensional participant-level representation. This representation is reduced to 50 dimensions using PCA fitted on the training split and then passed to the task-specific prediction heads.

E.3. Additional Prediction Task Results

Table 16 | **Prediction Per-Task Results.** Per-task primary metric across the 32 outcome tasks in five clinical domains. AUPRC (binary), Spearman ρ (ordinal), Pearson r (regression), all ↑. Values are point estimates on the held-out test split; superscripts and subscripts indicate the 95% percentile bootstrap confidence interval (B=1,000). Macro Skill S = mean of the 5 per-domain S (%; 0=LINEAR reference); aggregate companion in Table 2. See Appendix B.1 for Skill score details.

Domain	Task	Metric	LINEAR	MULTIROCKET	XGBOOST	GRU-D	LSM-2	WBM	TOTO	CHRONOS-2
Demographics	Age	Pearson r ↑	$0.284^{+0.042}_{-0.042}$	$0.437^{+0.035}_{-0.039}$	$\mathbf{0.585}^{+0.033}_{-0.034}$	$0.347^{+0.042}_{-0.043}$	$0.556^{+0.033}_{-0.036}$	$0.423^{+0.036}_{-0.039}$	$0.404^{+0.037}_{-0.041}$	$0.394^{+0.040}_{-0.040}$
	Biological Sex	AUPRC ↑	$0.876^{+0.019}_{-0.023}$	$0.918^{+0.015}_{-0.017}$	$\mathbf{0.940}^{+0.013}_{-0.015}$	$0.909^{+0.018}_{-0.019}$	$0.931^{+0.013}_{-0.016}$	$0.907^{+0.015}_{-0.018}$	$0.849^{+0.022}_{-0.025}$	$0.871^{+0.021}_{-0.024}$
	Domain Avg. Rank		$7.00^{+0.50}_{-0.00}$	$3.00^{+1.00}_{-0.01}$	$\mathbf{1.00}^{+0.50}_{-0.00}$	$5.50^{+0.50}_{-0.50}$	$2.00^{+0.50}_{-0.50}$	$4.50^{+1.00}_{-1.00}$	$6.50^{+0.50}_{-1.00}$	$6.50^{+0.00}_{-1.50}$
	Domain Skill S (%)		0.0	$+27.7^{+5.5}_{-5.2}$	$\mathbf{+46.9}^{+5.4}_{-6.0}$	$+18.2^{+8.2}_{-8.0}$	$+41.1^{+5.6}_{-6.4}$	$+22.2^{+4.3}_{-4.6}$	$-0.9^{+8.9}_{-9.4}$	$+6.2^{+8.7}_{-9.0}$
Medical conditions & risk	Atrial Fibrillation	AUPRC ↑	$\mathbf{0.099}^{+0.130}_{-0.051}$	$0.030^{+0.026}_{-0.012}$	$0.030^{+0.018}_{-0.012}$	$0.023^{+0.013}_{-0.008}$	$0.037^{+0.025}_{-0.015}$	$0.073^{+0.110}_{-0.038}$	$0.028^{+0.041}_{-0.012}$	$0.030^{+0.027}_{-0.013}$
	Cardiovascular Disease	AUPRC ↑	$\mathbf{0.370}^{+0.101}_{-0.078}$	$0.230^{+0.069}_{-0.048}$	$0.293^{+0.080}_{-0.065}$	$0.175^{+0.062}_{-0.040}$	$0.269^{+0.080}_{-0.053}$	$0.302^{+0.090}_{-0.063}$	$0.221^{+0.068}_{-0.048}$	$0.191^{+0.059}_{-0.039}$
	Cerebrovascular Disease	AUPRC ↑	$0.066^{+0.070}_{-0.032}$	$0.027^{+0.042}_{-0.012}$	$0.032^{+0.023}_{-0.013}$	$0.026^{+0.022}_{-0.010}$	$0.042^{+0.049}_{-0.021}$	$\mathbf{0.082}^{+0.119}_{-0.048}$	$0.024^{+0.016}_{-0.009}$	$0.024^{+0.015}_{-0.009}$
	Congenital Heart Disease	AUPRC ↑	$0.016^{+0.048}_{-0.011}$	$0.012^{+0.019}_{-0.007}$	$\mathbf{0.019}^{+0.040}_{-0.012}$	$0.017^{+0.054}_{-0.011}$	$0.016^{+0.037}_{-0.009}$	$0.018^{+0.037}_{-0.012}$	$0.011^{+0.025}_{-0.007}$	$0.009^{+0.011}_{-0.004}$
	Coronary Artery Disease	AUPRC ↑	$0.137^{+0.106}_{-0.060}$	$0.068^{+0.039}_{-0.022}$	$0.067^{+0.040}_{-0.024}$	$0.086^{+0.089}_{-0.037}$	$0.116^{+0.073}_{-0.049}$	$\mathbf{0.147}^{+0.116}_{-0.057}$	$0.088^{+0.082}_{-0.035}$	$0.065^{+0.035}_{-0.021}$
	Diabetes	AUPRC ↑	$0.131^{+0.105}_{-0.056}$	$0.093^{+0.086}_{-0.038}$	$0.077^{+0.067}_{-0.028}$	$0.128^{+0.149}_{-0.064}$	$\mathbf{0.137}^{+0.120}_{-0.053}$	$0.095^{+0.091}_{-0.041}$	$0.092^{+0.093}_{-0.042}$	$0.080^{+0.062}_{-0.032}$
	Framingham CVD Risk	Pearson r ↑	$0.139^{+0.234}_{-0.157}$	$0.247^{+0.126}_{-0.134}$	$\mathbf{0.318}^{+0.099}_{-0.104}$	$0.203^{+0.101}_{-0.110}$	$0.272^{+0.093}_{-0.099}$	$0.173^{+0.193}_{-0.153}$	$0.190^{+0.098}_{-0.102}$	$0.066^{+0.133}_{-0.130}$
	Heart Failure / CHF	AUPRC ↑	$0.035^{+0.044}_{-0.022}$	$0.012^{+0.024}_{-0.008}$	$0.018^{+0.049}_{-0.013}$	$0.056^{+0.210}_{-0.049}$	$\mathbf{0.088}^{+0.310}_{-0.080}$	$0.032^{+0.051}_{-0.021}$	$0.012^{+0.026}_{-0.008}$	$0.017^{+0.047}_{-0.013}$
	Hypertension	AUPRC ↑	$\mathbf{0.585}^{+0.080}_{-0.083}$	$0.440^{+0.091}_{-0.076}$	$0.487^{+0.078}_{-0.078}$	$0.402^{+0.091}_{-0.071}$	$0.451^{+0.089}_{-0.084}$	$0.518^{+0.085}_{-0.084}$	$0.368^{+0.089}_{-0.058}$	$0.351^{+0.083}_{-0.064}$
	Pulmonary Hypertension	AUPRC ↑	$0.019^{+0.047}_{-0.013}$	$0.152^{+0.412}_{-0.146}$	$\mathbf{0.190}^{+0.336}_{-0.182}$	$0.047^{+0.156}_{-0.038}$	$0.035^{+0.072}_{-0.024}$	$0.013^{+0.037}_{-0.010}$	$0.025^{+0.092}_{-0.021}$	$0.023^{+0.058}_{-0.016}$
	Sleep Disorder Diagnosis	AUPRC ↑	$0.267^{+0.051}_{-0.042}$	$0.246^{+0.053}_{-0.039}$	$0.281^{+0.067}_{-0.046}$	$0.260^{+0.056}_{-0.044}$	$\mathbf{0.343}^{+0.072}_{-0.055}$	$0.272^{+0.053}_{-0.044}$	$0.237^{+0.059}_{-0.041}$	$0.247^{+0.061}_{-0.040}$
	Vascular Disease	AUPRC ↑	$0.029^{+0.131}_{-0.024}$	$0.007^{+0.012}_{-0.004}$	$0.006^{+0.011}_{-0.004}$	$0.011^{+0.019}_{-0.007}$	$0.010^{+0.024}_{-0.007}$	$\mathbf{0.037}^{+0.186}_{-0.033}$	$0.007^{+0.011}_{-0.004}$	$0.009^{+0.016}_{-0.005}$
	Domain Avg. Rank		$3.00^{+0.67}_{-0.67}$	$5.25^{+0.75}_{-0.83}$	$3.92^{+1.17}_{-0.42}$	$4.67^{+1.00}_{-0.67}$	$\mathbf{2.92}^{+1.00}_{-0.42}$	$3.00^{+1.00}_{-0.50}$	$6.42^{+0.33}_{-1.50}$	$6.83^{+0.17}_{-1.33}$
	Domain Skill S (%)		**0.0**	$-4.5^{+5.8}_{-4.9}$	$-1.4^{+5.1}_{-5.7}$	$-6.2^{+3.7}_{-4.4}$	$-1.8^{+4.0}_{-4.4}$	$-2.0^{+2.3}_{-2.1}$	$-7.6^{+2.9}_{-4.7}$	$-9.6^{+2.9}_{-5.2}$
Vitals & blood biomarkers	BMI Categories	Spearman ρ ↑	$0.337^{+0.047}_{-0.045}$	$0.378^{+0.048}_{-0.048}$	$0.457^{+0.043}_{-0.045}$	$0.436^{+0.043}_{-0.042}$	$\mathbf{0.563}^{+0.039}_{-0.040}$	$0.401^{+0.048}_{-0.044}$	$0.056^{+0.050}_{-0.056}$	$0.064^{+0.054}_{-0.055}$
	BMI Value	Pearson r ↑	$0.322^{+0.099}_{-0.075}$	$0.515^{+0.049}_{-0.049}$	$0.593^{+0.070}_{-0.084}$	$0.517^{+0.046}_{-0.045}$	$\mathbf{0.706}^{+0.038}_{-0.038}$	$0.481^{+0.090}_{-0.078}$	$0.228^{+0.117}_{-0.113}$	$0.217^{+0.123}_{-0.124}$
	Blood Pressure Categories	Spearman ρ ↑	$0.107^{+0.096}_{-0.094}$	$0.073^{+0.092}_{-0.091}$	$0.077^{+0.095}_{-0.094}$	$\mathbf{0.284}^{+0.082}_{-0.089}$	$0.172^{+0.089}_{-0.094}$	$0.106^{+0.091}_{-0.089}$	$0.070^{+0.089}_{-0.099}$	$0.075^{+0.094}_{-0.101}$
	Body Weight	Pearson r ↑	$0.311^{+0.092}_{-0.074}$	$0.485^{+0.052}_{-0.049}$	$0.557^{+0.047}_{-0.056}$	$0.460^{+0.045}_{-0.052}$	$\mathbf{0.649}^{+0.044}_{-0.052}$	$0.431^{+0.083}_{-0.082}$	$0.521^{+0.050}_{-0.049}$	$0.519^{+0.049}_{-0.049}$
	HDL Cholesterol	Pearson r ↑	$0.186^{+0.111}_{-0.102}$	$0.153^{+0.103}_{-0.108}$	$0.168^{+0.097}_{-0.104}$	$0.148^{+0.091}_{-0.092}$	$0.151^{+0.095}_{-0.105}$	$\mathbf{0.213}^{+0.115}_{-0.105}$	$0.140^{+0.080}_{-0.087}$	$0.052^{+0.088}_{-0.097}$
	LDL Cholesterol	Pearson r ↑	$0.087^{+0.089}_{-0.085}$	$-0.049^{+0.096}_{-0.096}$	$0.089^{+0.094}_{-0.102}$	$-0.030^{+0.111}_{-0.108}$	$-0.018^{+0.131}_{-0.120}$	$0.014^{+0.084}_{-0.098}$	$\mathbf{0.105}^{+0.092}_{-0.102}$	$-0.050^{+0.102}_{-0.107}$
	Systolic Blood Pressure	Pearson r ↑	$0.162^{+0.093}_{-0.094}$	$0.159^{+0.092}_{-0.094}$	$0.134^{+0.088}_{-0.086}$	$0.201^{+0.089}_{-0.094}$	$\mathbf{0.260}^{+0.086}_{-0.082}$	$0.170^{+0.094}_{-0.090}$	$0.037^{+0.093}_{-0.088}$	$0.043^{+0.081}_{-0.089}$
	Total Cholesterol	Pearson r ↑	$-0.005^{+0.068}_{-0.063}$	$-0.012^{+0.091}_{-0.079}$	$\mathbf{0.093}^{+0.089}_{-0.090}$	$0.037^{+0.100}_{-0.102}$	$0.022^{+0.099}_{-0.091}$	$0.014^{+0.080}_{-0.060}$	$-0.025^{+0.080}_{-0.085}$	$0.035^{+0.079}_{-0.084}$
	Domain Avg. Rank		$4.75^{+1.12}_{-0.50}$	$5.50^{+0.62}_{-1.38}$	$2.88^{+1.12}_{-0.62}$	$3.62^{+1.12}_{-0.62}$	$\mathbf{2.50}^{+1.00}_{-0.75}$	$4.12^{+1.12}_{-1.50}$	$6.25^{+0.12}_{-1.50}$	$6.38^{+0.62}_{-0.88}$
	Domain Skill S (%)		0.0	$+5.6^{+3.7}_{-4.5}$	$+13.6^{+4.1}_{-4.6}$	$+10.5^{+3.9}_{-4.5}$	$\mathbf{+22.2}^{+3.4}_{-4.2}$	$+6.6^{+2.3}_{-2.6}$	$-4.5^{+4.6}_{-6.1}$	$-7.1^{+4.6}_{-6.4}$
Mental well-being	Feel Depressed	Spearman ρ ↑	$0.068^{+0.079}_{-0.075}$	$\mathbf{0.116}^{+0.087}_{-0.085}$	$0.055^{+0.080}_{-0.077}$	$0.095^{+0.081}_{-0.075}$	$0.110^{+0.085}_{-0.077}$	$0.055^{+0.086}_{-0.085}$	$0.063^{+0.082}_{-0.083}$	$0.070^{+0.076}_{-0.082}$
	Feel Happy	Spearman ρ ↑	$\mathbf{0.165}^{+0.065}_{-0.064}$	$0.121^{+0.067}_{-0.070}$	$0.004^{+0.068}_{-0.069}$	$0.106^{+0.071}_{-0.068}$	$0.135^{+0.072}_{-0.057}$	$0.059^{+0.064}_{-0.067}$	$0.062^{+0.067}_{-0.064}$	$0.031^{+0.069}_{-0.061}$
	Feel Worried	Spearman ρ ↑	$0.089^{+0.068}_{-0.066}$	$0.135^{+0.065}_{-0.070}$	$0.058^{+0.069}_{-0.067}$	$0.085^{+0.066}_{-0.068}$	$\mathbf{0.177}^{+0.071}_{-0.068}$	$0.032^{+0.069}_{-0.068}$	$0.065^{+0.066}_{-0.069}$	$0.095^{+0.067}_{-0.072}$
	Life Satisfaction	Spearman ρ ↑	$\mathbf{0.221}^{+0.060}_{-0.067}$	$0.159^{+0.064}_{-0.066}$	$0.080^{+0.066}_{-0.070}$	$0.116^{+0.066}_{-0.070}$	$0.169^{+0.067}_{-0.064}$	$0.148^{+0.066}_{-0.065}$	$0.059^{+0.068}_{-0.070}$	$0.075^{+0.067}_{-0.065}$
	Things Are Worthwhile	Spearman ρ ↑	$0.139^{+0.066}_{-0.061}$	$\mathbf{0.168}^{+0.066}_{-0.070}$	$0.124^{+0.067}_{-0.063}$	$0.118^{+0.067}_{-0.067}$	$0.129^{+0.066}_{-0.068}$	$0.068^{+0.069}_{-0.065}$	$0.045^{+0.071}_{-0.069}$	$0.091^{+0.068}_{-0.069}$
	Domain Avg. Rank		$2.60^{+1.60}_{-0.80}$	$\mathbf{2.00}^{+1.80}_{-0.40}$	$6.40^{+0.80}_{-1.80}$	$4.40^{+1.40}_{-1.60}$	$\mathbf{2.00}^{+1.60}_{-0.40}$	$6.60^{+0.20}_{-2.00}$	$6.60^{+0.80}_{-1.60}$	$5.40^{+1.60}_{-1.01}$
	Domain Skill S (%)		0.0	$+0.2^{+4.3}_{-4.9}$	$-8.5^{+4.6}_{-4.7}$	$-3.9^{+4.5}_{-4.8}$	$\mathbf{+0.7}^{+4.0}_{-4.2}$	$-7.5^{+3.6}_{-3.5}$	$-9.2^{+4.3}_{-4.9}$	$-7.6^{+4.5}_{-5.4}$
Sleep & lifestyle	Bedtime	Spearman ρ ↑	$0.108^{+0.052}_{-0.052}$	$0.190^{+0.053}_{-0.055}$	$\mathbf{0.221}^{+0.053}_{-0.055}$	$0.089^{+0.051}_{-0.055}$	$0.210^{+0.051}_{-0.053}$	$0.099^{+0.050}_{-0.052}$	$0.132^{+0.048}_{-0.052}$	$0.136^{+0.051}_{-0.049}$
	Currently Employed	AUPRC ↑	$0.871^{+0.027}_{-0.026}$	$0.894^{+0.023}_{-0.022}$	$0.901^{+0.021}_{-0.021}$	$0.871^{+0.026}_{-0.025}$	$\mathbf{0.920}^{+0.018}_{-0.017}$	$0.858^{+0.027}_{-0.028}$	$0.885^{+0.021}_{-0.022}$	$0.892^{+0.021}_{-0.020}$
	Sleep Duration	Spearman ρ ↑	$0.059^{+0.054}_{-0.063}$	$0.029^{+0.059}_{-0.057}$	$0.018^{+0.057}_{-0.057}$	$0.017^{+0.059}_{-0.056}$	$0.070^{+0.060}_{-0.059}$	$\mathbf{0.081}^{+0.055}_{-0.062}$	$-0.032^{+0.056}_{-0.057}$	$0.035^{+0.057}_{-0.059}$
	Vigorous Activity Minutes	Pearson r ↑	$0.213^{+0.098}_{-0.097}$	$\mathbf{0.222}^{+0.100}_{-0.094}$	$0.186^{+0.064}_{-0.066}$	$0.118^{+0.069}_{-0.066}$	$0.222^{+0.071}_{-0.059}$	$0.194^{+0.090}_{-0.082}$	$0.078^{+0.058}_{-0.051}$	$0.096^{+0.057}_{-0.055}$
	Wake-up Time	Spearman ρ ↑	$0.133^{+0.050}_{-0.050}$	$0.194^{+0.049}_{-0.051}$	$0.179^{+0.056}_{-0.056}$	$0.147^{+0.050}_{-0.050}$	$\mathbf{0.214}^{+0.051}_{-0.049}$	$0.108^{+0.052}_{-0.052}$	$0.118^{+0.053}_{-0.052}$	$0.134^{+0.052}_{-0.050}$
	Domain Avg. Rank		$4.80^{+1.20}_{-1.00}$	$2.80^{+1.40}_{-0.60}$	$3.40^{+1.00}_{-1.20}$	$6.40^{+0.60}_{-1.41}$	$\mathbf{1.60}^{+1.00}_{-0.40}$	$5.60^{+1.00}_{-1.00}$	$6.60^{+0.60}_{-1.20}$	$4.80^{+1.40}_{-0.80}$
	Domain Skill S (%)		0.0	$+6.6^{+4.4}_{-4.3}$	$+7.3^{+4.5}_{-4.8}$	$-3.3^{+5.4}_{-6.4}$	$\mathbf{+13.4}^{+4.0}_{-4.2}$	$-2.8^{+3.1}_{-3.4}$	$-2.6^{+5.3}_{-5.5}$	$+0.9^{+4.5}_{-5.4}$
	Macro Avg. Rank		$4.43^{+0.58}_{-0.20}$	$3.71^{+0.59}_{-0.23}$	$3.52^{+0.43}_{-0.39}$	$4.92^{+0.46}_{-0.44}$	$\mathbf{2.20}^{+0.60}_{-0.11}$	$4.76^{+0.31}_{-0.54}$	$6.47^{+0.03}_{-0.79}$	$5.98^{+0.32}_{-0.56}$
	Overall Skill S (%)		0.0	$+7.1^{+2.1}_{-2.1}$	$+11.6^{+2.2}_{-2.3}$	$+3.1^{+2.5}_{-2.6}$	$\mathbf{+15.1}^{+1.9}_{-2.3}$	$+3.3^{+1.4}_{-1.5}$	$-5.0^{+2.3}_{-2.9}$	$-3.4^{+2.3}_{-2.9}$

E.4. Gemini-Family LLM Baselines

Scope. This appendix evaluates the GEMINI family of frontier large language models (GEMINI-2.5-PRO and GEMINI-3.1-PRO-PREVIEW [Comanici et al., 2025]) as zero-shot/few-shot baselines on the 32 outcome prediction tasks in our benchmark. *It is not an exhaustive characterization of LLM capabilities on wearable-sensor health tasks*: cross-vendor evaluation and broader sweeps over open-weights LLMs are out of scope and left to future work. Findings here should be read as Gemini-family analysis, not a general LLM-vs-supervised verdict.

