A generative foundation model for five-class sleep staging with arbitrary sensor input

Hans van Gorp[1,2,✉], Merel M. van Gilst[1,3], Pedro Fonseca[1,2], Fokke B. van Meulen[1,3],
Johannes P. van Dijk[1,3,4], Sebastiaan Overeem[1,3], Ruud J. G. van Sloun[1]

ABSTRACT Gold-standard sleep scoring as performed by human technicians is based on a subset of polysomnographic sensor signals, namely the EEG, EOG, and EMG. Polysomnography, however, consists of many more signal derivations that could potentially be used to perform sleep staging, including cardiac and respiratory modalities. Leveraging this variety in signals would offer advantages, for example by increasing reliability, resilience to signal loss, and application to long-term non-obtrusive recordings. This paper proposes a deep generative foundation model for fully automatic sleep staging from a plurality of sensors and any combination thereof. We trained a score-based diffusion model with a transformer backbone using a dataset of 1947 expert-labeled overnight sleep recordings with 36 different signals, including neurological, cardiac, respiratory flow, and respiratory effort signals. We achieve zero-shot inference on any sensor set by using a novel Bayesian factorization of the score function across the sensors, i.e., it does not require retraining on specific combinations of signals. On single-channel EEG, our method reaches the performance limit in terms of polysomnography inter-rater agreement (5-class accuracy 85.6%, Cohen's kappa 0.791). At the same time, the method offers full flexibility to use any sensor set derived from other modalities, for example, as typically used in polygraphic home recordings that include finger PPG, nasal cannula and thoracic belt (5-class accuracy 79.0%, Cohen's kappa of 0.697), or by combining derivations not typically used for sleep staging such as the tibialis and sternocleidomastoid EMG (5-class accuracy 71.0%, Cohen's kappa of 0.575). Additionally, we propose a novel interpretability metric in terms of information gain per sensor and show that this is linearly correlated with classification performance. Lastly, our foundation model allows for post-hoc addition of entirely new sensor modalities by merely training a score estimator on the novel input.

INTRODUCTION

Sleep stage scoring is an essential tool in the clinical assessment of sleep and the diagnosis of sleep disorders. Traditionally, sleep staging has relied on overnight polysomnographic (PSG) recordings which at least include electroencephalography (EEG), electrooculography (EOG) and electromyography (EMG). The accepted gold standard is for experienced human scorers to perform this sleep staging manually, following the guidelines of the American Academy of Sleep Medicine (AASM) [1]. Accordingly, each 30-second segment of sleep, known as an epoch, is scored as belonging to one of five stages: Wake (W), Rapid Eye Movement (REM), or non-REM (NREM) stages 1-3 based on the visual recognition of established patterns on EEG, EOG and EMG signals. The representation of a sequence of sleep stages over the night is called a hypnogram. The visual analysis of its characteristics, such as the distribution and continuity of sleep stages help drive clinical interpretation [2].

There are several challenges in sleep scoring, namely its costs, its time requirements, and the need for trained personnel. In an effort to overcome these challenges, automatic sleep scoring based on PSG has been extensively described in literature. The EEG signal in particular, provides a strong basis to perform automatic sleep staging, and a single EEG derivation is often enough to reach performance on par with the human inter-rater agreement [3]. This is because most of the visual scoring rules for humans are based on the inspection of features in the EEG. While this alleviates some of the costs of human scoring, the EEG needs to be placed above the hairline which can cause patient discomfort, and due to its vulnerability to environmental noise and motion artifacts, is less well-suited for ambulatory, or prolonged measurements.

To provide an alternative, the measurement and analysis of surrogate physiological signals has been studied. Surrogate measurements make use of indirect measurements, such as movements often associated with wakefulness, or expressions of the sleep stages in autonomic nervous system activity, measured for example via cardiac and respiratory sensors. Surrogate measurement modalities often described in literature include actigraphy, cardiac activity, respiratory effort, and respiratory flow [4–6]. Because there are no visual sleep scoring rules for these signals, analysis must be performed automatically. Most successful approaches rely on the use of machine learning techniques on measurements of one or more signals, using as training ground-truth sleep stages derived from a human-scored, simultaneously recorded PSG study. Many approaches address a simplified sleep staging set-up, distinguishing only between sleep and wake, or distinguishing between 4 classes instead of the usual 5, where the N1 and N2 classes are merged into a joint N1/N2 class [4–12].