Headline finding. All five Gemini-family baseline configurations we evaluate underperform the LINEAR baseline on this cohort in macro Skill S (Table 17): the strongest probe (GEM-3.1 STATS) is −10.3 skill points relative to LINEAR and ~29 points below LSM-2. Per-task wins for the Gemini baselines total 10 of 32 outcomes, of which 8 fall in the rare-event Medical conditions & risk domain where every method's standard error is wide; the Gemini baselines win zero Demographics, zero Sleep & lifestyle, and only one Vitals & blood biomarkers task.

Setup. To complement the supervised methods reported in Table 2, we evaluate the Gemini baselines under three prompting strategies: *statistics* (38-channel weekly summary statistics rendered as text), *vision* (rendered weekly sensor heatmaps; rendering recipe and prompt template in Appendix E.4.1), and *agentic* (a modified version of the PHA Data Science Agent [Merrill et al., 2026, Heydari et al., 2025] with k=3 retrieved few-shot exemplars and tool-use; built on Gemini-2.5-Pro). We evaluate both Gemini variants per non-agentic strategy, yielding five probe configurations in total ($2 \times 2 + 1$). Because all probe strategies require a coherent week of data, we restrict evaluation to participants with ≥ 5 valid days within a 7-day window; per-task participant-method intersection cohorts span 119–600 participants (median 388) across the 32 tasks shown here, drawn from a strict subset of the 1,637-participant canonical cohort used in Table 2. LINEAR, XGBOOST, and LSM-2 columns are recomputed on each task's intersection cohort for direct paired comparison and may differ slightly from Table 2.

Per-task pattern. The per-task breakdown (Table 17) sharpens the picture: the Gemini baselines are competitive on roughly a third of tasks — almost entirely low-prevalence Medical conditions & risk binary outcomes where AUPRC standard errors are wide (e.g., Atrial Fibrillation, Cerebrovascular Disease, Heart Failure / CHF, Hypertension, Coronary Artery Disease) — but lose on every Demographics and Sleep & lifestyle task and on 7 of 8 Vitals & blood biomarkers tasks, where the supervised methods exploit signal that the Gemini baselines do not extract from text or vision representations of the same week. Two configurations (GEM-2.5 VIS and the agentic PHA-DSA) win zero tasks outright; PHA-DSA also has the worst macro Skill S and macro average rank in the table, indicating that agentic tooling does not rescue performance on this benchmark within the Gemini family.

Related Work on LLMs for Wearable Health Prediction. While our evaluation is scoped to the Gemini family, concurrent work documents a similar pattern across vendors and prompting modes, suggesting our findings are not idiosyncratic to a single vendor: PH-LLM [Cosentino et al., 2024] reports parity (not superiority) of fine-tuned Gemini-Ultra-1.0 with logistic regression on numerical wearable predictions; OpenTSLM [Langer et al., 2025] shows GPT-4o below random on human-activity recognition; Tan et al. [2024] demonstrate that removing the LLM from popular time-series-LLM pipelines often *improves* performance. We position our results as a within-Gemini analysis whose qualitative direction is consistent with these vendor-diverse reports; a controlled cross-vendor benchmark on this exact task suite remains future work.

Table 17 | **Gemini-Family LLM Baselines: Per-Task Primary Metrics (Cohort-Matched).** Per-task primary metric on the 8-method weekly cohort: participants with ≥ 5 valid days per 7-day window, per-task participant-method intersection 119–600 participants (median 388), strict subset of the 1,637-participant canonical cohort in Table 2. AUPRC (binary), Spearman ρ (ordinal), Pearson r (regression), all ↑. LINEAR, XGBOOST, LSM-2 recomputed on each task's intersection cohort. Standard errors from paired participant-level bootstrap (B=1,000). Macro Skill S = mean of the 5 per-domain S (%; 0=LINEAR reference).

Domain	Task	Metric	LINEAR	XGBOOST	LSM-2	GEM-2.5 ST	GEM-3.1 ST	GEM-2.5 VIS	GEM-3.1 VIS	PHA-DSA
raphics	Age	Pearson r ↑	$0.142_{\pm 0.036}$	$\mathbf{0.659_{\pm 0.028}}$	$0.638_{\pm 0.026}$	$0.086_{\pm 0.042}$	$0.155_{\pm 0.042}$	$0.018_{\pm 0.042}$	$0.055_{\pm 0.040}$	$-0.016_{\pm 0.043}$
	Biological Sex	AUPRC ↑	$0.816_{\pm 0.023}$	$\mathbf{0.937_{\pm 0.010}}$	$0.929_{\pm 0.010}$	$0.726_{\pm 0.023}$	$0.769_{\pm 0.023}$	$0.762_{\pm 0.023}$	$0.737_{\pm 0.023}$	$0.732_{\pm 0.026}$
	Domain Avg. Rank		3.50	**1.00**	2.00	6.50	3.50	6.00	6.00	7.50
	Domain Skill S (%)		$0.0_{\pm 0.0}$	$\mathbf{+63.1_{\pm 3.1}}$	$+59.5_{\pm 3.1}$	$-26.7_{\pm 8.1}$	$-11.9_{\pm 7.8}$	$-22.6_{\pm 8.5}$	$-26.5_{\pm 8.3}$	$-32.0_{\pm 9.3}$
Medical conditions & risk	Atrial Fibrillation	AUPRC ↑	$0.113_{\pm 0.059}$	$0.046_{\pm 0.017}$	$0.103_{\pm 0.042}$	$\mathbf{0.117_{\pm 0.049}}$	$0.097_{\pm 0.052}$	$0.055_{\pm 0.017}$	$0.092_{\pm 0.057}$	$0.049_{\pm 0.018}$
	Cardiovascular Disease	AUPRC ↑	$\mathbf{0.440_{\pm 0.063}}$	$0.411_{\pm 0.058}$	$0.388_{\pm 0.057}$	$0.292_{\pm 0.046}$	$0.415_{\pm 0.059}$	$0.308_{\pm 0.049}$	$0.372_{\pm 0.057}$	$0.292_{\pm 0.049}$
	Cerebrovascular Disease	AUPRC ↑	$0.043_{\pm 0.025}$	$0.048_{\pm 0.034}$	$0.051_{\pm 0.032}$	$0.025_{\pm 0.012}$	$\mathbf{0.079_{\pm 0.070}}$	$0.038_{\pm 0.021}$	$0.066_{\pm 0.062}$	$0.048_{\pm 0.033}$
	Congenital Heart Disease	AUPRC ↑	$0.027_{\pm 0.021}$	$0.016_{\pm 0.010}$	$0.045_{\pm 0.060}$	$\mathbf{0.057_{\pm 0.057}}$	$0.020_{\pm 0.017}$	$0.050_{\pm 0.071}$	$0.019_{\pm 0.010}$	$0.019_{\pm 0.008}$
	Coronary Artery Disease	AUPRC ↑	$0.160_{\pm 0.055}$	$0.112_{\pm 0.037}$	$0.117_{\pm 0.042}$	$0.113_{\pm 0.038}$	$\mathbf{0.175_{\pm 0.062}}$	$0.138_{\pm 0.055}$	$0.116_{\pm 0.039}$	$0.119_{\pm 0.039}$
	Diabetes	AUPRC ↑	$0.238_{\pm 0.074}$	$0.203_{\pm 0.062}$	$0.239_{\pm 0.069}$	$0.173_{\pm 0.047}$	$0.238_{\pm 0.072}$	$0.157_{\pm 0.048}$	$\mathbf{0.241_{\pm 0.074}}$	$0.172_{\pm 0.056}$
	Framingham CVD Risk	Pearson r ↑	$0.172_{\pm 0.092}$	$\mathbf{0.310_{\pm 0.116}}$	$0.099_{\pm 0.112}$	$-0.035_{\pm 0.089}$	$-0.077_{\pm 0.078}$	$-0.103_{\pm 0.081}$	$-0.131_{\pm 0.070}$	$0.058_{\pm 0.067}$
	Heart Failure / CHF	AUPRC ↑	$0.077_{\pm 0.058}$	$0.121_{\pm 0.162}$	$0.097_{\pm 0.085}$	$0.100_{\pm 0.132}$	$0.131_{\pm 0.163}$	$0.128_{\pm 0.114}$	$\mathbf{0.275_{\pm 0.238}}$	$0.180_{\pm 0.167}$
	Hypertension	AUPRC ↑	$0.574_{\pm 0.056}$	$0.566_{\pm 0.056}$	$0.583_{\pm 0.060}$	$0.548_{\pm 0.050}$	$\mathbf{0.598_{\pm 0.054}}$	$0.525_{\pm 0.056}$	$0.563_{\pm 0.054}$	$0.489_{\pm 0.055}$
	Pulmonary Hypertension	AUPRC ↑	$0.035_{\pm 0.030}$	$0.026_{\pm 0.019}$	$\mathbf{0.073_{\pm 0.063}}$	$0.018_{\pm 0.012}$	$0.023_{\pm 0.019}$	$0.016_{\pm 0.008}$	$0.014_{\pm 0.008}$	$0.030_{\pm 0.020}$
	Sleep Disorder Diagnosis	AUPRC ↑	$0.316_{\pm 0.043}$	$\mathbf{0.373_{\pm 0.047}}$	$0.368_{\pm 0.047}$	$0.242_{\pm 0.027}$	$0.295_{\pm 0.037}$	$0.278_{\pm 0.034}$	$0.307_{\pm 0.039}$	$0.278_{\pm 0.041}$
	Vascular Disease	AUPRC ↑	$0.012_{\pm 0.010}$	$0.012_{\pm 0.007}$	$0.015_{\pm 0.010}$	$0.031_{\pm 0.021}$	$0.031_{\pm 0.020}$	$0.070_{\pm 0.091}$	$\mathbf{0.083_{\pm 0.104}}$	$0.032_{\pm 0.019}$
	Domain Avg. Rank		3.67	4.92	3.42	5.58	**3.33**	5.42	4.42	5.25
	Domain Skill S (%)		$0.0_{\pm 0.0}$	$\mathbf{+0.9_{\pm 3.7}}$	$+0.3_{\pm 2.4}$	$-6.4_{\pm 2.9}$	$-1.0_{\pm 3.1}$	$-6.4_{\pm 2.9}$	$-1.2_{\pm 5.9}$	$-5.4_{\pm 3.6}$
tals & blood biomarkers	BMI Categories	Spearman ρ ↑	$0.331_{\pm 0.043}$	$0.541_{\pm 0.037}$	$\mathbf{0.666_{\pm 0.031}}$	$0.121_{\pm 0.047}$	$0.254_{\pm 0.043}$	$0.132_{\pm 0.046}$	$0.027_{\pm 0.046}$	$0.066_{\pm 0.047}$
	BMI Value	Pearson r ↑	$0.430_{\pm 0.053}$	$0.725_{\pm 0.034}$	$\mathbf{0.788_{\pm 0.024}}$	$0.271_{\pm 0.046}$	$0.264_{\pm 0.046}$	$0.181_{\pm 0.051}$	$0.127_{\pm 0.056}$	$0.057_{\pm 0.053}$
	Blood Pressure Categories	Spearman ρ ↑	$0.089_{\pm 0.069}$	$\mathbf{0.261_{\pm 0.066}}$	$0.112_{\pm 0.071}$	$0.108_{\pm 0.074}$	$0.116_{\pm 0.073}$	$0.121_{\pm 0.074}$	$0.181_{\pm 0.070}$	$0.151_{\pm 0.069}$
	Body Weight	Pearson r ↑	$0.945_{\pm 0.005}$	$0.960_{\pm 0.004}$	$\mathbf{0.962_{\pm 0.004}}$	$0.645_{\pm 0.063}$	$0.804_{\pm 0.026}$	$0.754_{\pm 0.053}$	$0.790_{\pm 0.032}$	$0.075_{\pm 0.056}$
	HDL Cholesterol	Pearson r ↑	$\mathbf{0.241_{\pm 0.070}}$	$0.195_{\pm 0.073}$	$0.233_{\pm 0.068}$	$0.162_{\pm 0.072}$	$0.226_{\pm 0.066}$	$0.192_{\pm 0.070}$	$0.166_{\pm 0.068}$	$0.149_{\pm 0.067}$
	LDL Cholesterol	Pearson r ↑	$\mathbf{0.178_{\pm 0.065}}$	$0.170_{\pm 0.065}$	$0.065_{\pm 0.061}$	$0.062_{\pm 0.069}$	$0.141_{\pm 0.065}$	$0.008_{\pm 0.073}$	$0.024_{\pm 0.060}$	$-0.057_{\pm 0.062}$
	Systolic Blood Pressure	Pearson r ↑	$0.143_{\pm 0.069}$	$0.209_{\pm 0.064}$	$\mathbf{0.259_{\pm 0.065}}$	$0.230_{\pm 0.061}$	$0.241_{\pm 0.064}$	$0.246_{\pm 0.066}$	$0.136_{\pm 0.068}$	$0.130_{\pm 0.062}$
	Total Cholesterol	Pearson r ↑	$0.173_{\pm 0.068}$	$0.130_{\pm 0.062}$	$0.065_{\pm 0.063}$	$-0.043_{\pm 0.066}$	$0.105_{\pm 0.065}$	$0.013_{\pm 0.066}$	$\mathbf{0.182_{\pm 0.064}}$	$-0.029_{\pm 0.069}$
	Domain Avg. Rank		3.38	**2.62**	**2.62**	6.00	3.88	5.12	5.25	7.12
	Domain Skill S (%)		$0.0_{\pm 0.0}$	$+18.0_{\pm 2.8}$	$\mathbf{+21.7_{\pm 2.3}}$	$-40.2_{\pm 6.0}$	$-22.5_{\pm 4.5}$	$-34.5_{\pm 5.9}$	$-33.7_{\pm 5.1}$	$-68.3_{\pm 5.9}$
ntal well-being	Feel Depressed	Spearman ρ ↑	$0.153_{\pm 0.066}$	$0.034_{\pm 0.069}$	$\mathbf{0.156_{\pm 0.067}}$	$-0.053_{\pm 0.067}$	$-0.172_{\pm 0.074}$	$-0.087_{\pm 0.072}$	$-0.027_{\pm 0.066}$	$0.007_{\pm 0.067}$
	Feel Happy	Spearman ρ ↑	$\mathbf{0.209_{\pm 0.052}}$	$0.105_{\pm 0.054}$	$0.152_{\pm 0.054}$	$0.167_{\pm 0.053}$	$0.137_{\pm 0.058}$	$0.112_{\pm 0.057}$	$0.001_{\pm 0.054}$	$0.020_{\pm 0.056}$
	Feel Worried	Spearman ρ ↑	$0.129_{\pm 0.060}$	$0.177_{\pm 0.061}$	$\mathbf{0.217_{\pm 0.058}}$	$-0.082_{\pm 0.053}$	$-0.128_{\pm 0.062}$	$-0.152_{\pm 0.057}$	$-0.059_{\pm 0.056}$	$0.005_{\pm 0.058}$
	Life Satisfaction	Spearman ρ ↑	$0.203_{\pm 0.054}$	$0.134_{\pm 0.058}$	$0.189_{\pm 0.053}$	$0.105_{\pm 0.051}$	$\mathbf{0.266_{\pm 0.055}}$	$0.129_{\pm 0.059}$	$0.071_{\pm 0.057}$	$0.040_{\pm 0.052}$
	Things Are Worthwhile	Spearman ρ ↑	$0.131_{\pm 0.056}$	$\mathbf{0.156_{\pm 0.053}}$	$0.144_{\pm 0.057}$	$0.120_{\pm 0.053}$	$0.065_{\pm 0.052}$	$0.021_{\pm 0.058}$	$0.144_{\pm 0.054}$	$0.119_{\pm 0.059}$
	Domain Avg. Rank		2.40	3.20	**2.20**	5.00	5.40	6.60	5.40	5.80
	Domain Skill S (%)		$0.0_{\pm 0.0}$	$-5.4_{\pm 3.9}$	$\mathbf{+0.7_{\pm 3.1}}$	$-13.3_{\pm 4.6}$	$-14.4_{\pm 4.8}$	$-18.7_{\pm 4.9}$	$-16.7_{\pm 4.5}$	$-15.4_{\pm 4.9}$
eep & lifestyle	Bedtime	Spearman ρ ↑	$0.175_{\pm 0.058}$	$0.166_{\pm 0.059}$	$\mathbf{0.189_{\pm 0.059}}$	$0.025_{\pm 0.057}$	$-0.024_{\pm 0.055}$	$0.102_{\pm 0.059}$	$0.011_{\pm 0.055}$	$0.050_{\pm 0.056}$
	Currently Employed	AUPRC ↑	$0.869_{\pm 0.023}$	$\mathbf{0.934_{\pm 0.013}}$	$0.916_{\pm 0.016}$	$0.856_{\pm 0.022}$	$0.907_{\pm 0.018}$	$0.842_{\pm 0.021}$	$0.879_{\pm 0.021}$	$0.772_{\pm 0.027}$
	Sleep Duration	Spearman ρ ↑	$0.089_{\pm 0.051}$	$0.109_{\pm 0.050}$	$\mathbf{0.114_{\pm 0.050}}$	$-0.010_{\pm 0.051}$	$0.064_{\pm 0.050}$	$0.045_{\pm 0.050}$	$-0.026_{\pm 0.050}$	$-0.003_{\pm 0.049}$
	Vigorous Activity Minutes	Pearson r ↑	$0.300_{\pm 0.072}$	$0.208_{\pm 0.055}$	$\mathbf{0.377_{\pm 0.057}}$	$0.192_{\pm 0.055}$	$0.213_{\pm 0.066}$	$0.210_{\pm 0.062}$	$0.215_{\pm 0.068}$	$-0.015_{\pm 0.063}$
	Wake-up Time	Spearman ρ ↑	$0.079_{\pm 0.052}$	$0.100_{\pm 0.058}$	$\mathbf{0.118_{\pm 0.053}}$	$-0.048_{\pm 0.054}$	$0.029_{\pm 0.038}$	$-0.090_{\pm 0.049}$	$0.028_{\pm 0.045}$	$-0.014_{\pm 0.052}$
	Domain Avg. Rank		3.00	2.80	**1.20**	6.60	4.60	5.80	5.40	6.60
	Domain Skill S (%)		$0.0_{\pm 0.0}$	$+11.3_{\pm 4.4}$	$\mathbf{+12.1_{\pm 2.9}}$	$-13.9_{\pm 5.4}$	$-1.5_{\pm 4.6}$	$-13.3_{\pm 5.5}$	$-8.2_{\pm 5.2}$	$-27.7_{\pm 6.0}$
	Macro Avg. Rank		3.19	2.91	**2.29**	5.94	4.14	5.79	5.29	6.46
	Overall Skill S (%)		$0.0_{\pm 0.0}$	$+17.6_{\pm 1.6}$	$\mathbf{+18.8_{\pm 1.2}}$	$-20.1_{\pm 2.6}$	$-10.3_{\pm 2.3}$	$-19.1_{\pm 2.7}$	$-17.2_{\pm 2.7}$	$-29.8_{\pm 2.8}$

E.4.1. Gemini Vision: Rendering and Prompt Template

This subsection documents the Vision-mode rendering and prompt template used by the GEM-2.5 VIS and GEM-3.1 VIS probes. Per-task results across all three Gemini-family probe configurations (Statistics, Vision, and Agentic) are reported in Appendix E.4.

Rendering wearable weeks as images. Each (168, 19) weekly tensor is rendered as a single stacked-subplot figure, following the multivariate layout of He et al. [2025]: the seven continuous channels (iPhone and Apple Watch step counts and distances, flights climbed, heart rate, active energy) appear as line plots on top, and the twelve binary activity/sleep indicators as filled "active" strips below. We omit binary channels with no positive samples in the week to keep the figure compact. We use minimal axis decoration (no tick marks, frames, or legend blocks), but retain a small per-channel title of the form `"<channel> (<unit>) [vmin – vmax]"` so the VLM can identify each row and recover absolute scale (we set per-channel y-limits rather than a shared global range). Each channel is drawn in a unique, fixed color, following the per-channel coloring scheme of Liu et al. [2025]. Gray vertical lines every 24 hours mark day boundaries. The accompanying prompt (Figure 11) pairs the image with a fixed channel-legend block describing the layout, a per-task description with the index-to-label map, and a forced `Answer: <...>` suffix for parsing. We rely on the cross-VLM ablation in He et al. [2025, Appendix B], which finds the rendering transfers across GPT-4o, Claude 3.5, Gemini 2.0, and Qwen-2.5-72B, to motivate applying the same design to Gemini 3.1 Pro.

Wearable VLM Prompt

You are shown one image and asked to make a single prediction. The image will contain:

- A plot of one week (168 hours) of wearable sensor data from a single participant.
- **Top panels:** seven continuous signals as line plots – iPhone step count, iPhone distance, flights climbed, Apple Watch step count, Apple Watch distance, heart rate (bpm), and active energy (cal/hr). Each panel title states the channel name, unit, and observed value range `[vmin–vmax]`.
- **Bottom panels:** binary activity and sleep indicators rendered as colored strips (filled = active). Channels with no activity in the week are omitted.
- Faint vertical gray lines mark day boundaries every 24 hours.

Task. ⟨*task description*⟩

- *Binary (BiologicalSex):*
  `Predict the person's biological sex.\n 0=Female, 1=Male`
- *Ordinal (BMI_categories):*
  `Predict the person's BMI category.\n 0=Underweight, 1=Normal weight, 2=Overweight, 3=Obese`
- *Regression (age):*
  `Predict the person's age in years.`

Response format. End your response with a single line of the form:

`Answer: ⟨one of the labels above, or a numeric value⟩`

Figure 11 | Prompt template used for the Gemini VLM zero-shot evaluation.

F. Imputation Tasks

Wearable sensor data is inherently incomplete: devices are removed, batteries die, sensors malfunction, and physiological signals drop out during specific activities. Any model operating on such data must either tolerate or recover from missingness. We define a structured imputation benchmark that evaluates methods across six masking approaches grounded in real-world failure modes, using both continuous and binary reconstruction metrics. We provide two primary tasks for imputation. First, a *single-day imputation* task. Here a model has to impute based on a single day of passive data. Second, a *long-context imputation* task, where the goal is to impute using longer context windows. While we do not limit the context window for the public benchmark, for practical purposes in this evaluation we limited neural models to 7-day windows. Since imputation is particularly useful if it can be applied for higher-frequency data such that the resulting imputed data is broadly usable, we set this task to take place on minute-resolution passive data.

We denote an imputation sample as $\mathbf{X} \in \mathbb{R}^{C \times T}$ with C channels and T time steps (minutes). For single-day methods, $T = 1{,}440$; for 7-day methods, $T = 10{,}080$. The original validity mask $\mathbf{M} \in \{0, 1\}^{C \times T}$ indicates positions with observed data ($M_{c,t} = 1$). Each masking approach produces an artificial mask $\mathbf{A} \in \{0, 1\}^{C \times T}$ where $A_{c,t} = 1$ marks positions to be imputed, with $\mathbf{A} \leq \mathbf{M}$ element-wise (only observed positions can be masked). Data preprocessing for imputation follows the general benchmark pre-processing pipeline outlined in Appendix D.3.

F.1. Masking Approaches

We organize masking approaches into two tiers. *Structural* masks simulate generic data-collection failures, while *semantic* masks target physiologically meaningful gaps tied to specific activities or sensor limitations (see Figure 12).

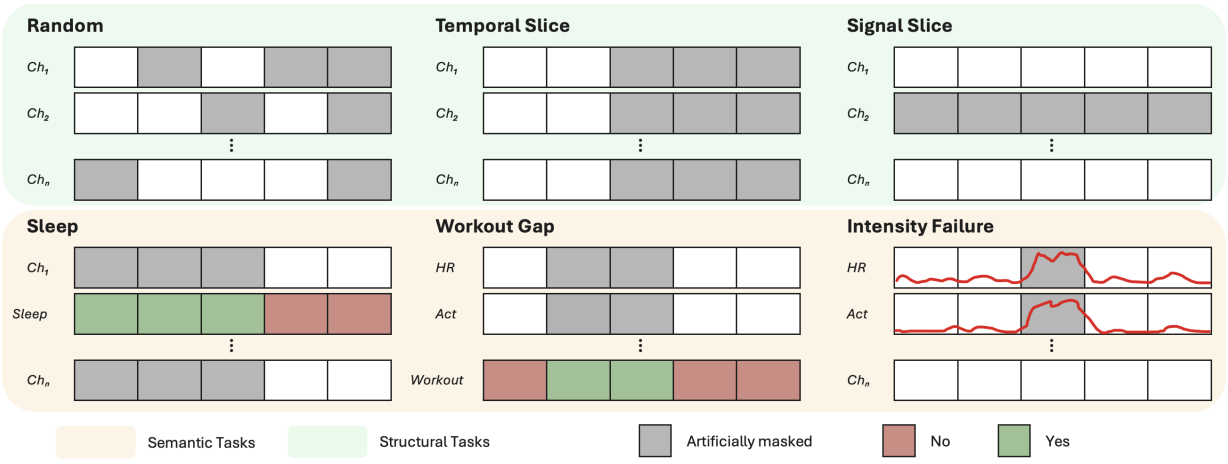

Figure 12 | **Imputation task structure.** Six masking approachs on daily minute-level multi-channel wearable tensors, organized into *structural* (top, signal-agnostic) and *semantic* (bottom, event-driven) tiers. Gray cells mark artificially masked positions on which methods are scored.

The pre-defined imputation masks for all imputation tasks are pre-defined. We generate the masks across the provided user split and cap each user to 91 days in the test set for imputation evaluation. The rational for 91-days is two-fold, first, we want to avoid a few superusers with 10 years of data to dominate the results, and second, performance, this makes running the evaluation pipeline in a reasonable amount of time possible. When comparing performance to the full run we found only minor differences and most of those were due to long-term users being over-represented.

Tier 1: Structural masks.

- **Random noise.** Short channel-local bursts are masked to simulate sporadic transmission errors or transient sensor glitches. We partition the tensor into non-overlapping patches of $p = 30$ consecutive minutes per channel, yielding $C \cdot \lfloor T/p \rfloor$ candidate patches. Candidates are randomly permuted and greedily selected until the number of newly masked valid positions reaches $r \cdot \|\mathbf{M}\|_1$ with mask ratio $r = 0.5$.
- **Temporal slice.** Contiguous blocks of time are masked across all channels simultaneously, simulating periods where the device was removed or powered off. Block sizes are drawn uniformly from $[b_{\min}, b_{\max}] = [30, 60]$ minutes. The number of blocks is estimated as

$$n_b = \left\lceil \frac{\log(1-r)}{\log\left(1 - \bar{b}/T\right)} \right\rceil, \quad \bar{b} = \frac{b_{\min} + b_{\max}}{2}, \tag{9}$$

  with target mask ratio $r = 0.25$. Each block start is sampled uniformly from valid time steps, and the resulting timestep mask is broadcast to all channels: $A_{c,t} = M_{c,t} \cdot \mathbf{1}[t \in \text{masked blocks}]$.
- **Signal slice.** Entire sensor channels are masked for the full observation window, simulating complete sensor failure or a missing modality. With equal probability, one of two modes is selected:
  - *Mode A (individual channels):* $\lceil r \cdot |C_{\text{valid}}| \rceil$ channels are sampled uniformly without replacement from the set of channels with any observed data, with $r = 0.5$.
  - *Mode B (device group):* One device group (e.g., all watch channels or all phone channels) is selected uniformly at random, and all its channels are masked.

  For each selected channel c, we set $A_{c,:} = M_{c,:}$.

Tier 2: Semantic masks.

- **Sleep gap.** All channels are masked during detected sleep periods, simulating the common behavior of removing a wearable device before bed. Sleep is detected at each time step as

$$s_t = \mathbf{1}\left[X_{\text{asleep},t} > 0\right] \vee \mathbf{1}\left[X_{\text{inbed},t} > 0\right], \tag{10}$$

  where X_{asleep} and X_{inbed} denote the corresponding activity indicator channels. For all time steps where $s_t = 1$, every channel except the two sleep indicators is masked: $A_{c,t} = M_{c,t} \cdot s_t$ for $c \notin \{\text{asleep}, \text{inbed}\}$.
- **Workout gap.** Continuous sensor channels are masked during detected workout periods, simulating motion-artifact–induced signal dropout during vigorous exercise. A workout is detected at time t when any of the binary workout-type channels is active:

$$w_t = \bigvee_{c \in C_{\text{workout}}} \mathbf{1}\left[X_{c,t} > 0\right], \tag{11}$$

  where C_{workout} spans 10 activity-type channels (walking, running, cycling, etc.). Only heart rate and active energy burned channels are masked during detected workouts; binary channels are not affected.
- **Intensity failure.** Continuous sensor channels are masked during high-intensity activity intervals, simulating sensor saturation or clipping at elevated physiological loads. High intensity is detected where heart rate exceeds a threshold $\tau = 160$ BPM. Only contiguous runs of ≥ 5 consecutive high-intensity minutes qualify:

$$h_t = \mathbf{1}\left[X_{\text{HR},t} > \tau\right], \quad \mathcal{R} = \left\{(t_s, t_e) : h_t = 1 \; \forall\, t \in [t_s, t_e],\; t_e - t_s \geq 4\right\}. \tag{12}$$

  Heart rate and active energy burned channels are masked at all time steps belonging to qualifying runs $\mathcal{R}$. Binary channels are not affected.

Table 18 | Overview of masking approaches. "Channels masked" indicates whether continuous (C), binary (B), or both channel types are affected.

Approach	Real-world failure mode	Tier	Channels masked	Key parameters
Random noise	Sporadic transmission errors	Structural	C + B	r=0.5, p=30
Temporal slice	Device removal or power-off	Structural	C + B	r=0.25, $b \in [30, 60]$
Signal slice	Complete sensor/modality failure	Structural	C + B	r=0.5
Sleep gap	Device removed during sleep	Semantic	C + B	data-driven
Workout gap	Motion-artifact dropout in exercise	Semantic	C only	data-driven
Intensity failure	Sensor saturation at high intensity	Semantic	C only	τ=160 BPM, $\geq$ 5 min

Table 18 summarizes the approaches, their real-world motivation, and key parameters.

F.2. Imputation Methods Overview

We evaluate seventeen imputation methods in total. Among the deep learning architectures, BRITS, DLinear, FEDformer, and TimesNet are implemented via the PyPOTS library [Du, 2023]. For BRITS, we implement gradient clipping (maximum norm of 1.0) to prevent training instability and numerical overflow caused by large loss spikes. While we explored tuning the MIT/ORT weights for DLinear, this did not yield significant improvements; consequently, due to computational resource constraints, we opted not to extend this tuning to the other PyPots neural imputers. Finally, we excluded certain widely used classic imputation methods, such as MICE [Azur et al., 2011], for three primary reasons: 1) they do not scale efficiently to large-scale, high-frequency (minute-level) multivariate wearable time-series data; 2) traditional statistical approaches have been shown to perform relatively poorly on this specific data modality compared to simple methods like linear interpolation [Toye et al., 2025, Xu et al., 2025b]; and 3) their use would violate theoretical assumptions regarding the underlying missingness-generating process [Xu et al., 2025b].

F.2.1. Single-day methods

The following methods operate solely on a single-day sample without access to further user history.

Statistical baselines.

- **Mean**: replaces each masked entry with the per-channel training-set mean.
- **Mode**: replaces each masked entry with the per-channel training-set mode (primarily relevant for binary channels).
- **Linear interpolation**: interpolates linearly between the nearest observed values in each channel along the time axis.
- **Last observation carried forward (LOCF)**: fills each masked entry with the most recent observed value in the same channel.
- **Temporal mean**: computes per-channel, per-minute-of-day means from the training set by folding multi-day windows into a standard 24-hour (1440-minute) diurnal profile via modular indexing (t mod 1440), capturing population-level circadian patterns such as resting heart rate at night or step-count peaks during the day. Falls back to the global channel mean for (channel, minute) pairs with no observations.
- **Temporal mode**: analogous to temporal mean but uses the per-channel, per-minute-of-day mode (most frequent rounded value), primarily relevant for binary channels whose activity patterns follow regular diurnal schedules.

Learned models.

- **BRITS** [Cao et al., 2018]: bidirectional RNN imputation model (Appendix F.6.2).
- **DLinear** [Zeng et al., 2023]: decomposition-linear model with trend/seasonality separation (Appendix F.6.2).
- **FEDformer** [Zhou et al., 2022]: Fourier-enhanced Transformer (Appendix F.6.2).
- **TimesNet** [Wu et al., 2023]: temporal 2D convolution model over reshaped time series (Appendix F.6.2).
- **LSM2** [Xu et al., 2025b]: masked autoencoder with adaptive and inherited masking (Appendix F.6.1).

F.2.2. Long-context methods

When a user's context data across multiple weeks is available, imputation can exploit individual-level patterns. The following methods operate on 7-day windows built within each evaluation split by sorting each user's daily samples chronologically and chunking them into non-overlapping windows. Calendar gaps are preserved rather than interpolated, incomplete tail windows are left-padded with sentinel days, and each day slot carries its true calendar-day offset relative to the first non-padded day. Methods impute the full 7-day tensor, but evaluation slices predictions back to real daily segments and scores only non-padded days with nonzero artificial masks.

Personalized statistical methods compute per-user fill values from that user's samples within the same evaluation split, falling back to population-level statistics when user-specific data is insufficient.

- **Personalized mean**: replaces each masked entry with the per-channel mean computed from the user's own context samples. Falls back to the population mean for channels where the user has no observations.
- **Personalized mode**: replaces each masked entry with the per-channel mode from the user's context samples. Falls back to the population mode.
- **Personalized temporal mean**: computes per-user, per-channel, per-minute-of-day means by folding the user's multi-week context into a personalized 24-hour diurnal profile. Applies a three-level fallback chain: user-specific minute mean → user-specific channel mean → population-level temporal mean.
- **DLinear (7-day)**: the single-day DLinear trained on concatenated 7-day inputs ($C \times 10{,}080$ time steps; Appendix F.6.2).
- **LSM2 (7-day)**: the daily LSM2 encoder–decoder applied to 7-day concatenated inputs with a coarser 60-minute patch size. This reduces the weekly token count to $19 \times 24 \times 7 = 3{,}192$, making dense self-attention tractable and providing a dense weekly baseline for LSM2-SPARSE. See Appendix F.6.1.
- **LSM2-Sparse (7-day)**: a two-stage masked autoencoder that extends the daily LSM2 to exploit multi-day context via a sparse cross-day decoder while retaining 10-minute per-day patches. Its cross-day attention layers consume the true calendar `day_offsets` for each window and use RoPE to distinguish consecutive from gappy histories. See Appendix F.6.1.