An unsolved problem remains: between different individual recordings, and more importantly, between measurement setups, the combination of available input signals can vary widely. This can be because of different devices, measurement protocols, sensors inadvertently being disconnected during the recording, or due to interference, noise, and artifacts. Some models described in the literature partially solve this issue and can perform sleep staging on a range of input signals. For example, U-Sleep has been trained specifically to work with any combination of single-channel EEG and single-channel EOG signals, even when using derivations between electrodes not recommended by the AASM [13, 14]. In the realm of surrogate measurement modalities, the CardioRespiratory Sleep Staging (CReSS) algorithm was developed to use any combination of instantaneous heart rate (IHR), respiratory effort, and respiratory flow signals [5]. Lastly, the SleepFM model [15] combines cardiac, respiratory, and brain activity signals using self-supervised learning.

However, no proposed system can straightforwardly scale up to new sensors after being trained on a specific set. Such a scale-up would not only require one to retrain the entire system but also the collection of a sufficiently large number of new recordings where all the old signals plus the newly desired signals are measured simultaneously. Since deep learning systems require substantial amounts of training data, the need for successive new rounds of data collection makes the addition of new sensors a practical obstacle to the development and introduction of new sensors in actual practice.

We introduce a deep generative foundation model to accurately and scalably perform sleep staging using any combination of signal modalities used to measure sleep, both PSG-derived as well as

[1] Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands. [2] Philips Sleep and Respiratory Care, Eindhoven, The Netherlands. [3] Sleep Medicine Centre, Kempenhaeghe Foundation, Heeze, the Netherlands. [4] Department of Orthodontics, Ulm University, Ulm, Germany. ✉ Corresponding author: h.v.gorp@tue.nl
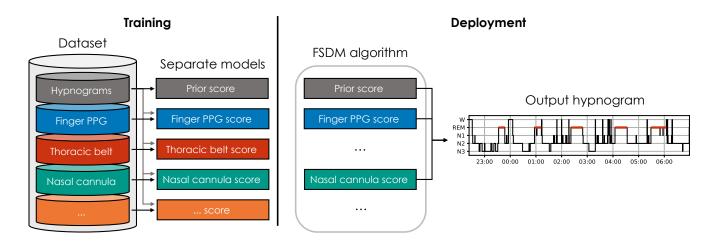
Figure 1: Visual overview of the proposed FSDM pipeline. During training, each signal-specific score-network is trained on the subset of data where its sensor was used, including the ground-truth hypnogram. During deployment, any combination of signal modalities can occur. Each signal that was present in the measurement is used with its specific score-network. The proposed FSDM algorithm then fuses the results with a prior score to obtain a posterior using equation (1), from which the hypnogram is sampled.

surrogate. Such a foundation model can be highly beneficial in clinical practice, as it has the flexibility to adapt to any circumstance and recording protocol, while at the same time being robust to disconnected sensors, missing measurements, and noisy sensors. Our approach employs a novel algorithm based on a Bayesian factorization of score-based diffusion models, which we term Factorized Score-based Diffusion Modeling (FSDM), see Fig. 1. In the proposed framework, score estimation networks are trained separately on each signal modality. Each signal-specific model is agnostic to the existence of other signals. Only during deployment are the different score estimation models combined into a joint posterior, which allows zero-shot inference on subjects with arbitrary combinations of measurement modalities, i.e., the model does not require retraining on specific combinations of signals. Additionally, the proposed framework permits a natural means of expressing the information gain from each sensor, calculated as the divergence between the likelihood of each sensor and the learned prior. The information gain can be leveraged as an interpretability metric and as a measure of the usefulness of each sensor for the sleep staging task. In conclusion, the proposed deep generative foundation model is highly scalable and enables the exploration of new sensors and sensor combinations at a relatively low cost, as the separate training eliminates the need to collect new data with all combinations present in every recording each time a new sensor is added.

RESULTS

Factorized Score-based Diffusion Models

To perform zero-shot inference using arbitrary combinations of input sensors, we use a score-based diffusion model as our generative backbone [16]. Rather than scoring epoch-by-epoch, this type of model generates the entire hypnogram given the input data. To do so, it requires an estimate of the posterior score. Our key result is that this posterior score is well estimated by:

$$\underbrace{\nabla_{\boldsymbol{y}} \log p\left(\boldsymbol{y}|X^{(1:N)}\right)}_{\text{posterior}} \approx \underbrace{\nabla_{\boldsymbol{y}} \log q_{\theta^{(0)}}(\boldsymbol{y})}_{\text{global prior}} + \frac{1}{N} \sum_{i=1}^{N}$$

$$\left(\underbrace{\nabla_{\boldsymbol{y}} \log q_{\theta^{(i)}}\left(\boldsymbol{y}|\boldsymbol{x}^{(i)}\right)}_{\text{individual likelihood}} - \underbrace{\nabla_{\boldsymbol{y}} \log q_{\theta^{(i)}}(\boldsymbol{y})}_{\text{individual prior}} \right), \quad (1)$$

where we have factorized the posterior score into its Bayesian components. In equation (1), $\boldsymbol{y}$ denotes the hypnogram, $X^{(1:N)}$ denotes the combination of input data coming from N different sensors, $\nabla_. \log p(.)$ denotes a true score, and $\nabla_. \log q_{\theta^{(i)}}(.)$ denotes a score as estimated by a neural network with parameters $\theta^{(i)}$,

which we will simply call a score-network for the sake of brevity. Each score-network is specific to an input signal with index i, for example, $i = 1$ could denote a respiratory effort signal and $i = 2$ a cardiac signal. We refer the reader to the methods section for the full derivation of the FSDM algorithm.

A crucial insight from equation (1) is that the posterior score can be inferred a-posteriori from individually learned likelihood and prior scores. Each of these individual scores is estimated by a score-network trained solely on its own sensor data using denoising score matching techniques. The score-networks are agnostic to the existence of other types of measurement data.

Dataset

To evaluate the proposed method on a large set of signals we leveraged the Sleep and OSA Monitoring with Non-Invasive Applications (SOMNIA) dataset [17] and the HealthBed dataset [18]. Both datasets comprise overnight polysomnographic recordings captured at Sleep Medicine Center Kempenhaeghe. The SOMNIA data comes from a diverse clinical population (1851 recordings), while the HealthBed comes from healthy participants without sleep disorders (96 recordings). A total of 1947 overnight recordings were thus included, of which 500 were used for hold-out testing. We extracted 18 different signal groups consisting of 36 individual signals from the recordings. A signal group refers to a group of signals with similar characteristics, such as the EOG group consisting of the E1-M2 and E2-M2 electrode derivations, and the respiratory inductance plethysmography (RIP) group consisting of the abdominal and the thoracic belt signals. We not only included the AASM-recommended signals used to perform visual sleep scoring, such as the EEG, EOG, and chin EMG, but also signals typically not used for sleep staging, such as peripheral oxygen saturation (SpO2) and EMG signals from the sternocleidomastoid (SCM) and the flexor digitorum superficialis (FDS). Two derived signal measures were also included, namely instantaneous heart rate (IHR) and instantaneous breathing rate (IBR), derived from the raw cardiac and respiratory signals, respectively. Additionally, we included 241 SOMNIA recordings where the subject used a PAP device, from which we measured PAP flow. For further details regarding the cohort demographics, we refer the reader to tables 1 and 2 in the methods section. Additionally, for further details regarding the extracted signals, we refer the reader to table 3, also in the methods section.