F.3. Raw Metrics

We calculate two complementary metrics evaluated exclusively on artificially masked positions. For each channel c, let $\mathcal{A}_c = \{t : A_{c,t} = 1\}$ denote the set of artificially masked time indices.

MAE. For continuous channels C_{cont}, we compute the per-channel mean absolute error over artificially masked positions:

$$\text{MAE}_c = \frac{1}{|\mathcal{A}_c|} \sum_{t \in \mathcal{A}_c} \left| \hat{x}_{c,t} - x_{c,t} \right|, \tag{13}$$

where $\hat{x}_{c,t}$ is the imputed value and $x_{c,t}$ the ground truth. We use MAE as the continuous error fed to the aggregate Skill Score and Average Rank; for binary channels the corresponding error is $1 - \text{ROC AUC}_c$ (defined next). Appendix F.4 details how these per-channel errors are aggregated into the reported scores.

Macro ROC AUC. For binary channels C_{bin}, we compute the area under the receiver operating characteristic curve per channel using continuous-valued predictions against binarized ground truth ($y_{c,t} = \mathbf{1}[x_{c,t} > 0.5]$). The macro average is:

$$\text{macro ROC AUC} = \frac{1}{|C^*_{\text{bin}}|} \sum_{c \in C^*_{\text{bin}}} \text{ROC AUC}_c, \tag{14}$$

where $C^*_{\text{bin}} \subseteq C_{\text{bin}}$ excludes channels with only a single class present in the masked ground truth. Workout gap and intensity failure mask only continuous channels and therefore have no ROC AUC scores. For aggregate scoring, each binary channel contributes the error $1 - \text{ROC AUC}_c$ (Appendix F.4).

Handling non-finite imputations. Any artificially masked cell that an imputer leaves non-finite (`NaN` or `Inf`), or fails to fill, is substituted before scoring with a per-channel fallback computed on the training split: the channel mean for continuous channels and the majority class for binary channels (the **Mean/Mode** baseline of Appendix F.2.1). This mirrors the forecasting track's NaN handling and surfaces a model's inability to impute in its score rather than silently dropping those cells; we report the resulting fallback substitution rate.

F.4. Aggregation and Scoring

Tasks, scopes, and channel categories. The headline imputation columns in Table 3 (the overall skill score S, the average rank, the fairness skill score S_{fair}, and the per-category Activity / Physiology / Sleep / Workout / Semantic columns) are produced by aggregating the per-channel errors of Section F.3 through the unified skill-score machinery of Appendix B. We reuse the base clipping, geometric-mean, fairness, and bootstrap definitions given there (clip bounds $\ell = 0.01$, $u = 100$) and specialize them to the imputation track below, where channels split very unevenly across categories and the aggregation is therefore *category-balanced*: the four reporting categories act as equal-weight buckets in a hierarchical mean rather than being pooled as a flat set of per-channel tasks. Throughout, $p \in \mathcal{P}$ indexes participants, a task $r = (s, c)$ is a (approaches, channel) pair, m is the evaluated method, and b = LOCF is the reference baseline.

The unit of evaluation is a *task* $r = (s, c)$: a single channel c scored under a single masking approach s. A *scope* is a collection of tasks aggregated into one reported number; each column of Table 3 corresponds to a scope, defined by a set of masking approaches $\mathcal{S}_{\text{scope}}$ together with the channel buckets scored within them. The $C = 19$ channels partition into the continuous channels C_{cont} (seven channels) and the binary channels C_{bin} (twelve channels), and are further grouped into four reporting categories that serve as aggregation *buckets*: *Activity* (five continuous channels—phone and watch step counts and distances, and flights climbed), *Physiology* (two continuous channels—heart rate and active energy), *Sleep* (two binary channels—the asleep and in-bed indicators), and *Workout*

(the ten binary workout-type indicators). The *structural* approaches (random noise, temporal slice, signal slice) and *semantic* approaches (sleep gap, workout gap, intensity failure) are those defined in Section F.1.

Per-participant error. For each task we first reduce all participants artificially masked cells to a single error. For a continuous task ($c \in C_{\text{cont}}$) we pool the absolute errors into E^{cont} (superscript cont: *continuous*); for a binary task ($c \in C_{\text{bin}}$) we use that participant's pooled ROC AUC to form $\tilde{E}^{\text{bin}}$ (bin: *binary*):

$$E^{\text{cont}}_{m,p,r} = \frac{1}{N_{p,r}} \sum_{\text{masked cells}} |\hat{x} - x|, \qquad \tilde{E}^{\text{bin}}_{m,p,r} = \max\left(1 - \text{ROC AUC}_{m,p,r},\ \varepsilon\right), \tag{15}$$

where the sum runs over participant p's artificially masked cells ($A_{c,t} = 1$) for task r, pooled over that participant's daily samples; $N_{p,r}$ is their count and $\varepsilon = 0.005$. Participants whose binary task is single-class (an undefined AUC) are dropped. The tilde in $\tilde{E}$ marks the ε-floored binary error, which enters only the paired-ratio path (17); the unfloored $1 - \text{ROC AUC}$ (written without a tilde) is used for the average rank (19).

Collapsed binary categories. The reported Sleep and Workout columns, and the binary side of the overall column, collapse each binary category $\kappa \in \{\text{Sleep}, \text{Workout}\}$ into a single per-participant error per approach (superscript coll: *collapsed*), averaging the per-channel binary errors over the category's channels that have a defined AUC for that participant and then applying the ε floor once to the resulting mean:

$$\tilde{E}^{\text{coll}}_{m,p,(s,\kappa)} = \max\left(\frac{1}{|\kappa_p|} \sum_{c \in \kappa_p} \left(1 - \text{ROC AUC}_{m,p,(s,c)}\right),\ \varepsilon\right), \tag{16}$$

where $\kappa_p \subseteq \kappa$ is the set of channels in category κ with a defined AUC for participant p. Each structural approach therefore contributes one Sleep task and one Workout task rather than two and ten per-channel tasks; this equal weighting prevents the ten workout channels from dominating the two sleep channels and the seven continuous channels. Continuous categories are not collapsed.

Per-task paired ratio. Let $E^{\star}_{m,p,r}$ denote the per-participant error used for task r: E^{cont} for a continuous task, $\tilde{E}^{\text{bin}}$ for a per-channel binary task, and $\tilde{E}^{\text{coll}}$ for a collapsed-category task from (15) and (16). For each task the ratio against the baseline is formed *within* each participant and then geometrically averaged over participants (it is not a ratio of pooled errors):

$$R_{m,r} = \exp\left(\frac{1}{|\mathcal{P}_r|} \sum_{p \in \mathcal{P}_r} \log \text{clip}\left(E^{\star}_{m,p,r} / E^{\star}_{b,p,r},\ \ell, u\right)\right), \tag{17}$$

where $\mathcal{P}_r$ is the set of participants for whom both the method and baseline errors are defined and finite with $E^{\star}_{b,p,r} > 0$.

Skill score per scope. The skill score is a category-balanced hierarchical geometric mean over the buckets introduced above. Within a making approach s, let $\mathcal{K}_s$ be the categories with at least one scored task, and let $\mathcal{T}_{s,k}$ be the tasks of bucket k in approach s—the continuous per-channel tasks for Activity and Physiology, and the single collapsed task (s, κ) of (16) for Sleep ($\kappa = \text{Sleep}$) and Workout ($\kappa = \text{Workout}$). Per-channel binary tasks never enter directly; a binary category reaches a scope only

through its collapsed task. For a scope spanning the approach set $\mathcal{S}_{\text{scope}}$,

$$S_{m,\text{scope}} = 1 - \exp\left(\frac{1}{|\mathcal{S}_{\text{scope}}|}\sum_{s\in\mathcal{S}_{\text{scope}}}\frac{1}{|\mathcal{K}_s|}\sum_{k\in\mathcal{K}_s}\frac{1}{|\mathcal{T}_{s,k}|}\sum_{r\in\mathcal{T}_{s,k}}\log\text{clip}(R_{m,r},\ell,u)\right). \tag{18}$$

Reading the three means from the inside out: within each (approach, bucket) pair we average the clipped log-ratios over the bucket's tasks, then average over the buckets present in the approach, then over the approaches in the scope. Each bucket therefore has equal voice within a approach and each approach equal voice in the scope, so neither Activity's five channels nor Workout's ten can dominate the aggregate.

Scopes reported in the main table. Each column fixes the approach set $\mathcal{S}_{\text{scope}}$ of (18):

- **Overall (S):** all six approaches. Within each approach the present buckets are averaged as in (18); the Activity and Physiology buckets contribute wherever their channels are masked (all six approaches), while the Sleep and Workout buckets contribute only in the three structural approaches, since binary channels are excluded from the semantic approaches.
- **Activity / Physiology:** a single bucket over the three structural approaches—equivalently a per-channel geometric mean over the structural approaches crossed with the five Activity channels (15 tasks) or the two Physiology channels (6 tasks).
- **Sleep / Workout:** the single collapsed bucket over the three structural approaches, i.e. a geometric mean over the collapsed tasks $\{(s,\kappa) : s \text{ structural}\}$ for $\kappa \in \{\text{Sleep}, \text{Workout}\}$ (3 tasks each; (16)).
- **Semantic:** the three semantic approaches. Only continuous buckets are present (binary channels are excluded from these approaches), so each approach reduces to its Activity and/or Physiology buckets.

The per-category Activity, Physiology, Sleep, and Workout columns are restricted to the structural approaches, whereas the Overall column spans all six; the binary categories are scored only on the structural approaches, so they reach the Overall column through exactly the structural collapsed tasks that define the Sleep and Workout columns.

Channel restriction. Only artificially masked channels are scored: the pairing step emits records only at masked positions, so unmasked channels contribute no tasks. In addition, binary channels are excluded from the three semantic approaches (sleep gap, workout gap, intensity failure). The net scored channels are the two Physiology channels (heart rate and active energy) for workout gap and intensity failure, and all seven continuous channels for sleep gap.

Average rank. For each task, methods are ranked by their per-participant error—E^{cont} for a continuous task, and the *unfloored* 1 − ROC AUC (per channel, or its per-category mean for a collapsed task) for a binary task—in ascending order with ties averaged. The per-participant ranks of method m are averaged into a task rank $\bar{\rho}_{m,r}$, which is then reduced to the scope through the *same* category-balanced hierarchy as the skill score, but with arithmetic means at every level:

$$\bar{\rho}_{m,r} = \frac{1}{|\mathcal{P}_r|}\sum_{p}\text{rank}_p\left(E_{\cdot,p,r}\right), \qquad \rho_{m,\text{scope}} = \frac{1}{|\mathcal{S}_{\text{scope}}|}\sum_{s\in\mathcal{S}_{\text{scope}}}\frac{1}{|\mathcal{K}_s|}\sum_{k\in\mathcal{K}_s}\frac{1}{|\mathcal{T}_{s,k}|}\sum_{r\in\mathcal{T}_{s,k}}\bar{\rho}_{m,r}, \tag{19}$$

where $E_{\cdot,p,r}$ is the vector of these per-participant errors across the competing methods and rank_p returns method m's position within it. Ranks are computed on errors, not on skill scores; the reported average rank uses the overall scope (all six approaches), as S does.

Fairness skill score. We follow the disparity-ratio fairness skill score of Appendix B.2, applying the same category-balancing used for the skill score. Let $\mathcal{G}$ be a sensitive attribute (here age group or sex) with mutually exclusive subgroups $g \in \mathcal{G}$, and let $E^{(g)}_{m,r}$ be method m's error on task r restricted to the participants in subgroup g—aggregated participant-micro (15) and averaged over that subgroup's participants, with the binary side collapsed per category (16). The per-task disparity is the average gap across pairs of subgroups, $D^{(\mathcal{G})}_{m,r} = \frac{1}{|\mathcal{G}|(|\mathcal{G}|-1)} \sum_{g,g' \in \mathcal{G}, g \neq g'} \left| E^{(g)}_{m,r} - E^{(g')}_{m,r} \right|$, evaluated over the subgroups common to method and baseline (tasks with fewer than two common subgroups are dropped). The per-attribute fairness skill score balances the per-task disparity ratios by category, but—unlike the skill and rank scopes—pools each bucket over *all six* approaches instead of adding a per-approach level:

$$S^{(\mathcal{G})}_{\text{fair}} = 1 - \exp\left(\frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \frac{1}{|\mathcal{T}_k|} \sum_{r \in \mathcal{T}_k} \log \text{clip}\left(D^{(\mathcal{G})}_{m,r} / D^{(\mathcal{G})}_{b,r},\ \ell, u \right) \right), \tag{20}$$

where $\mathcal{K}$ is the set of categories present and $\mathcal{T}_k$ collects all (approach, channel) tasks of bucket k across the six approaches, with the same bucket sourcing as the skill score (Activity and Physiology from continuous per-channel tasks, Sleep and Workout from their collapsed tasks, per-channel binary excluded). The reported S_{fair} averages over the two attributes, $S_{\text{fair}} = \frac{1}{2}\left(S^{(\text{age\_group})}_{\text{fair}} + S^{(\text{sex})}_{\text{fair}} \right)$; we report it at the overall scope as well as per attribute.

Bootstrap confidence intervals. All reported scores carry participant-level bootstrap confidence intervals (CIs). We resample participants with replacement using a single shared, seeded draw matrix per split ($B = 1000$ replicates). Each participant's contribution is precomputed once—the pooled absolute errors for a continuous channel, and a single ROC AUC for a binary channel—and every draw re-aggregates these per-participant statistics through the full skill / rank / fairness pipeline. We report each score as its point estimate (computed on the held-out test split) together with a 95% bootstrap CI: for the skill score and average rank this is the percentile interval, while the fairness skill score uses the bias-corrected and accelerated (BCa) interval of Appendix B.2, anchored at the deterministic point estimate.

F.5. Imputation Results

Here we present additional results of the imputation evaluation. Results for *Single-day imputation* (Section F.5.1) uses only the current daily sample, while *long-context imputation* (Section F.5.2) additionally leverages a user's historical data as well as any additional information that could be leveraged.

Table 19 complements the main-paper summary in Table 3 by listing all six masking approaches explicitly. We additionally report raw approach-level metrics for interpretability: MAE for continuous channels and macro ROC AUC for binary channels—the same per-channel errors that feed the aggregate skill score and average rank (Appendix F.4).

F.5.1. Single-day imputation

Among single-day methods, LSM-2 retains the strongest overall per-approach profile, especially on the structural approaches, while the constant mode-based baselines remain hardest to beat on the sleep gap. Table 20 lists the raw MAE and ROC AUC values for every approach.

Table 19 | **Imputation Results by Masking Scenario.** Aggregate Skill Score S (in %; 0 =LOCF reference), Average Rank R, Fairness Skill Score S_{fair} (disparity-ratio; see Appendix B.2), and per-scenario Skill Scores across all six masking scenarios (lower is better for R; higher otherwise). Single-day methods above; long-context methods ($\geq 7 \times 1440$ time steps) below. Gradients computed within each track. Values are point estimates on the held-out test split; sub/superscripts give the 95% bootstrap confidence interval (B=1000): the percentile interval for every column except S_{fair}, which uses the bias-corrected and accelerated (BCa) interval.