Evaluation

We first evaluated the proposed method in terms of agreement with the human scored hypnogram, where we tested each of the 36 signals individually and when used in combinations. Fig. 2 shows the resulting 5-class accuracy on the hold out test set.
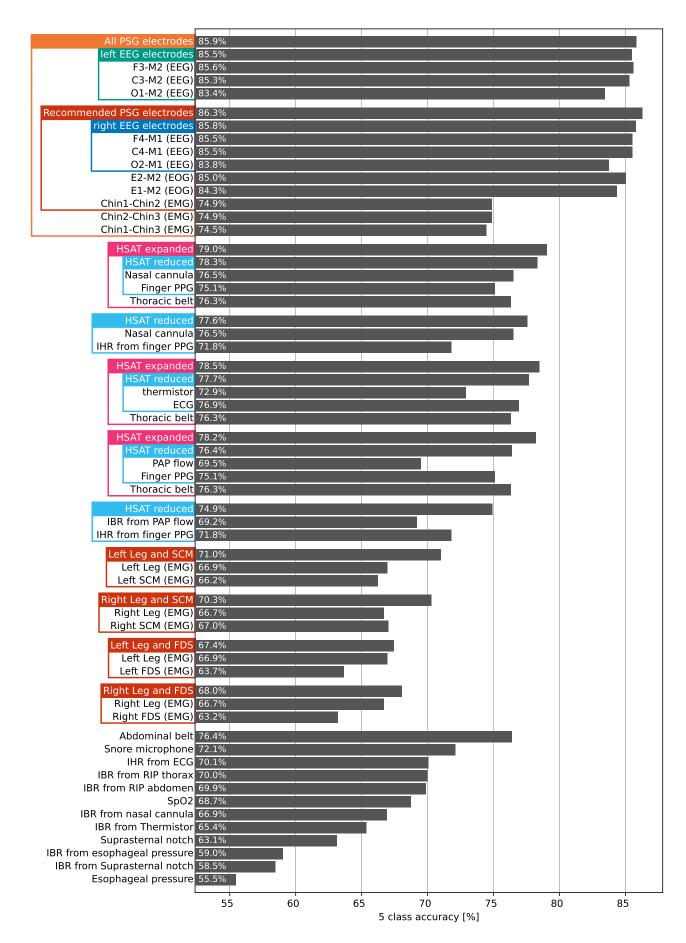
Figure 2: Average classification accuracy when using the model with separate signals and signal combinations. The colored boxes around the labels indicate all the signals that were part of a signal combination. For example, "right EEG electrodes" combines the signals F4-M1, C4-M1, and O2-M1.
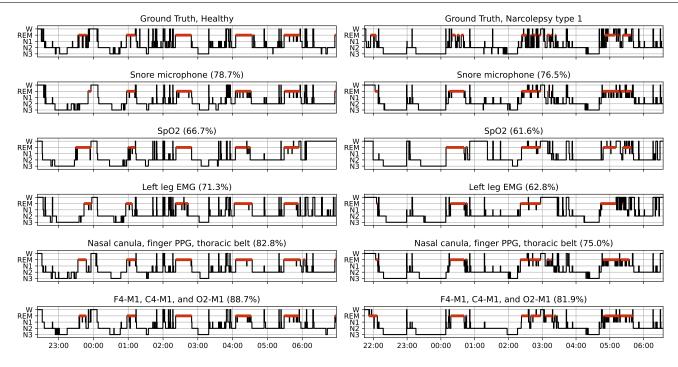
Figure 3: Qualitative examples of using five different signal combinations on a healthy subject (left column), and on a subject with narcolepsy type 1 (right column). The accuracy on each recording is listed between brackets.

Moreover, tables 4, 5, and 6 in the supplemental materials provide the accuracy and Cohen's kappa for 5- and 4-class sleep staging, and the per-class F1 scores. The resulting accuracies for the single-channel EEG and EOG models are the highest out of all the evaluated signals, achieving values between 83.4% and 85.6%, indicating that any of these signals on their own enables high quality sleep staging. Using these neurological signals together further improves performance, with the highest accuracy of 86.3% achieved when using the combination of signals for sleep staging recommended by the AASM [1].

Fig. 2 also shows the results for several signal combinations typically available with home sleep apnea tests (HSATs) [1]. These were split into reduced HSATs, which combine a cardiac signal with a respiratory flow signal, and expanded HSATS, which add a respiratory effort signal as well. We observe that using these signals in combination leads to better sleep staging accuracy as compared to using each of these signals on their own. Furthermore, the expanded HSATs improve upon their respective reduced sets, indicating that the respiratory effort signal further supplements the information available in the cardiac and respiratory flow signals. Similar results were found when combining two EMG signals, where the combination results in higher accuracy than each of the signals individually.