Method	$S\uparrow$	$R\downarrow$	$S_{\text{fair}}\uparrow$	Random noise ↑	Temporal slice ↑	Signal slice ↑	Sleep gap ↑	Workout gap ↑	Intensity failure ↑
Single-day imputation									
Statistical Models									
Linear	$+21.5^{+0.7}_{-1.2}$	$7.0^{+0.1}_{-0.1}$	$+34.7^{+11.6}_{-6.5}$	$+47.1^{+1.3}_{-2.0}$	$+56.8^{+0.9}_{-2.2}$	$+0.0^{+0.0}_{-0.0}$	$-4.1^{+2.4}_{-2.9}$	$+15.1^{+1.3}_{-1.1}$	$\mathbf{-15.9}^{+5.2}_{-5.7}$
LOCF *(reference)*	0.0	$8.4^{+0.1}_{-0.1}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Temporal mode	$-6.2^{+2.4}_{-2.3}$	$10.0^{+0.1}_{-0.1}$	$+55.9^{+11.2}_{-12.7}$	$-34.1^{+3.4}_{-3.0}$	$-53.2^{+4.7}_{-4.7}$	$+49.9^{+0.8}_{-0.8}$	$\mathbf{+75.1}^{+1.6}_{-2.0}$	$-25.7^{+3.0}_{-2.9}$	$-346.7^{+38.9}_{-40.6}$
Mode	$-27.3^{+2.7}_{-2.5}$	$10.6^{+0.1}_{-0.1}$	$+91.2^{+0.7}_{-0.8}$	$-90.6^{+4.8}_{-4.3}$	$-124.5^{+7.3}_{-6.8}$	$+29.2^{+0.6}_{-0.6}$	$+74.4^{+1.6}_{-1.9}$	$-23.5^{+3.0}_{-2.9}$	$-343.9^{+38.9}_{-40.2}$
Temporal mean	$-31.2^{+3.1}_{-3.2}$	$10.4^{+0.1}_{-0.1}$	$-28.9^{+44.9}_{-45.3}$	$-16.2^{+3.6}_{-3.5}$	$-31.9^{+4.6}_{-5.4}$	$+53.7^{+1.3}_{-1.3}$	$-51.6^{+10.6}_{-13.3}$	$-10.3^{+2.3}_{-2.1}$	$-329.8^{+31.3}_{-34.7}$
Mean	$-119.7^{+4.7}_{-4.4}$	$13.4^{+0.0}_{-0.0}$	$\mathbf{+92.2}^{+0.5}_{-0.7}$	$-147.1^{+6.5}_{-5.9}$	$-191.8^{+9.9}_{-9.5}$	$+0.0^{+0.0}_{-0.0}$	$-212.9^{+21.5}_{-25.7}$	$-14.0^{+2.3}_{-2.2}$	$-336.8^{+32.5}_{-35.5}$
Neural Models									
LSM-2 Xu et al. [2025b]	$\mathbf{+61.4}^{+0.5}_{-1.2}$	$\mathbf{3.8}^{+0.1}_{-0.1}$	$+57.6^{+9.6}_{-8.2}$	$\mathbf{+81.1}^{+0.1}_{-1.0}$	$\mathbf{+61.1}^{+0.6}_{-2.0}$	$\mathbf{+86.7}^{+0.4}_{-0.6}$	$+44.7^{+3.6}_{-4.1}$	$\mathbf{+53.6}^{+1.4}_{-1.6}$	$-32.7^{+8.1}_{-9.6}$
BRITS Cao et al. [2018]	$+6.8^{+1.8}_{-1.9}$	$7.8^{+0.1}_{-0.1}$	$-30.3^{+30.4}_{-30.0}$	$+8.4^{+2.6}_{-3.1}$	$-10.7^{+3.6}_{-4.1}$	$+45.1^{+1.8}_{-1.7}$	$-31.5^{+8.0}_{-9.0}$	$+38.1^{+2.0}_{-2.1}$	$-45.1^{+8.8}_{-10.2}$
DLinear Zeng et al. [2023]	$-5.7^{+2.1}_{-2.1}$	$8.2^{+0.1}_{-0.1}$	$+30.1^{+12.9}_{-6.4}$	$+10.7^{+2.8}_{-3.0}$	$+18.0^{+2.8}_{-3.0}$	$+38.5^{+1.9}_{-1.8}$	$-20.4^{+8.0}_{-10.0}$	$-9.4^{+2.4}_{-2.3}$	$-135.7^{+15.2}_{-17.7}$
FEDformer Zhou et al. [2022]	$-53.7^{+3.3}_{-3.0}$	$11.3^{+0.1}_{-0.1}$	$+35.4^{+20.1}_{-9.4}$	$-84.2^{+4.8}_{-4.5}$	$-97.1^{+7.0}_{-6.8}$	$+23.0^{+0.5}_{-0.5}$	$-1.9^{+6.7}_{-8.3}$	$-22.5^{+2.6}_{-2.5}$	$-277.5^{+29.2}_{-30.2}$
TimesNet Wu et al. [2023]	$-66.0^{+3.5}_{-3.5}$	$10.9^{+0.1}_{-0.1}$	$+6.2^{+27.3}_{-17.4}$	$-56.2^{+5.3}_{-5.0}$	$-131.7^{+8.8}_{-8.6}$	$+31.2^{+2.1}_{-2.1}$	$-139.2^{+16.5}_{-19.5}$	$-3.6^{+2.3}_{-2.2}$	$-238.7^{+24.1}_{-26.4}$
Long-context imputation ($\geq 7 \times 1440$ ***time steps)***									
Statistical Models									
Pers. temp. mean	$-7.7^{+2.8}_{-2.8}$	$9.1^{+0.1}_{-0.1}$	$-50.7^{+35.7}_{-67.6}$	$-7.3^{+4.0}_{-4.0}$	$-20.4^{+5.1}_{-5.5}$	$+63.8^{+1.1}_{-1.2}$	$+16.1^{+5.8}_{-6.6}$	$-1.0^{+2.0}_{-2.0}$	$-294.3^{+28.7}_{-32.4}$
Pers. mode	$-26.1^{+2.6}_{-2.4}$	$10.5^{+0.1}_{-0.1}$	$\mathbf{+76.4}^{+4.7}_{-5.5}$	$-89.8^{+4.8}_{-4.4}$	$-123.5^{+7.2}_{-6.8}$	$+29.7^{+0.6}_{-0.6}$	$\mathbf{+76.0}^{+1.6}_{-1.9}$	$-25.2^{+2.8}_{-2.8}$	$-348.4^{+38.0}_{-40.7}$
Pers. mean	$-114.1^{+4.4}_{-4.3}$	$13.3^{+0.1}_{-0.1}$	$-26.7^{+37.2}_{-26.2}$	$-160.3^{+6.8}_{-6.0}$	$-207.7^{+10.0}_{-9.7}$	$+4.3^{+1.0}_{-1.0}$	$-166.2^{+18.4}_{-21.2}$	$-11.0^{+2.1}_{-2.1}$	$-325.2^{+31.4}_{-35.0}$
Neural Models									
LSM-2-Sparse (7-day)	$\mathbf{+64.7}^{+0.4}_{-1.2}$	$\mathbf{3.3}^{+0.1}_{-0.1}$	$+68.2^{+6.0}_{-1.7}$	$\mathbf{+82.4}^{+0.1}_{-1.0}$	$\mathbf{+64.5}^{+0.5}_{-2.0}$	$\mathbf{+89.0}^{+0.2}_{-0.6}$	$+48.9^{+3.4}_{-3.9}$	$\mathbf{+55.6}^{+1.3}_{-1.4}$	$\mathbf{-23.4}^{+8.1}_{-9.7}$
LSM-2 (7-day)	$+46.9^{+1.0}_{-1.5}$	$5.5^{+0.1}_{-0.1}$	$+46.2^{+13.1}_{-6.3}$	$+64.9^{+0.8}_{-1.6}$	$+39.6^{+2.1}_{-2.7}$	$+86.2^{+0.4}_{-0.7}$	$+34.0^{+4.3}_{-5.3}$	$+44.1^{+1.5}_{-1.7}$	$-106.3^{+13.9}_{-17.1}$
DLinear (7-day) Zeng et al. [2023]	$-28.3^{+2.5}_{-2.6}$	$9.5^{+0.1}_{-0.1}$	$+10.2^{+25.0}_{-20.4}$	$-20.3^{+3.4}_{-3.5}$	$-4.5^{+3.8}_{-4.2}$	$+27.3^{+1.7}_{-1.8}$	$-64.9^{+11.3}_{-13.6}$	$-18.1^{+2.5}_{-2.4}$	$-150.1^{+16.0}_{-18.8}$

Table 20 | **Single-Day Imputation Raw Metrics.** Scenario-level raw metrics on the test split. Each MAE entry is the mean per-channel MAE across applicable continuous channels; each ROC AUC entry is the macro-average across applicable binary channels. Sleep gap, workout gap, and intensity failure mask only continuous channels, so ROC AUC is not reported there. Values are point estimates over the test cohort.

Method	Random MAE↓	Random AUC↑	Temporal MAE↓	Temporal AUC↑	Signal MAE↓	Signal AUC↑	Sleep MAE↓	Workout MAE↓	Intensity MAE↓
Statistical Models									
Linear	44.3	0.829	42.7	0.867	65.9	0.500	28.9	1512.3	1103.3
Temporal mean	61.5	0.688	62.7	0.680	60.6	0.690	24.8	1753.9	3299.8
LOCF *(reference)*	51.2	0.715	50.5	0.742	65.9	0.500	29.7	1537.4	**979.0**
Temporal mode	47.8	0.557	48.8	0.558	47.2	0.557	**6.7**	1922.3	3492.7
Mode	47.8	0.500	48.8	0.500	47.2	0.500	6.7	1922.3	3492.7
Mean	66.9	0.500	67.1	0.500	65.9	0.500	56.2	1796.2	3348.5
Neural Models									
LSM-2 Xu et al. [2025b]	**25.1**	**0.983**	**40.1**	**0.951**	**35.0**	**0.954**	8.0	**749.0**	1083.5
DLinear Zeng et al. [2023]	47.1	0.787	48.3	0.826	58.0	0.588	25.4	1571.0	2301.5
BRITS Cao et al. [2018]	81.0	0.691	150.4	0.674	44.2	0.583	54.2	1044.6	1115.4
TimesNet Wu et al. [2023]	60.1	0.589	62.1	0.528	60.8	0.554	53.0	1767.7	2996.4
FEDformer Zhou et al. [2022]	55.8	0.533	53.9	0.549	54.2	0.498	18.7	1784.8	2514.0

F.5.2. Long-context imputation

Long-context methods improve most consistently on the structural approaches, with LSM-2-Sparse giving the strongest raw MAE profile overall and the personalized temporal mean baseline remaining competitive on several structured gaps. Table 21 lists the corresponding raw metrics.

Table 21 | **Long-Context Imputation Raw Metrics.** Scenario-level raw metrics on the test split. Each MAE entry is the mean per-channel MAE across applicable continuous channels; each ROC AUC entry is the macro-average across applicable binary channels. Sleep gap, workout gap, and intensity failure mask only continuous channels, so ROC AUC is not reported there. Values are point estimates over the test cohort.

Method	Random MAE↓	Random AUC↑	Temporal MAE↓	Temporal AUC↑	Signal MAE↓	Signal AUC↑	Sleep MAE↓	Workout MAE↓	Intensity MAE↓
Statistical Models									
Pers. temp. mean	57.6	0.715	58.5	0.724	55.5	0.757	15.9	1694.8	3194.9
Pers. mean	62.7	0.196	62.9	0.201	60.9	0.244	49.6	1803.1	3343.4
Pers. mode	47.6	0.499	48.7	0.499	47.1	0.500	**6.6**	1922.2	3492.7
Neural Models									
LSM-2 (7-day)	37.0	0.954	48.8	0.917	37.4	0.946	9.8	856.4	1565.1
LSM-2-Sparse (7-day)	**25.1**	**0.985**	**40.7**	**0.960**	**33.4**	**0.960**	8.1	**694.0**	**1024.2**
DLinear (7-day) Zeng et al. [2023]	46.8	0.693	49.0	0.764	54.2	0.534	26.3	1612.4	2370.0

F.6. Imputation Models

F.6.1. LSM-2 Reimplementation and Adaptations

We provide a detailed description of our reimplementation of Google's LSM-2 masked autoencoder Xu et al. [2025b]. We implement three variants that differ in temporal scope and attention strategy: a single-day model (LSM-2) that mimicks the original as faithfuly as possible, a 7-day variant (LSM-2-Weekly), and a sparse 7-day model (LSM-2-Sparse). All share the same basic patch embedding and masking approaches; they differ in how attention is organized across days and in patch size.

Throughout, we denote the number of sensor channels as C, the per-day sequence length as L (minutes), the patch size as p (minutes), and the number of patches per channel per day as $T = L/p$. The total token count per day is $N_{\text{day}} = C \cdot T$.

LSM-2 (Daily).

- **Input and tokenization.** The daily model receives a single-day input $\mathbf{X} \in \mathbb{R}^{C \times L}$ with C=19 channels and L=1,440 minutes. Non-overlapping patches of p=10 minutes are extracted per channel and projected to d_{enc}-dimensional token embeddings via a shared 1D convolution:

$$\mathbf{h}_i = \text{Conv1D}\big(\mathbf{X}[\text{patch } i]\big) + \mathbf{e}_i^{\text{pos}}, \quad \mathbf{h}_i \in \mathbb{R}^{d_{\text{enc}}}, \tag{21}$$

  where $\mathbf{e}_i^{\text{pos}}$ is a fixed 2D sinusoidal positional embedding that encodes both the channel index and the intra-day time index of patch i, each axis contributing $d_{\text{enc}}/2$ dimensions. This produces $N_{\text{day}} = C \cdot T = 19 \times 144 = 2{,}736$ tokens per day.
- **Encoder and decoder.** The encoder is a 12-layer 1D (Vision) Transformer with d_{enc}=384 and 6 attention heads. The decoder uses 4 Transformer layers with d_{dec}=256 and 4 attention heads. Self-attention operates over all N_{day} tokens, giving a per-layer complexity of $O(N_{\text{day}}^2) = O\big((CT)^2\big)$.
- **Adaptive and Inherited Masking (AIM).** AIM jointly handles real-world missingness and self-supervised masking by combining two binary masks per token: the *inherited mask* M_i^I where $M_i^I = \mathbf{1}[\text{patch } i \text{ contains NaN}]$ from actual data gaps, and an *artificial mask* M_i^A generated by one of three strategies applied with equal probability:
  - *Random*: each patch masked independently with probability r=0.5.
  - *Temporal slice*: 50% of time indices masked across all channels.
  - *Sensor slice*: 50% of channels masked across all time steps.

  A priority score determines which tokens are physically removed from the sequence:

$$s_i = 100 \cdot (M_i^I \vee M_i^A) + \epsilon_i, \quad \epsilon_i \sim \text{Uniform}(0, 1). \tag{22}$$

Tokens are sorted by s_i in descending order; the $(1-\rho)\cdot N_{\text{day}}$ tokens with lowest priority (most likely observed) are retained, with ρ=0.5. Any retained token that is nonetheless masked ($M_i^I \vee M_i^A = 1$) receives an attention logit of $-\infty$, preventing it from being attended to while still receiving information for reconstruction.

- **Reconstruction loss.** The decoder predicts the raw patch values $\hat{\mathbf{x}}_i \in \mathbb{R}^p$ for all positions. The loss is computed only over artificially masked patches with valid ground truth:

$$\mathcal{L} = \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} w_{c(i)}\cdot \ell\big(\hat{\mathbf{x}}_i,\, \mathbf{x}_i\big), \quad \ell = \begin{cases}\text{MSE} & \text{if } c(i)\in C_{\text{cont}},\\ \text{BCE} & \text{if } c(i)\in C_{\text{bin}},\end{cases} \tag{23}$$

  where $\mathcal{S} = \{i : M_i^A = 1 \ \wedge \ M_i^I = 0\}$, $c(i)$ denotes the channel of patch i, $w_{c(i)}$ is an optional per-channel weight, $C_{\text{cont}} = \{0,\ldots,6\}$ are continuous channels (steps, distance, flights, exercise time, stand time, heart rate, active energy), and $C_{\text{bin}} = \{7,\ldots,18\}$ are binary channels (sleep stages, workout types). Binary channels use BCE with logit predictions. Because the Apple Watch does not record heart rate every minute, zero-valued minutes within a heart rate patch represent absent measurements rather than true observations; the MSE for heart rate patches is therefore computed only over non-zero minutes within each patch. All reported results were derived with equal channel loss weighting.

LSM-2-Weekly (Dense). LSM-2-Weekly applies the same encoder–decoder architecture to a 7-day concatenated input $\mathbf{X} \in \mathbb{R}^{C\times DL}$ with D=7 days. To keep the total token count tractable for dense self-attention, the patch size is increased to p'=60 minutes (6× coarser than the daily model). This yields $T' = L/p' = 24$ patches per channel per day and a total of

$$N_{\text{week}} = C\cdot D\cdot T' = 19\times 7\times 24 = 3{,}192 \text{ tokens.}$$

Full self-attention over the entire week gives a per-layer complexity of $O(N_{\text{week}}^2) = O\big((CDT')^2\big)$. This scales quadratically with the number of days D, meaning that extending the context window further would become prohibitively expensive. The coarser temporal resolution (60 vs. 10 minutes) also limits the model's ability to capture fine-grained temporal patterns.

LSM-2-Sparse. LSM-2-Sparse achieves multi-day context while preserving the fine-grained p=10-minute patch resolution of the daily model and allows for irregularly sampled days (thus missing days). It uses a two-stage architecture: a frozen per-day encoder (based on the regular daily model) followed by a sparse cross-day decoder.

- **Per-day Encoder.** Each of the D=7 daily slices $\mathbf{X}_d \in \mathbb{R}^{C\times L}$ is encoded independently by the daily LSM-2 encoder, producing $N_{\text{day}} = CT$ = 2,736 latent tokens per day. The encoder weights are loaded from the pre-trained daily checkpoint and frozen during weekly training; only the decoder parameters below are learned.
- **Sparse Cross-Day Decoder.** The decoder consists of 4 Transformer layers (d_{dec}=256, 4 heads) that alternate between two attention patterns:
  - **Day-local layers (layers 0, 2).** Standard self-attention restricted to tokens within a single day. Each day's $N_{\text{day}} = CT$ tokens are processed independently, giving per-layer complexity $O\big((CT)^2\big)$, identical to the daily model and independent of D.
  - **Cross-day window layers (layers 1, 3).** Tokens are regrouped into temporal windows that span all days to keep the memory layout contiguous and thus allow to keep the benefits of modern optimizations like FlashAttention out of the box. Per-day tokens are reshaped from their channel-major layout into windows:

$$\underbrace{(B,\ D,\ C\cdot T,\ d)}_{\text{per-day tokens}} \longrightarrow \underbrace{(B\cdot W,\ D\cdot C\cdot P_w,\ d)}_{\text{per-window tokens}}, \tag{24}$$

where $W = T/P_w$ is the number of windows per day and $P_w = w/p$ is the number of patches per window (w is the window width in minutes). Each window thus contains tokens from all D days, all C channels, within one contiguous time-of-day slot of w minutes. Self-attention is applied independently within each window, with per-layer complexity $O\big((D \cdot C \cdot P_w)^2\big)$.
In our configuration, w=120 minutes, $P_w = 120/10 = 12$, and $W = 144/12 = 12$ windows per day. Each window contains $D \cdot C \cdot P_w = 7 \times 19 \times 12 = 1{,}596$ tokens.

- **Rotary position embeddings (RoPE) for day offsets.** In cross-day layers, all tokens from the same day share a calendar-day offset $\delta_d \in \{0, \ldots, D-1\}$. RoPE encodes this offset into queries $\mathbf{q}$ and keys $\mathbf{k}$ as:

$$\mathbf{q}'_i = \mathbf{q}_i \odot \cos(\delta_d \cdot \boldsymbol{\theta}) \; - \; \bar{\mathbf{q}}_i \odot \sin(\delta_d \cdot \boldsymbol{\theta}), \tag{25}$$

  where $\boldsymbol{\theta}_j = 10000^{-2j/d_h}$ for head dimension d_h, and $\bar{\mathbf{q}}_i$ denotes the rotation partner (first and second halves of the head dimension swapped). Keys are rotated analogously. This enables the model to handle non-contiguous calendar days when some days are absent from a user's history, as the relative offset between any two days is encoded implicitly through the rotation angles. While we limit our experiments to 7-days, it is straightforward to extend to more days due to how RoPE encodes relative day offset differences.
- **Attention Complexity Comparison.** Table 22 summarizes the per-layer attention complexity of each variant. We define $T = L/p$ (patches per channel per day for the given patch size), P_w (patches per window), and $T' = L/p'$ (patches per channel per day in the weekly-dense model).

Table 22 | **Per-layer self-attention complexity for LSM-2 variants.** The "Tokens / layer" column shows the concrete sequence length for our configuration (C=19, D=7, T=144, T'=24, P_w=12).

Variant	Layer type	Complexity	Tokens / layer	Multi-day
LSM-2 (Daily)	Full	$O\big((CT)^2\big)$	2,736	No
LSM-2-Weekly	Full	$O\big((CDT')^2\big)$	3,192	Yes
LSM-2-Sparse	Day-local	$O\big((CT)^2\big)$	2,736	—
LSM-2-Sparse	Cross-day	$O\big((DCP_w)^2\big)$	1,596	Yes

The sparse decoder decouples the number of days D from the temporal resolution T. Day-local layers have complexity $O\big((CT)^2\big)$, identical to the daily model—multi-day context adds zero overhead. Cross-day layers scale as $O\big((DCP_w)^2\big)$ with $P_w \ll T$ ($P_w/T = 12/144 \approx 0.08$ in our setting), making cross-day attention strictly cheaper than day-local attention.

In contrast, the dense weekly model couples all dimensions into a single sequence with cost $O\big((CDT')^2\big)$, even with 6× coarser patches, its per-layer cost exceeds the sparse model's. In summary, LSM-2-Sparse achieves 7-day context at the same maximum per-layer cost as the single-day model while preserving 10-minute temporal resolution and thus allows us to handle longer context for imputation and forecasting tasks.

Hyperparameter Selection. Both stages of the LSM-2 pretraining pipeline, the daily MAE encoder and the sparse cross-day decoder, were tuned via Bayesian optimization sweeps conducted with Weights & Biases [Snoek et al., 2012]. Each sweep uses Hyperband early termination [Li et al., 2018] with minimum iterations = 5 and reduction factor η=3, optimizing the validation reconstruction loss on the validation split. Hyperparameters not listed in the tables below are held fixed at their default values.

Table 23 reports the search space and selected configuration for the daily LSM-2 encoder (15 Bayesian trials, 25 training epochs, cosine LR schedule).

Table 23 | Daily LSM-2 encoder HPO search space and selected configuration (Bayesian optimization, 15 trials, 25 epochs, cosine schedule).

Hyperparameter	**Search space**	**Selected value**
Learning rate (`lr`)	log-U$[10^{-6}, 10^{-3}]$	2.447×10^{-4}
Weight decay (`wd`)	log-U$[10^{-6}, 10^{-2}]$	1.500×10^{-3}
Batch size	$\{16, 32, 64, 80\}$	16

Table 24 reports the search space and selected configuration for the LSM-2-Sparse weekly decoder, which is trained with the daily encoder frozen. The sweep runs 30 Bayesian trials over optimizer, masking, and architecture hyperparameters (50 training epochs, cosine LR schedule).

Table 24 | LSM-2-Sparse weekly decoder HPO search space and selected configuration (Bayesian optimization, 30 trials, 50 epochs, cosine schedule).

Hyperparameter	**Search space**	**Selected value**
Learning rate (`lr`)	log-U$[10^{-5}, 5 \times 10^{-3}]$	2.12×10^{-4}
Weight decay (`wd`)	log-U$[10^{-6}, 10^{-2}]$	3.19×10^{-5}
Decoder depth (`decoder_depth`)	$\{2, 4\}$	4
Mask ratio (`mask_ratio`)	$\{0.3, 0.5, 0.75\}$	0.5

F.6.2. Imputation Deep Learning Baselines

Besides LSM2, all deep learning imputation baselines are trained via the PyPOTS library [Du, 2023] on minute-resolution daily tensors (C=19 channels, T=1,440 time steps). For BRITS, gradient clipping (max norm 1.0) is applied to prevent loss spikes from sporadic large gradients; the other PyPOTS baselines (DLinear, FEDformer, TimesNet) are trained without gradient clipping. During training, each channel is standardized by subtracting the training-set mean and dividing by the training-set standard deviation; the same scaler is applied to validation and test splits.

For each learned imputation model (BRITS, DLinear, FEDformer, TimesNet), we perform hyperparameter optimization on MHC-XS, using Bayesian optimization with Hyperband early stopping (η=3), running 15 trials per model and selecting the configuration that minimizes validation MAE. Per-model search spaces and selected configurations are reported in each model's section below.

DLinear. DLinear [Zeng et al., 2023] is a time series baseline that decomposes the input sequence into trend and seasonal components and applies separate linear projections to each part. DLinear is used in two roles in this benchmark: as a forecasting baseline (see Appendix G.2.2) and as an imputation baseline. The architecture is identical across both roles; the two roles differ only in training objective and hyperparameter optimization, detailed below. The selected hyperparameters reflect this: the forecasting variant favors a coarser moving-average window (301 minutes) and a larger batch size, while the imputation variant prefers a much finer window (51 minutes), a smaller batch size, and an order-of-magnitude larger learning rate, with additional ORT/MIT loss weights tuned in a separate sweep. We use the PyPOTS implementation for both [Du, 2023].

For imputation, the Observed Reconstruction Term (ORT) penalizes reconstruction error on observed positions, while the Masked Imputation Term (MIT) penalizes error on randomly masked positions. For DLinear, a two-stage search is used as it would not converge under a single-stage joint sweep (although this only marginally helped after all): the first stage selects architecture and optimizer

hyperparameters, after which the loss weights (ORT and MIT) are tuned in a second stage with the remaining parameters fixed. Table 25 reports the search space and selected configuration.

Table 25 | DLinear selected configuration for imputation (Bayesian optimization, 15 trials, 25 epochs, patience 10). Loss weights tuned in a separate second-stage sweep.

Hyperparameter	**Search space**	**Selected value**
Learning rate (`lr`)	log-U$[10^{-5}, 10^{-1}]$	6.469×10^{-3}
Batch size	$\{128, 256, 512, 1024\}$	128
Moving avg. window size	$\{25, 51, 101, 201, 301\}$	51
`d_model`	fixed	256
ORT weight	log-U$[0.01, 10]$	0.010
MIT weight	log-U$[0.01, 10]$	9.546

Long-context (7-day) imputation. DLinear is also extended to the long-context imputation track. We train the same DLinear architecture on concatenated 7-day inputs ($C \times 10{,}080$ time steps), allowing the model to exploit cross-day temporal patterns; the training objective (PyPOTS ORT/MIT) is identical to the single-day imputation variant. All architecture hyperparameters are shared with the single-day model; only the sequence length and batch size differ.

BRITS. BRITS (Bidirectional Recurrent Imputation for Time Series) [Cao et al., 2018] is an RNN-based model that learns temporal dynamics in both forward and backward directions, jointly estimating missing values and the underlying data-generating process. At each time step, the model combines its recurrent hidden state with feature-level regression to produce imputation estimates, and a consistency loss encourages agreement between the forward and backward passes. Table 26 reports the search space and selected configuration.

Table 26 | BRITS selected configuration for imputation (Bayesian optimization, 15 trials, 25 epochs, patience 24).

Hyperparameter	**Search space**	**Selected value**
Learning rate (`lr`)	log-U$[10^{-5}, 10^{-2}]$	2.780×10^{-3}
Batch size	$\{128, 256, 512\}$	256
RNN hidden size	$\{64, 128\}$	128

FEDformer. FEDformer (Frequency Enhanced Decomposed Transformer) [Zhou et al., 2022] replaces standard self-attention with Fourier-enhanced blocks that perform attention in the frequency domain, capturing global temporal dependencies with linear complexity. Like DLinear, it uses a seasonal-trend decomposition, but applies Transformer layers to each component. We use the Fourier version with random mode selection.

For FEDformer, we tune the learning rate, batch size, and moving-average window size for the imputation task, while keeping the remaining architecture parameters fixed. Table 27 reports the search space and selected configuration.

TimesNet. TimesNet [Wu et al., 2023] reshapes 1D time series into 2D tensors based on learned period lengths, then applies Inception-style 2D convolutions ("TimesBlocks") to jointly capture intra-period and inter-period variations. The top-k most salient periods are identified via FFT and processed in parallel. For TimesNet, we also tune the learning rate, batch size, and dropout for the imputation task, while keeping the remaining architecture parameters fixed. Table 28 reports the search space

Table 27 | FEDformer selected configuration for imputation (Bayesian optimization, 15 trials, 50 epochs, patience 10).

Hyperparameter	**Search space**	**Selected value**
Learning rate (`lr`)	log-U$[10^{-5}, 10^{-1}]$	1.000×10^{-3}
Batch size	$\{128, 256, 512\}$	512
Moving avg. window size	$\{25, 51, 101, 201, 301\}$	25
`n_layers`	fixed	2
`d_model`	fixed	512
`d_ffn`	fixed	512
`n_heads`	fixed	8
Fourier modes	fixed	64
Dropout	fixed	0.1

and selected configuration.

Table 28 | TimesNet selected configuration for imputation (Bayesian optimization, 15 trials, 50 epochs, patience 5).

Hyperparameter	**Search space**	**Selected value**
Learning rate (`lr`)	log-U$[10^{-5}, 10^{-1}]$	5.000×10^{-3}
Batch size	$\{64, 128, 256\}$	256
Dropout	U$[0, 0.5]$	0.4
`n_layers`	fixed	2
`d_model`	fixed	128
`d_ffn`	fixed	512
`top_k`	fixed	5
`n_kernels`	fixed	6
Weight decay	fixed	10^{-4}

G. Forecasting

In Appendix G.1, we formalize the forecasting task, specifically the data preprocessing and evaluation setup. In Appendix G.2, we discuss the forecasting models we train and evaluate. Finally, in Appendix G.3, we report additional forecasting model results.

G.1. Forecasting Task

G.1.1. Data Preprocessing

We formulate a wearable data forecasting task by constructing hourly user-level trajectories. We first aggregate the raw minute-level data to hourly resolution; see Section D.3 for details. For each individual $i \in [1:n]$, the hourly observations are ordered chronologically and concatenated across days to form a multivariate trajectory

$$Y_{i,1:T_i} = [Y_{i,1}, Y_{i,2}, \ldots, Y_{i,T_i}], \qquad Y_{i,t} = \left(Y_{i,t}^{(c)}\right)_{c=1}^{C},$$

where T_i is the trajectory length for individual i and $C = 19$ is the number of channels. These 19 channels consist of activity-related measurements from iPhone and Apple Watch, sleep indicators, and workout indicators; see Table 6. Each vector $Y_{i,t}$ contains the hourly values of all wearable channels at time step t. The goal of the forecasting model is to predict future vectors $Y_{i,t}$ given past observations.

Missingness Mask. Apple HealthKit does not provide information to differentiate whether a particular measurement is zero or missing. In order to prevent the model from predicting all zeros (e.g., due to days in which the individual did not wear their watch), we use the following crude procedure to define what we consider as "missing" versus a "true zero".

Specifically, we construct the following binary mask:

$$M_{i,1:T_i} = [M_{i,1}, M_{i,2}, \ldots, M_{i,T_i}], \qquad M_{i,t} = \left(M_{i,t}^{(c)}\right)_{c=1}^{C} \in \{0,1\}^C$$

where $M_{i,t}^{(c)} = 1$ indicates that channel c at time step t for individual i is treated as "observed", while $M_{i,t}^{(c)} = 0$ indicates that it is treated as "missing". We apply the zero-to-NaN transform defined in Section D.3 to the raw time-series data $Y_{i,t}^{(c)}$. We then set the corresponding entries of $M_{i,t}^{(c)}$ to 0 wherever the transformed values are NaN, and to 1 otherwise.

G.1.2. Evaluation Procedure

We consider forecast horizons $H = 24$, corresponding to predicting the next 24 hours. Using the individual-level trajectory data $(Y_{i,1:T_i}, M_{i,1:T_i})$, we form a sequence of forecasting examples for each $t \in \mathcal{T}_i$ (we discuss how we choose $\mathcal{T}_i$ below):

$$(Y_{i,1:t}, M_{i,1:t}) \longrightarrow Y_{i,t+1:t+H}. \tag{26}$$

That is, the model takes as input the historical target sequence available up to the forecasting horizon, and the historical validity mask, and predicts the future target sequence over the next H steps.

To form our training, validation, and test sets, we follow the same official splits used in the health outcome prediction and imputation tasks.

Specifying Evaluation Timepoints $\mathcal{T}_i$. We only evaluate on forecasting samples, akin to (26), starting at values of $t \in \mathcal{T}_i$. Specifically,

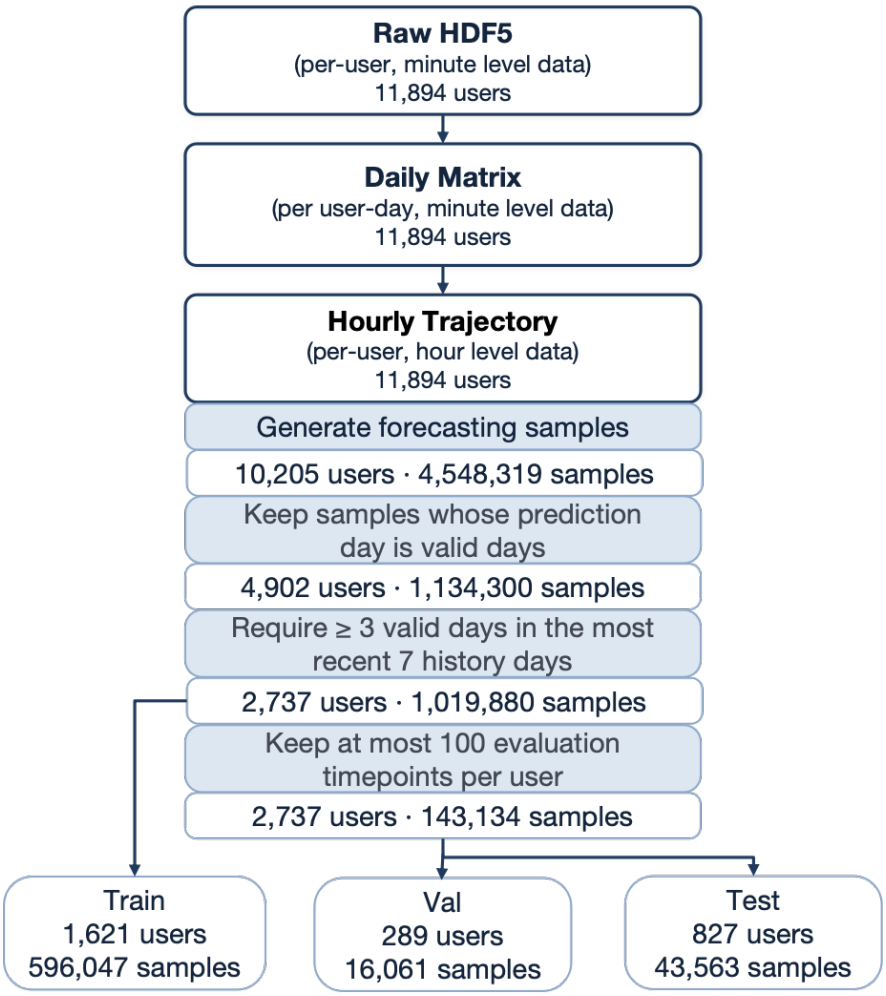

Figure 13 | Sample generation pipeline for the forecasting task. The initial participant filtering process is shown in Figure 4, and the construction details of the Daily Matrix are described in Section D.1.

- $\mathcal{T}_i$ only includes timepoints t that correspond to midnight according to the individual's local time; this allows evaluation to focus on daytime periods with more informative activity patterns.
- We require that all days used to form $Y_{i,t+1:t+H}$ are valid days. In addition, we require at least three of the most recent seven days in the historical sequence $Y_{i,1:t}$ to be valid. A day is considered valid if its minute-level data pass wear-time filter and low-variance filter described in Sections D.3.
- For evaluation, we restrict $|\mathcal{T}_i| \leq 100$ by subsampling at most 100 timesteps t that satisfy all the above criteria.

The complete sample generation workflow, as well as the number of users and samples retained at each stage, is shown in Figure 13. After applying these filters, the test cohort comprises 827 users contributing 43,563 forecasting sub-trajectories, with a mean of 52.7 (median 48) evaluation days per user; 352 of the 827 users hit the per-user cap of 100 days.

Forecasting Model Predictions. The model outputs

$$\hat{Y}_{i,t+1:t+H} = [\hat{Y}_{i,t+1}, \ldots, \hat{Y}_{i,t+H}], \qquad \hat{Y}_{i,t+h} = (\hat{Y}^{(c)}_{i,t+h})_{c=1}^{C},$$

where $\hat{Y}^{(c)}_{i,t+h}$ denotes the predicted value of channel c at forecast step h.

Handling NaN Point Forecasts. If a model returns `NaN` for any point forecast, we substitute the corresponding Seasonal Naive prediction at that channel and forecast step before scoring.

G.1.3. Evaluation Metrics

Among the 19 channels shown in Table 6, there are two types of data: continuous and binary.

- For continuous channels, we use MAE and QL to evaluate the quality of point forecasts and probabilistic forecasts, respectively. For models that do not natively support probabilistic forecasting, we report MAE only.
- For binary channels, we use AUROC to evaluate the prediction results.

Mean Absolute Error (MAE). For continuous channels, the primitive evaluation quantity is the masked absolute error

$$AE_{i,t,h}^{(c)} = M_{i,t+h}^{(c)} \left| Y_{i,t+h}^{(c)} - \hat{Y}_{i,t+h}^{(c)} \right|,$$

where i indexes participants, $t \in \mathcal{T}_i$ indexes valid forecasting windows, $h \in \{0, \ldots, H-1\}$ indexes the forecast horizon, and $M_{i,t+h}^{(c)}$ indicates whether channel c is observed at that target time.

For participant i and continuous channel c, we pool the absolute error over all observed forecast hours across that participant's windows—equivalently, weighting each window by its number of observed forecast hours:

$$MAE_i^{(c)} = \frac{\sum_{t \in \mathcal{T}_i} \sum_{h=0}^{H-1} AE_{i,t,h}^{(c)}}{\sum_{t \in \mathcal{T}_i} \sum_{h=0}^{H-1} M_{i,t+h}^{(c)}}.$$

Area Under the Receiver Operating Characteristic Curve (AUROC).

For binary channels, we evaluate predicted scores using AUROC. For each participant and binary channel, we pool the valid binary target hours across all of that participant's forecast windows and compute a single AUROC over the pooled (target, score) pairs. A participant–channel whose pooled targets contain only a single class does not define an AUROC value and is therefore excluded.

For model-level comparison, each participant thus contributes one AUROC value per binary channel. The main results group binary channels into sleep and workout categories by averaging the available participant-level channel AUROC values within each group (larger AUROC is better). For skill scores, AUROC is converted to an error scale as $e = 1 - \text{AUROC}$, floored at $\varepsilon = 0.005$, and then compared with the Seasonal Naive baseline using the same paired participant-level procedure as for the continuous metrics. The category-balanced aggregation of these per-channel skills and ranks into the headline S and R is detailed in Section G.1.4.

G.1.4. Aggregation and Scoring

The headline forecasting columns, the aggregate skill score S, the average rank R, and the per-category skill scores, are produced from the per-participant errors above through the unified skill-score machinery of Appendix B (clip bounds $\ell = 0.01$, $u = 100$; baseline $b = \textsc{Seasonal Naive}$), specialized to the forecasting track. Because the 19 channels split very unevenly across sensor types, the aggregation is *category-balanced*: the channels partition into four reporting categories that act as equal-weight buckets—*Activity* (channels 0–4: phone and watch step counts and distances, and flights climbed; continuous, scored by MAE), *Physiology* (channels 5–6: heart rate and active energy; continuous, MAE), *Sleep* (channels 7–8: the asleep and in-bed indicators; binary, AUROC), and *Workout* (channels 9–18: the ten workout-type indicators; binary, AUROC).