Qualitative results for a subset of the signals and signal combinations are shown in Fig. 3, which shows the results for a healthy subject and a subject with narcolepsy type 1 who displayed a sleep onset REM period (SOREMP), hich is one of the diagnostic criteria of that sleep disorder [19]. These examples were selected to illustrate performance on both healthy sleep and a sleep disorder that manifests in the hypnogram. It can be observed from Fig. 2 and Fig. 3 that unconventional signals can also be leveraged to perform sleep staging, albeit at a lower accuracy. For example, the SpO2 signal reaches an average 5-class accuracy of 68.7% and reasonably captures the overall shape of the hypnograms in Fig. 3. It however misses the SOREMP in the subject with narcolepsy of Fig. 3. The more conventional signal combinations fare much better in this regard, with especially the EEG combination clearly detecting the SOREMP.

Fig. 4 and Fig. 5 show two evaluations of the estimation of the overnight sleep statistics. Fig. 4 displays the Bland-Altman plots that result form estimating the overnight sleep statistics over all recordings in the hold-out test set using the AASM recommended PSG set-up as input. Fig. 5 shows four Bland-Altman plots for

specific combinations of overnight sleep statistic and input signal(s) evaluated only on the subjects in the hold-out test set with a certain disorder, which were chosen to highlight relevant use-cases in sleep medicine and research. For example, our method is able to measure the total sleep time for subjects with obstructive sleep apnea (OSA) using an HSAT, a parameter that is typically not captured by that measurement setup. We also show wake after sleep onset (WASO) estimation for insomnia subjects with a finger PPG, REM onset latency for narcolepsy subjects using a PSG set-up, and time in REM for subjects with REM sleep behavior disorder (RBD) measured with a single-channel EEG.

In general, the proposed method displays low bias in its estimation of the overnight statistics. In the estimation of REM onset latency however, some large outliers can be observed. These happen due to the all-or-nothing nature in the estimation of this statistic; if a bout of REM is missed by the method the REM onset latency will be postponed by an entire sleep cycle leading to over-estimations in the order of 100 minutes, while if the method puts a bout of REM sleep an entire cycle earlier this leads to under-estimations. A similar, albeit smaller, effect can be observed in the estimation of sleep onset latency.

To quantitatively evaluate our novel interpretability metric based on information gain, we calculated the average information gain for each single-sensor set-up and compared it to the classification accuracy it achieved on the hold-out test set. These results are shown in Fig. 6 on the left. The effects of noise and missing data on the information gain metric were also evaluated. To that end, we retrospectively removed segments from the ECG signal, replacing them with zeroes, or added Gaussian noise at a different signal-to-noise ratios (SNRs). The resulting information gain dropped, as can be observed in Fig. 6 on the right. The linear fit as found for the sensors still held for the retrospective addition of these artifacts, as the exact same line achieves an r-squared of 0.90. This same experiment for Cohen's kappa instead of accuracy can be found in the supplemental materials.

A qualitative example of the information gain metric is shown in Fig. 7, which displays the information gain per epoch when using PAP flow combined with finger PPG for a subject with obstructive sleep apnea (OSA). The information gain is calculated as the difference between the signal likelihood and the prior, where the prior estimates the probability of an entire hypnogram over the night as compared to hypnograms seen in the training set. Samples from the prior are shown in the supplemental materials.
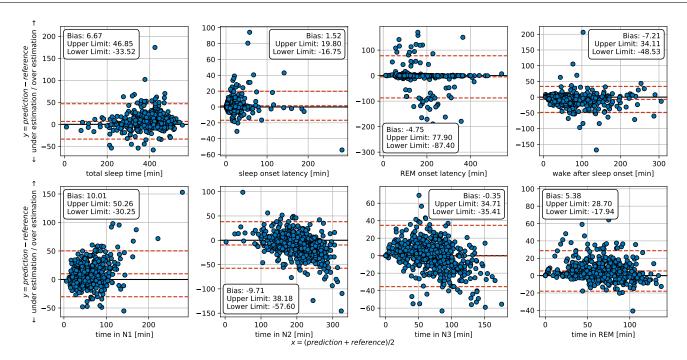
Figure 4: Bland-Altman plots for the overnight sleep statistics as predicted by the recommended PSG setup over all recordings in the hold-out test set. The limits of agreement are given at the 95% confidence interval.
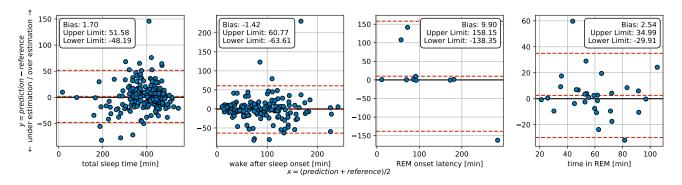

Figure 5: Bland-Altman plots of four combinations of sleep statistic, underlying sleep disorder, and input signal(s). The limits of agreement are given at the 95% confidence interval. From left to right: total sleep time for subjects with obstructive sleep apnea (OSA) as measured using an HSAT (Nasal Cannula + finger PPG + Thoracic Belt), Wake after sleep onset (WASO) [min] for subjects with insomnia as measured using finger PPG, REM onset latency [min] for subjects with narcolepsy as measured using the recommended PSG setup, time in REM [min] for subjects with REM sleep behavior disorder (RBD) as measured using single channel EEG (F4-M1).

Figure 6: Average information gain over the hold-out test set versus the average 5 class accuracy. Left, the average information gain per sensor over all test recordings shows a clear linear correlation with respect to the accuracy. The text boxes highlight the position of some of the sensors. Right, reducing the usefulness of the ECG signal by removing segments or adding noise reduces down-stream accuracy and information gain. The text boxes display the percentage of the recording removed or the SNR of the added noise. The linear relationship as fitted on the data from the left plot still provides a good fit here.
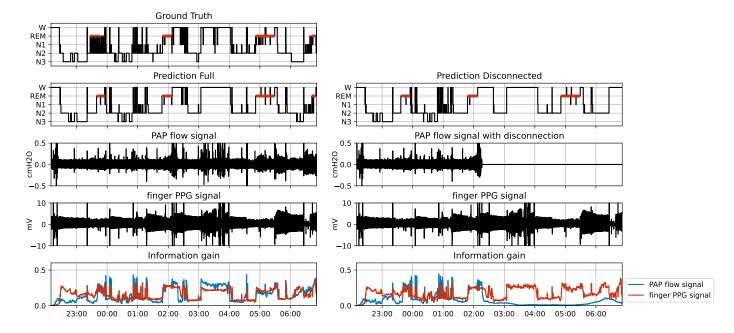
Figure 7: The proposed method is robust to the disconnection of sensors. Left, output of using the PAP flow and finger PPG signals over the entire night. Right, artificially created example of what would happen if the user took of their PAP device halfway through the night at 2:15 hours. Bottom row, our novel interpretability metric in terms of per-signal information gain. It can be observed that in the disconnection case, the information gain from the PAP flow goes to zero after 2:15 hours.

Fig. 7 shows that the information gain is low when the prediction coincides with the prior, for example when predicting wake at the start of the recording or N2 during the night, while information gain is high when it departs from the prior, for example when predicting a long sequence of wake in the middle of the night or a period of N3 at the end of the night.

Similar to the segment of zeroes as tested and illustrated in Fig. 6, a post-hoc simulation of sensor disconnection was introduced into the recording. This result is shown in the right column of Fig. 7. Here, we artificially simulated the scenario where the user of a PAP device takes of their mask during the night, in this case at 2:15 hours, while keeping the PPG device connected. The model elegantly handles this situation and is still able to perform adequate sleep staging, even with such a sensor disconnection. From the information gain plots in the last row of Fig. 7, it can also be observed how the information gained from the PAP flow signal goes to zero in the second half of the night, except for the final awakening, as the model has learned to correlate switching off of devices with awakenings. We can observe that the sleep staging prediction does suffer in quality from the sensor disconnection, as it for example misses the very last N3 and REM bouts between 6:00 and 7:00.

DISCUSSION

We introduced a generative foundation model for sleep staging with arbitrary sensor input. We have shown that the model can be applied to not only standard sleep staging signals, such as the EEG, or surrogate signals, such as the finger PPG, but also to unconventional ones, such as the SpO2 signal or the Leg EMG. Additionally, by leveraging the factorized score-based diffusion rule, the model can be applied to any combination of sensor inputs using separately trained models. This results in two highly desirable properties. Firstly, the proposed system can extend naturally to newly developed measurement modalities, as we only need to train one separate network solely on this new signal, after which it can be seamlessly adopted into our framework. This makes the framework highly scalable, because the integration of a new sensor does not require the collection of recordings with all sensors present. Secondly, the ad-hoc combination of sensors, especially surrogate sensors, opens up sleep staging to other fields of medicine, where sleep is largely under-studied but still of vital importance. For example, allowing identification of sleep disorders from Holter ECG in the context of cardiac arrythmias or tracking sleep based on vital signs monitored in the ICU.