Per-channel paired ratio. For each channel c we form the ratio against the baseline *within* each participant and geometrically average it over participants $\mathcal{P}^{(c)}$:

$$R_m^{(c)} = \exp\left(\frac{1}{|\mathcal{P}^{(c)}|}\sum_{p\in\mathcal{P}^{(c)}} \log \mathrm{clip}\left(E_{m,p}^{(c)}/E_{b,p}^{(c)},\ \ell, u\right)\right),$$

where $E_{m,p}^{(c)}$ is participant p's error for model m on channel c—the pooled MAE $MAE_i^{(c)}$ for a continuous channel and $\max(1 - \mathrm{AUROC}_{m,p}^{(c)}, \varepsilon)$ with $\varepsilon = 0.005$ for a binary channel—and $\mathcal{P}^{(c)}$ is the set of participants for whom both errors are defined and finite with $E_{b,p}^{(c)} > 0$.

Category-balanced skill score. Let a scope be a set of categories $\mathcal{K}$ and let C_k denote the channels of category k. The skill score averages the clipped log-ratios over the channels within each category, then over the categories with equal weight:

$$S_{m,\mathcal{K}} = 1 - \exp\left(\frac{1}{|\mathcal{K}|}\sum_{k\in\mathcal{K}}\frac{1}{|C_k|}\sum_{c\in C_k}\log \mathrm{clip}(R_m^{(c)},\ \ell, u)\right). \tag{27}$$

The reported per-category columns take $\mathcal{K} = \{k\}$ (a single inner mean over that category's channels), while the aggregate S takes $\mathcal{K}$ to be all four categories. Each category therefore has equal voice, so the ten Workout channels cannot dominate the two Sleep or the seven continuous channels.

Category-balanced average rank. For each channel c, models are ranked *within* each participant by their per-participant error—$E_i^{(c)}$ for a continuous channel and the *unfloored* 1 – AUROC for a binary channel (ascending, ties averaged)—and these per-participant ranks are averaged into a task rank $\bar{\rho}_{m,c}$. The scope rank then follows the same equal-weight category hierarchy, with arithmetic means at every level (rank is already scale-free, so there is no log/exp):

$$\rho_{m,\mathcal{K}} = \frac{1}{|\mathcal{K}|}\sum_{k\in\mathcal{K}}\frac{1}{|C_k|}\sum_{c\in C_k}\bar{\rho}_m^{(c)}. \tag{28}$$

The reported average rank R uses the overall (four-category) scope, as S does.

Unlike the imputation track (Appendix F.4), the forecasting aggregation has no scenario level and keeps per-channel binary tasks—a binary category is balanced through the equal-weight category means of (27) and (28) rather than by collapsing its channels into a single per-participant task.

G.2. Forecasting Models

Since our data are multivariate time series, we aimed to include models that natively support multivariate forecasting, in order to better exploit cross-channel information. This primarily drove the decision around which time series foundation models to evaluate. For models that do not support this setting (e.g., statistical models), we perform univariate forecasting for each channel separately and then aggregate the results into multivariate forecasts.

G.2.1. Statistical models

Statistical models' inference is performed on a CPU server equipped with 8 cores. Because historical data in some forecasting examples in our benchmark contain particularly long time series, we impose a maximum input length on each forecasting examples due to time constraints, truncating each historical data to 336 time steps (14 days).

We consider three statistical baselines:

- **Seasonal Naive** [Hyndman and Athanasopoulos, 2018]: a simple forecasting baseline that repeats the value observed at the corresponding seasonal lag in the past. In our hourly setting, it predicts each future time step using the value from the same hour of the previous day.
- **Auto_ARIMA** (implemented with SkTime library [Löning et al., 2019]): an automatically configured AutoRegressive Integrated Moving Average model that selects the ARIMA orders from data. It captures linear temporal dependencies, trends, and autocorrelation in the historical series.
- **Auto_ETS** (implemented with SkTime library [Löning et al., 2019]): an automatically configured exponential smoothing model with error, trend, and seasonality components. It is designed to model level, trend, and seasonal patterns in time series.

G.2.2. Deep Learning Models (Trained from Scratch): DLinear, MixLinear, SegRNN

We evaluate three deep learning forecasting models trained from scratch on the benchmark's training split: DLinear [Zeng et al., 2023], MixLinear [Ma et al., 2024], and SegRNN [Lin et al., 2025]. To train these models, we run targeted Bayesian optimization using Weights & Biases sweeps [Snoek et al., 2012]. To limit compute cost, the sweep is capped at 15 trials and uses Hyperband early termination [Li et al., 2018] with minimum iterations = 5 and reduction factor $\eta = 3$. Table 29 lists the range of parameters we search over for each model.

During training, the input trajectories are standardized using a StandardScaler fitted only on the training split, where each channel is transformed by subtracting the training-set mean and dividing by the training-set standard deviation. The same scaler is then applied to the validation and test splits to avoid data leakage. For all deep learning models, we use implementations readily available in the PyPOTS library [Du, 2023].

Table 29 | **Hyperparameter optimization search space for model training.**

Model	**Hyperparameter**		
DLinear	Batch size	Learning rate	Window size of moving average
Search Range	[128, 256, 512, 1024]	$[10^{-5} : 10^{-1}]$	[25, 51, 101, 201, 301]
MixLinear	Batch size	Learning rate	Segment length
Search Range	[256, 512, 1024]	$[10^{-5} : 10^{-1}]$	[2, 4, 6, 8]
SegRNN	Batch size	Learning rate	Segment length
Search Range	[64, 128, 256, 512]	$[10^{-5} : 10^{-1}]$	[6, 12, 24]

DLinear. DLinear [Zeng et al., 2023] is a time series baseline that decomposes the input sequence into trend and seasonal components and applies separate linear projections to each part. DLinear was also used for imputation (identical architecture, different loss function/hyperparameter optimization, and time-scale). The forecasting variant favors a coarser moving-average window (301 minutes) and a larger batch size. As with the imputation variant, we use the PyPOTS implementation [Du, 2023].

Trained with standard next-horizon MSE on the benchmark training split (StandardScaler fit on train, applied to validation and test); 15 Bayesian trials with Hyperband early termination (η=3), 30 epochs, patience 10. The full forecasting HPO search space across deep learning baselines is summarized in Table 29; the selected configuration is in Table 30.

MixLinear. MixLinear [Ma et al., 2024] is an extremely lightweight forecasting model that combines segment-based linear modeling in the time domain with adaptive low-rank filtering in the frequency domain.

Table 30 | DLinear selected configuration (Bayesian optimization, 15 trials, 30 epochs, patience 10).

Hyperparameter	Search space	Selected value
Learning rate (`lr`)	log-U$[10^{-5}, 10^{-1}]$	3.686×10^{-4}
Batch size	{128, 256, 512, 1024}	512
Moving avg. window size	{25, 51, 101, 201, 301}	301

Table 31 | **MixLinear selected configuration (Bayesian optimization, 15 trials, 30 epochs, patience 10).**

Hyperparameter	Search space	Selected value
Learning rate (`lr`)	log-U$[10^{-5}, 10^{-1}]$	2.214×10^{-4}
Batch size	{256, 512, 1024}	1024
Segment length (`period_len`)	{2, 4, 6, 8}	6

SegRNN. SegRNN [Lin et al., 2025] is an RNN-based model designed for long-horizon forecasting through segment-wise iterations and parallel multi-step forecasting.

Table 32 | SegRNN selected configuration (Bayesian optimization, 15 trials, 30 epochs, patience 10).

Hyperparameter	Search space	Selected value
Learning rate (`lr`)	log-U$[10^{-5}, 10^{-1}]$	2.025×10^{-3}
Batch size	{64, 128, 256, 512}	256
Segment length (`seg_len`)	{6, 12, 24}	12

G.2.3. ToTo

We additionally evaluate Toto [Cohen et al., 2025] both in zero-shot and fine-tuned settings. The fine-tuned model is also used a extractor for health outcome prediction.

Architecture. Toto [Cohen et al., 2025] is a Transformer decoder-only foundation model for multivariate time series forecasting. It models temporal and cross-variate dependencies through proportional factorized attention and uses causal patch-wise normalization with a Student-t mixture output head. We use the publicly available `Datadog/Toto-Open-Base-1.0` checkpoint from Hugging Face (PyPI version 0.1.4), which was trained for a maximum context length of 4,096 tokens.

Fine-tuning on MHC data. Toto is fine-tuned on the MHC forecasting training split following the official implementation.[1] The fine-tuned checkpoint corresponds to epoch 24 (step 116,225, validation loss −1.3597). The forecasting task uses a 336-step (14-day) historical context to predict the next 24 hours across all 19 wearable channels.

G.2.4. Chronos-2

We additionally evaluate Chronos-2 [Ansari et al., 2025] both in zero-shot and fine-tuned settings. The fine-tuned model is also used a extractor for health outcome prediction.

Architecture. Chronos-2 [Ansari et al., 2025] is a Transformer encoder-only time-series foundation

[1] https://github.com/datadog/toto

Table 33 | **Toto fine-tuning configuration.**

Parameter	Value
Base model	Datadog/Toto-Open-Base-1.0
Max context length	2,048
Number of channels (C)	19
Forecast horizon	24 hours
Best checkpoint	epoch 24, val_loss = −1.3597
Fine-tuning method	LoRA (r=32, α=32, dropout = 0.097)
Optimizer	AdamW (β_1=0.9, β_2=0.999, weight decay = 0.01)
Learning rate	8.56×10^{-4} peak; warmup/stable/decay = 200/1000/1000 steps; min = 10^{-5}
Batch size	128
Epochs	25

model that introduces a group attention mechanism to enable information sharing across related series. It natively handles univariate, multivariate, and covariate-informed forecasting in a unified framework. We use the publicly available `amazon/chronos-2` checkpoint from Hugging Face (PyPI version 2.2.2), which supports a maximum context length of 8,192 tokens.

Fine-tuning on MHC data. Chronos-2 is fine-tuned on the MHC forecasting training split using LoRA (Low-Rank Adaptation Hu et al. [2022]), following the official implementation.[2] The fine-tuning uses a 168-step (7-day) minimum past context with 336-step (14-day) input sequences to predict the next 24 hours across 19 channels.

Table 34 | **Chronos-2 LoRA fine-tuning configuration.**

Parameter	Value
Base model	amazon/chronos-2
Max context length	8,192
Number of channels (C)	19
Input steps (`n_steps`)	336
Prediction steps (`n_pred_steps`)	24
Context length (minimum past)	168
Fine-tuning method	LoRA
LoRA rank (r)	16
LoRA alpha (α)	16
LoRA dropout	0.0
Optimizer learning rate	4.65×10^{-5}
Batch size	760
Epochs	25

Hyperparameter selection. The LoRA fine-tuning hyperparameters are selected via Bayesian optimization using Weights & Biases sweeps [Snoek et al., 2012], capped at 15 trials with Hyperband early termination (minimum iterations = 3, reduction factor η=3). The optimization metric is `eval/loss` (minimize). Table 35 reports the search space and selected values.

[2] https://github.com/amazon-science/chronos-forecasting

Table 35 | **Chronos-2 LoRA HPO search space and selected values.**

Hyperparameter	Search space	Selected
Learning rate	Log-uniform $[10^{-7}, 10^{-4}]$	4.65×10^{-5}
LoRA rank (r)	$\{4, 8, 16\}$	16
LoRA alpha (α)	$\{8, 16, 32\}$	16
LoRA dropout	Uniform $[0.0, 0.2]$	0.0

G.3. Additional Results

This section presents detailed channel-level skill scores for each model, reported separately for continuous in Table 36 and binary channels in Table 37. The detailed channel information can find in Table 6.

Overall, these findings are consistent with the results reported in Table 4. Notably, Toto performs very poorly on the HeartRate channel, with a skill score of −8.7, but its performance improves substantially after fine-tuning. In addition, although DLinear achieves an overall skill score of 5.1 for the Workout group, this improvement appears to be driven primarily by strong performance on only a few specific workout channels.

In addition, Figure 14 provides an example of model forecasts on our dataset.

Figure 14 | **Example forecasting results across three channels: StepCount(iPhone), HeartRate, and ActiveEnergyBurned.** Each panel shows the 48-hour historical context, followed by the 24-hour forecasting horizon with both ground-truth observations and model predictions. FT denotes fine-tuned.

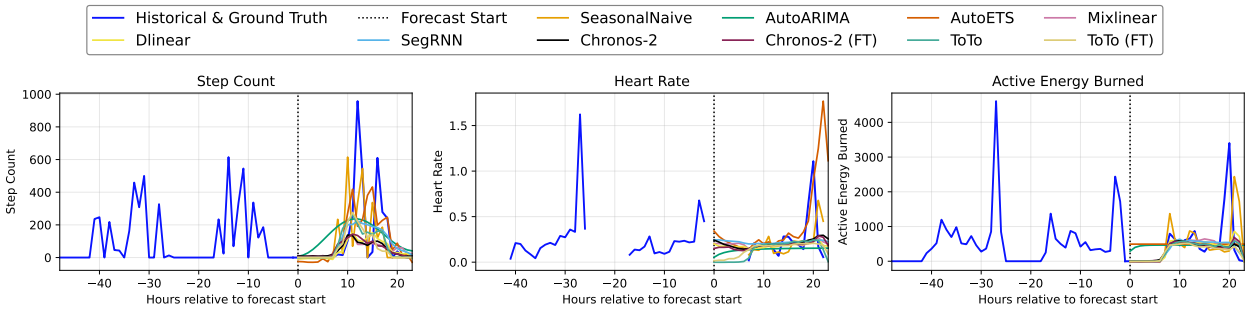

Table 36 | **Channel-level forecasting performance on continuous channels.** The table reports the skill score (in %) of each model relative to the Seasonal Naive baseline for seven continuous channels. Subscripts and superscripts indicate the 95% bootstrap confidence interval based on 1000 resamples. FT denotes fine-tuned.

Method	StepCount (iPhone)	Distance (iPhone)	Flights Climbed	StepCount (Watch)	Distance (Watch)	HeartRate	Energy Burned
Statistical Models							
Seasonal Naive	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$
AutoARIMA	$-1.5^{+1.6}_{-2.0}$	$-1.5^{+1.5}_{-1.7}$	$-5.7^{+1.7}_{-1.7}$	$-0.2^{+1.2}_{-1.4}$	$-0.3^{+1.3}_{-1.4}$	$-11.2^{+2.9}_{-2.7}$	$-6.9^{+1.4}_{-1.5}$
AutoETS	$+1.1^{+1.4}_{-1.8}$	$+1.8^{+1.5}_{-1.7}$	$-4.0^{+1.5}_{-1.6}$	$+1.8^{+1.8}_{-1.5}$	$+2.3^{+1.1}_{-1.2}$	$-49.0^{+3.9}_{-4.2}$	$-8.0^{+2.7}_{-3.5}$
Neural Models							
MixLinear	$+19.3^{+1.4}_{-1.5}$	$+20.9^{+1.3}_{-1.4}$	$+36.0^{+2.2}_{-2.1}$	$+19.6^{+1.3}_{-1.5}$	$+19.8^{+1.3}_{-1.6}$	$+10.9^{+2.8}_{-3.3}$	$+15.7^{+1.7}_{-1.9}$
DLinear	$+22.4^{+1.2}_{-1.4}$	$+22.9^{+1.2}_{-1.4}$	$+33.9^{+1.9}_{-1.9}$	$+22.7^{+1.2}_{-1.4}$	$+22.7^{+1.2}_{-1.4}$	$+12.3^{+2.9}_{-3.2}$	$+20.5^{+1.5}_{-1.8}$
SegRNN	$+25.4^{+1.2}_{-1.2}$	$+24.8^{+1.4}_{-1.5}$	$+27.6^{+2.4}_{-2.5}$	$+24.6^{+1.2}_{-1.4}$	$+24.7^{+1.2}_{-1.4}$	$+17.4^{+2.4}_{-2.6}$	$+24.1^{+1.2}_{-1.5}$
Foundation Models							
Chronos-2	$+29.3^{+1.2}_{-1.3}$	$+29.6^{+1.3}_{-1.4}$	$+41.1^{+2.8}_{-2.4}$	$+25.4^{+0.9}_{-1.0}$	$+25.7^{+0.9}_{-0.9}$	$+28.9^{+1.0}_{-1.0}$	$+24.0^{+0.9}_{-0.9}$
Chronos-2 (FT)	$\mathbf{+29.4}^{+1.2}_{-1.3}$	$\mathbf{+29.8}^{+1.3}_{-1.4}$	$\mathbf{+41.3}^{+2.8}_{-2.5}$	$\mathbf{+25.7}^{+0.9}_{-1.0}$	$\mathbf{+26.0}^{+0.9}_{-0.9}$	$\mathbf{+29.2}^{+1.0}_{-1.0}$	$+24.5^{+0.9}_{-0.9}$
Toto	$+27.9^{+1.3}_{-1.3}$	$+28.3^{+1.4}_{-1.4}$	$\mathbf{+41.3}^{+2.6}_{-2.4}$	$+23.1^{+1.1}_{-1.1}$	$+23.6^{+1.2}_{-1.1}$	$-8.7^{+3.2}_{-3.1}$	$+20.0^{+1.2}_{-1.2}$
Toto (FT)	$+27.2^{+1.1}_{-1.1}$	$+27.5^{+1.2}_{-1.3}$	$+41.2^{+2.7}_{-2.5}$	$+24.9^{+1.0}_{-1.0}$	$+25.1^{+0.9}_{-0.9}$	$+27.5^{+1.2}_{-1.2}$	$\mathbf{+24.6}^{+0.8}_{-0.8}$

Table 37 | **Channel-level forecasting performance on binary channels.** The table reports the skill score (in %) of each model relative to the Seasonal Naive baseline for twelve binary channels, displayed in two connected panels. Subscripts and superscripts indicate the 95% bootstrap confidence interval based on 1000 resamples. FT denotes fine-tuned.