The factorized score-based diffusion rule permits a natural means of expressing the information gained from each input signal by comparing its likelihood and prior terms. This is different from current interpretability approaches in automatic sleep staging, which can broadly be categorized into three strategies. Gradient-based approaches that leverage the Gradient-weighted Class Activation Mapping (Grad-CAM) algorithm [20–22], which relate how much each input element contributes to the classification output. Attention-based approaches, which characterizes which parts of the input space are used by the model for each decision, leveraging both 'soft' [23] and 'hard' attention [24]. Lastly, SHapley Additive exPlanations (SHAP) methods, which assign an additive importance value to each input feature [25, 26]. The information metric contrasts with the aforementioned approaches, as it is not concerned with relating the decision to each specific sample (or mini-epoch) in the input signals. Rather, it calculates for each epoch how much information each signal contributed, with information defined as a divergence from rational beliefs (the prior). Based on this, the average information gain per sensor can be calculated, which we found to be strongly correlated to its overall classification performance. Furthermore, as we have shown, the average information gain drops when a signal becomes less 'useful', for example during segments of missing data, or in the presence of additive noise.

In literature on automatic sleep staging, human inter-rater agreement serves as an upper limit on performance since it characterizes how consistent the ground truth is to which we are comparing the automatic sleep staging models [27]. The large-scale study by Rosenberg and Van Hout conducted based on the AASM inter-rater agreement program [28] found an average agreement of 82.6% using the scoring behavior of over 2,500 scorers. Because the data used in the present study came from one clinic, its inter-rater agreement serves as the upper limit. An average agreement of around 86% has been measured, based on both an internal institutional inter-rater agreement assessment and the AASM inter-rater agreement program. We verified this on the 111 recordings of the dataset where two scorings from different technicians were available, finding an average agreement of 85.8%. The proposed model reaches this upper limit for inputs of single-channel EEG, single-channel EOG, or combinations that include EEG/EOG. We

speculate that it is highly implausible that sleep staging performance for those inputs can be improved any further. For all other signals, it is much more difficult to ascertain whether the performance limit has been reached, as one would need to characterize exactly how much the model could be improved further (epistemic uncertainty), versus how much inherent sleep stage ambiguity is present in these signals (aleatoric uncertainty) [27]. In particular when using surrogate signals where visual scoring by humans is not possible, these limits have not been formally established.

The present work also offers some surprising new insights on the potential of different sensors for the sleep staging task. Firstly, the relatively good performance of the snoring microphone both quantitatively and qualitatively, see Fig. 2 and Fig. 3, respectively. The snoring microphone is placed in contact with the skin directly above the trachea and is typically only used to monitor snoring. However, we hypothesize that the vibrations caused by cardiac and respiratory activity are picked up by the microphone and enable sleep staging to be performed by our model. Secondly, the fact that it is possible to perform sleep staging using the SpO2 signal. While 5-class accuracy using the SpO2 signal is only 68.7% (Fig. 2), sleep-wake accuracy reaches 89.1% (Table 4 in the supplemental materials). Lastly, the sleep staging performances of using any of the single-channel EMG signals are surprisingly high, see Fig. 2. Typically, these sensors are only used to detect muscle atonia during the REM stage, but we show here that they carry information about all the sleep stages. The performance of using only EMG signals becomes even better when considering the EMG signal from two different muscle groups, such as the leg together with the sternocleidomastoid or the flexor digitorum superficialis.