Method	Asleep	In Bed	Walking	Cycling	Running	Other
Statistical Models						
Seasonal Naive	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$
AutoARIMA	$+13.7^{+6.2}_{-6.3}$	$-0.2^{+6.9}_{-6.8}$	$-9.5^{+8.5}_{-10.2}$	$+3.9^{+9.2}_{-11.2}$	$+14.0^{+9.7}_{-10.6}$	$+26.8^{+11.2}_{-10.8}$
AutoETS	$+38.1^{+4.0}_{-4.0}$	$+37.0^{+3.9}_{-3.8}$	$\mathbf{+8.5}^{+7.0}_{-7.9}$	$\mathbf{+15.0}^{+8.0}_{-8.6}$	$\mathbf{+31.5}^{+8.1}_{-8.9}$	$+31.2^{+11.4}_{-11.7}$
Neural Models						
MixLinear	$+64.4^{+2.5}_{-2.3}$	$+64.8^{+1.8}_{-1.8}$	$-58.5^{+13.2}_{-15.2}$	$-0.9^{+11.6}_{-14.0}$	$-1.3^{+11.6}_{-12.9}$	$\mathbf{+32.6}^{+9.1}_{-9.6}$
DLinear	$\mathbf{+69.5}^{+1.8}_{-1.9}$	$\mathbf{+73.4}^{+1.7}_{-1.6}$	$-2.8^{+6.9}_{-8.2}$	$-31.4^{+14.2}_{-18.1}$	$+0.3^{+13.7}_{-15.9}$	$+3.6^{+12.4}_{-13.5}$
SegRNN	$+66.1^{+2.5}_{-2.6}$	$+70.1^{+1.8}_{-2.1}$	$-0.5^{+7.2}_{-8.8}$	$-1.5^{+11.6}_{-14.8}$	$-3.1^{+10.4}_{-12.0}$	$+3.5^{+12.4}_{-13.5}$
Foundation Models						
Chronos-2	$+59.0^{+3.0}_{-2.9}$	$+65.4^{+2.3}_{-2.3}$	$+3.3^{+7.9}_{-9.8}$	$+9.3^{+10.1}_{-11.8}$	$+12.6^{+10.3}_{-12.2}$	$+19.1^{+9.8}_{-10.4}$
Chronos-2 (FT)	$+60.9^{+2.9}_{-2.8}$	$+66.6^{+2.1}_{-2.2}$	$+5.6^{+8.0}_{-9.2}$	$+9.1^{+10.1}_{-11.7}$	$+14.2^{+10.3}_{-11.5}$	$+16.4^{+9.6}_{-10.2}$
Toto	$+47.9^{+3.3}_{-3.2}$	$+53.1^{+2.7}_{-2.8}$	$-9.1^{+7.6}_{-9.0}$	$+0.5^{+8.0}_{-10.8}$	$+7.2^{+8.4}_{-9.5}$	$+11.6^{+9.8}_{-12.1}$
Toto (FT)	$+45.1^{+3.3}_{-3.2}$	$+47.1^{+2.6}_{-2.7}$	$-2.4^{+7.1}_{-8.2}$	$+6.5^{+8.5}_{-9.5}$	$+11.5^{+8.7}_{-9.5}$	$+12.9^{+9.6}_{-12.4}$

Table 37 continued: remaining binary channels

Method	Cardio	Strength	Elliptical	HIIT	Functional	Yoga
Statistical Models						
Seasonal Naive	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$	$+0.0^{+0.0}_{-0.0}$
AutoARIMA	$+28.0^{+34.1}_{-44.5}$	$+14.1^{+18.5}_{-24.3}$	$+25.9^{+11.0}_{-12.2}$	$+39.6^{+16.9}_{-21.3}$	$\mathbf{+41.8}^{+14.4}_{-16.4}$	$+39.5^{+11.2}_{-12.0}$
AutoETS	$+16.8^{+34.3}_{-42.8}$	$\mathbf{+27.8}^{+15.3}_{-18.6}$	$\mathbf{+38.8}^{+11.1}_{-12.9}$	$\mathbf{+48.8}^{+13.7}_{-14.6}$	$+40.3^{+13.4}_{-15.8}$	$\mathbf{+44.4}^{+10.3}_{-11.3}$
Neural Models						
MixLinear	$+36.7^{+28.9}_{-49.0}$	$-41.3^{+22.5}_{-30.2}$	$-7.7^{+16.5}_{-19.2}$	$-43.9^{+33.8}_{-52.0}$	$-5.8^{+21.7}_{-30.8}$	$-24.6^{+18.8}_{-24.6}$
DLinear	$+26.7^{+43.7}_{-94.7}$	$+27.4^{+14.9}_{-18.8}$	$+29.2^{+13.5}_{-16.8}$	$-38.8^{+34.4}_{-49.8}$	$+8.2^{+21.5}_{-26.9}$	$+8.4^{+16.9}_{-18.9}$
SegRNN	$-30.2^{+30.7}_{-50.1}$	$-2.6^{+23.8}_{-32.1}$	$+25.9^{+14.2}_{-14.9}$	$+12.1^{+20.8}_{-32.3}$	$+6.5^{+18.6}_{-25.9}$	$+6.4^{+20.5}_{-25.0}$
Foundation Models						
Chronos-2	$+39.5^{+30.1}_{-53.3}$	$+16.3^{+16.6}_{-22.3}$	$+19.4^{+17.0}_{-19.5}$	$+11.2^{+23.9}_{-31.3}$	$-7.0^{+26.8}_{-34.2}$	$+16.6^{+16.5}_{-17.2}$
Chronos-2 (FT)	$+32.6^{+35.5}_{-68.0}$	$+21.7^{+16.1}_{-20.1}$	$+21.5^{+17.1}_{-20.4}$	$+13.3^{+20.0}_{-23.6}$	$+6.9^{+24.5}_{-33.3}$	$+25.0^{+16.6}_{-17.8}$
Toto	$+9.6^{+34.3}_{-51.1}$	$-4.8^{+15.7}_{-22.9}$	$+22.9^{+13.8}_{-15.3}$	$+16.9^{+24.3}_{-34.5}$	$+26.3^{+13.3}_{-16.3}$	$+29.7^{+13.5}_{-14.5}$
Toto (FT)	$\mathbf{+61.0}^{+29.1}_{-71.6}$	$+1.5^{+16.9}_{-23.8}$	$+13.6^{+12.9}_{-12.9}$	$+11.8^{+18.8}_{-27.3}$	$+24.0^{+14.1}_{-17.3}$	$+24.3^{+14.7}_{-16.5}$

H. Full Registry of Linked Variables

This section enumerates every variable exposed through the labels API in our accompanying codebase. (`src/labels/`) for the OpenMHC release. The registry contains 169 labels in total: 7 longitudinal HealthKit-derived metrics extracted from raw Apple Watch, and 162 self-reported survey variables collected through the MHC app's questionnaires and ingested by, and the. Variables are grouped into 16 semantic categories. The **Source** column distinguishes HealthKit (`HK`) from survey-derived (`Survey`) labels. The **Role** column marks each label as a downstream prediction *target* (`T`, exposed through `TARGET_NAMES`) or an auxiliary *context* covariate (`C`, exposed through `CONTEXT_NAMES`). The **Type** column reports the value type enforced by `src/labels/api.py`.

Table 39 | Complete labels registry for the OpenMHC release (N = 169). HealthKit (HK) labels are derived from raw Apple Watch `HKQuantityTypeIdentifier` records; survey labels are self-reported through the MHC app questionnaires. Roles are T (predictive target) or C (auxiliary context covariate).

Label	Category	Source	Role	Type
BMI_categories	anthropometrics	Survey	T	ordinal
BMI_values	anthropometrics	Survey	T	continuous
WeightKilograms	anthropometrics	Survey	T	continuous
field_HeightCentimeters	anthropometrics	Survey	C	continuous
Diabetes	cardiometabolic_labs	Survey	T	binary
Hdl	cardiometabolic_labs	Survey	T	continuous
Hypertension	cardiometabolic_labs	Survey	T	binary
Ldl	cardiometabolic_labs	Survey	T	continuous
SystolicBloodPressure	cardiometabolic_labs	Survey	T	continuous
TotalCholesterol	cardiometabolic_labs	Survey	T	continuous
blood_pressure_categories	cardiometabolic_labs	Survey	T	ordinal
field_BloodGlucose	cardiometabolic_labs	Survey	C	continuous
framingham_risk	cardiometabolic_labs	Survey	T	continuous
Atrial fibrillation (Afib)	cardiovascular_disease_history	Survey	T	binary
CAD	cardiovascular_disease_history	Survey	T	binary
Cerebrovascular Disease	cardiovascular_disease_history	Survey	T	binary
Congenital Heart	cardiovascular_disease_history	Survey	T	binary
Heart Failure or CHF	cardiovascular_disease_history	Survey	T	binary
PH	cardiovascular_disease_history	Survey	T	binary
Peripheral/Systemic Vascular Disease	cardiovascular_disease_history	Survey	T	binary
cardiovascular_disease	cardiovascular_disease_history	Survey	T	binary
field_family_history	cardiovascular_disease_history	Survey	C	multi_categorical
field_medications_to_treat	cardiovascular_disease_history	Survey	C	multi_categorical
field_antibiotics	covid_19	Survey	C	multi_categorical
field_building	covid_19	Survey	C	ordinal
field_conditions	covid_19	Survey	C	multi_categorical
field_covid	covid_19	Survey	C	ordinal
field_covid_serologic	covid_19	Survey	C	ordinal
field_daily_activities	covid_19	Survey	C	ordinal
field_days_admitted	covid_19	Survey	C	continuous
field_exposure	covid_19	Survey	C	ordinal
field_face_covering	covid_19	Survey	C	ordinal
field_healthcare_worker	covid_19	Survey	C	categorical
field_most_intense_care	covid_19	Survey	C	ordinal
field_self_isolating	covid_19	Survey	C	ordinal
field_severity	covid_19	Survey	C	ordinal
field_severity_covid	covid_19	Survey	C	ordinal
field_symptoms_past_week	covid_19	Survey	C	multi_categorical
field_symptoms_week_preceding	covid_19	Survey	C	multi_categorical
BiologicalSex	demographics	Survey	T	binary
age	demographics	Survey	T	continuous
field_Ethnicity_heartage	demographics	Survey	C	categorical
field_FitzpatrickSkinType	demographics	Survey	C	ordinal
field_education	demographics	Survey	C	ordinal
field_ethnicity	demographics	Survey	C	ordinal
field_race	demographics	Survey	C	multi_categorical
field_alcohol	diet	Survey	C	ordinal

Continued on next page.

Table 39 continued from previous page.

Label	Category	Source	Role	Type
field_fish	diet	Survey	C	continuous
field_fruit	diet	Survey	C	continuous
field_grains	diet	Survey	C	continuous
field_sodium	diet	Survey	C	multi_categorical
field_sugar_drinks	diet	Survey	C	continuous
field_vegetable	diet	Survey	C	continuous
field_country	geography	Survey	C	categorical
field_zip	geography	Survey	C	categorical
Watch_BasalEnergyBurned	healthkit_watch_metrics	HK	T	continuous
Watch_HeartRateVariabilitySDNN	healthkit_watch_metrics	HK	T	continuous
Watch_RespiratoryRate	healthkit_watch_metrics	HK	T	continuous
Watch_RestingHeartRate	healthkit_watch_metrics	HK	T	continuous
Watch_StandTime	healthkit_watch_metrics	HK	T	continuous
Watch_VO2Max	healthkit_watch_metrics	HK	T	continuous
Watch_WalkingHeartRateAverage	healthkit_watch_metrics	HK	T	continuous
field_beneficial	mindset_measures	Survey	C	ordinal
field_body_remarkable_self_healing	mindset_measures	Survey	C	ordinal
field_body_self_healing_from_most_conditions_and_diseases	mindset_measures	Survey	C	ordinal
field_body_self_healing_in_many_different_circumstances	mindset_measures	Survey	C	ordinal
field_chronic_illness_body_betrayal	mindset_measures	Survey	C	ordinal
field_chronic_illness_body_blame	mindset_measures	Survey	C	ordinal
field_chronic_illness_body_coping	mindset_measures	Survey	C	ordinal
field_chronic_illness_body_failure	mindset_measures	Survey	C	ordinal
field_chronic_illness_body_handling	mindset_measures	Survey	C	ordinal
field_chronic_illness_body_management	mindset_measures	Survey	C	ordinal
field_chronic_illness_body_meaning	mindset_measures	Survey	C	ordinal
field_chronic_illness_challenge	mindset_measures	Survey	C	ordinal
field_chronic_illness_empowering	mindset_measures	Survey	C	ordinal
field_chronic_illness_handling	mindset_measures	Survey	C	ordinal
field_chronic_illness_impact	mindset_measures	Survey	C	ordinal
field_chronic_illness_management	mindset_measures	Survey	C	ordinal
field_chronic_illness_more_meaning_in_life	mindset_measures	Survey	C	ordinal
field_chronic_illness_positive_opportunity	mindset_measures	Survey	C	ordinal
field_chronic_illness_relatively_normal_life	mindset_measures	Survey	C	ordinal
field_chronic_illness_runing_life	mindset_measures	Survey	C	ordinal
field_chronic_illness_spoil	mindset_measures	Survey	C	ordinal
field_convenient	mindset_measures	Survey	C	ordinal
field_disease	mindset_measures	Survey	C	ordinal
field_easy	mindset_measures	Survey	C	ordinal
field_fun	mindset_measures	Survey	C	ordinal
field_indulgent	mindset_measures	Survey	C	ordinal
field_muscles	mindset_measures	Survey	C	ordinal
field_pleasurable	mindset_measures	Survey	C	ordinal
field_relaxing	mindset_measures	Survey	C	ordinal
field_social	mindset_measures	Survey	C	ordinal
field_unhealthy	mindset_measures	Survey	C	ordinal
field_weight	mindset_measures	Survey	C	ordinal

Continued on next page.

Table 39 continued from previous page.

Label	Category	Source	Role	Type
field_chestPain	parq_readiness	Survey	C	binary
field_chestPainInLastMonth	parq_readiness	Survey	C	binary
field_dizziness	parq_readiness	Survey	C	binary
field_heartCondition	parq_readiness	Survey	C	binary
field_jointProblem	parq_readiness	Survey	C	binary
field_physicallyCapable	parq_readiness	Survey	C	binary
field_prescriptionDrugs	parq_readiness	Survey	C	binary
field_activity1_intensity	physical_activity	Survey	C	ordinal
field_activity1_option	physical_activity	Survey	C	binary
field_activity1_time	physical_activity	Survey	C	continuous
field_activity1_type	physical_activity	Survey	C	categorical
field_activity2_intensity	physical_activity	Survey	C	ordinal
field_activity2_option	physical_activity	Survey	C	binary
field_activity2_time	physical_activity	Survey	C	continuous
field_activity2_type	physical_activity	Survey	C	categorical
field_atwork	physical_activity	Survey	C	ordinal
field_moderate_act	physical_activity	Survey	C	continuous
field_phys_activity	physical_activity	Survey	C	ordinal
vigorous_act	physical_activity	Survey	T	continuous
work	physical_activity	Survey	T	binary
field_riskfactors1	risk_perception	Survey	C	ordinal
field_riskfactors2	risk_perception	Survey	C	ordinal
field_riskfactors3	risk_perception	Survey	C	ordinal
field_riskfactors4	risk_perception	Survey	C	ordinal
GoSleepTime_categories	sleep	Survey	T	ordinal
WakeUpTime_categories	sleep	Survey	T	ordinal
field_GoSleepTime	sleep	Survey	C	continuous
field_WakeUpTime	sleep	Survey	C	continuous
field_sleep_diagnosis2	sleep	Survey	C	multi_categorical
field_sleep_time	sleep	Survey	C	continuous
field_sleep_time1	sleep	Survey	C	continuous
field_sleep_time_daily	sleep	Survey	C	continuous
sleep_diagnosis1	sleep	Survey	T	binary
sleep_time_categories	sleep	Survey	T	ordinal
field_device_activity_band	study_metadata	Survey	C	binary
field_device_iphone	study_metadata	Survey	C	binary
field_device_other	study_metadata	Survey	C	binary
field_device_smartwatch	study_metadata	Survey	C	binary
field_labwork	study_metadata	Survey	C	categorical
field_phone_on_user	study_metadata	Survey	C	ordinal
field_cannabisSmoking	tobacco_vaping_cannabis	Survey	C	ordinal
field_cannabisVaping	tobacco_vaping_cannabis	Survey	C	ordinal
field_currentCannabisSmoking	tobacco_vaping_cannabis	Survey	C	ordinal
field_currentCannabisVaping	tobacco_vaping_cannabis	Survey	C	ordinal
field_currentSmokeless	tobacco_vaping_cannabis	Survey	C	ordinal
field_currentSmoking	tobacco_vaping_cannabis	Survey	C	ordinal
field_currentVaping	tobacco_vaping_cannabis	Survey	C	ordinal
field_durationQuitSmokeless	tobacco_vaping_cannabis	Survey	C	categorical
field_durationQuitSmoking	tobacco_vaping_cannabis	Survey	C	categorical
field_durationQuitVaping	tobacco_vaping_cannabis	Survey	C	categorical
field_everQuitSmokeless	tobacco_vaping_cannabis	Survey	C	binary
field_everQuitSmoking	tobacco_vaping_cannabis	Survey	C	binary

Continued on next page.

Table 39 continued from previous page.

Label	Category	Source	Role	Type
field_everQuitVaping	tobacco_vaping_cannabis	Survey	C	binary
field_lastCannabisSmoking	tobacco_vaping_cannabis	Survey	C	ordinal
field_lastCannabisVaping	tobacco_vaping_cannabis	Survey	C	ordinal
field_onsetSmokeless	tobacco_vaping_cannabis	Survey	C	continuous
field_onsetSmoking	tobacco_vaping_cannabis	Survey	C	continuous
field_onsetVaping	tobacco_vaping_cannabis	Survey	C	continuous
field_pastCannabisSmoking	tobacco_vaping_cannabis	Survey	C	ordinal
field_pastCannabisVaping	tobacco_vaping_cannabis	Survey	C	ordinal
field_pastSmokeless	tobacco_vaping_cannabis	Survey	C	ordinal
field_pastVaping	tobacco_vaping_cannabis	Survey	C	ordinal
field_readinessQuitSmokeless	tobacco_vaping_cannabis	Survey	C	ordinal
field_readinessQuitSmoking	tobacco_vaping_cannabis	Survey	C	ordinal
field_readinessQuitVaping	tobacco_vaping_cannabis	Survey	C	ordinal
field_smokingHistory	tobacco_vaping_cannabis	Survey	C	binary
field_tobaccoProducts	tobacco_vaping_cannabis	Survey	C	multi_categorical
field_tobaccoProductsEver	tobacco_vaping_cannabis	Survey	C	multi_categorical
feel_worthwhile1	wellbeing	Survey	T	ordinal
feel_worthwhile2	wellbeing	Survey	T	ordinal
feel_worthwhile3	wellbeing	Survey	T	ordinal
feel_worthwhile4	wellbeing	Survey	T	ordinal
happiness	wellbeing	Survey	T	continuous
happiness_categories	wellbeing	Survey	T	ordinal
satisfiedwith_life	wellbeing	Survey	T	ordinal