The use of only an SpO2 signal or only an EMG signal to perform sleep staging has not been described in the literature before. In future work, it needs to be investigated whether the findings regarding their use for sleep staging hold across different acquisition setups. In the SOMNIA and HealthBed datasets, only minimal preprocessing is performed on the front-end of the sensors and all data is stored in its most raw form, with low-pass filtering, sampling rates and quantization specifications beyond those recommended by the AASM. This is in general not the case for data measured in many sleep laboratories, where forms of data compression, quantization, filtering, and resampling, are usually applied. One hypothesis is that sleep staging based on the SpO2 and EMG signals is possible in this set, because the minimal preprocessing leaves the possibility of desirable (insofar as sleep staging is concerned) contamination of cardiac, respiratory, and blood pressure signals. This effect has already been explored for EOG-based sleep staging, where EEG contamination can be leveraged to achieve high levels of agreement against manual scoring from PSG [29, 30], and in sleep staging based on suprasternal notch sensor, from which the cardiac and respiratory signals can be extracted and used for sleep staging [10, 31].

This work opens several avenues for future research. The model could be expanded to cover often-used wearables and nearables by training signal-specific networks for each of them, these include wrist-worn reflective PPG [4, 7, 8, 12], under-the-mattress sensors [6], microphones placed near the bed [32], and video [11, 18]. Care must be taken to synchronize these devices to the PSG from which the ground truth is derived, e.g. by matching the inter-beat intervals of two cardiac signals. Furthermore, the model could be applied to recordings coming from different clinics, enabling the evaluation of direct transfer and different transfer learning strategies [33, 34]. The model could also be applied to datasets with multiple scorings available per recording in order to evaluate to what degree the inter-rater agreement (of overnight sleep statistics [35]) is captured. Additionally, observing that the current method has a high computational cost as sampling from a score-based diffusion model required inferring multiple times from the neural networks, future work focusing on reducing the computational cost could consider hyper-parameter optimization, neural network pruning, consistency models [36, 37], and model distillation [38]. Lastly, while the FSDM framework is proposed here

to factorize the score over different sensor modalities, other factorizations could also be considered. For example, by factorizing over different patient populations, such as patients with intellectual disabilities or patients in an intensive care unit, one could tailor the automatic sleep staging model to their specific characteristics. A factorization across different sleep clinics could also be applied, which would enable a type of federated learning where clinics only need to share their final score model with one another and not the underlying training data.

To conclude, we developed a generative foundation model for the task of 5-class sleep staging that can use arbitrary (combinations of) sensor input. The unified framework allows for the direct comparison of different combinations of input measurements on sleep staging performance. Our proposed factorized solution is highly flexible, can be applied to a myriad of settings, and can easily be extended to new sensors while at the same time being robust to missing data. Furthermore, we proposed a novel interpretability metric based on information gain, allowing us to express at what time and by how much the model makes use of each signal for its decision. This work represents a fundamental step in the direction of a true universal sleep staging algorithm that goes beyond traditional fixed measurement set-ups and paves the way for more accessible and adaptable sleep analysis in diverse clinical populations and settings.

METHODS

Derivation of factorized score-based diffusion

This section introduces the theoretical underpinnings of our proposed Factorized Score-Based Diffusion Model (FSDM). We will first explain how we derive the posterior score, then we will detail how the individual scores can be learned, how to sample from an FSDM, and we will end with an explanation of our information gain calculation method. We will use the following notation:

- Let $y \in \mathcal{R}^{5 \times E}$ be a hypnodensity, i.e., the sleep stage probabilities, of size 5 stages by number of sleep epochs E.
- A hypnogram $h \in [W, N1, N2, N3, R]^E$ can be expressed as a hypnodensity y through one-hot encoding.
- Let $x \in \mathcal{R}^{E \cdot F \cdot 30}$ be a signal measured concurrently with the hypnogram at sampling frequency F.
- A collection of input signals can be written as:
$X^{(1:N)} = [x^{(1)}, x^{(2)}, \ldots, x^{(N)}]$.

Factorized posterior score

We are interested in estimating the probability distribution of hypnograms given a set of measurements signals, expressed as $p\left(y|X^{(1:N)}\right)$. This estimation problem can be re-written using Bayes' rule as:

$$p\left(y|X^{(1:N)}\right) = \frac{p(y)}{p\left(X^{(1:N)}\right)} p\left(X^{(1:N)}|y\right). \quad (2)$$

By assuming that the individual input signals $x^{(i)}$ are conditionally independent given y, we arrive at the Naive Bayes estimator:

$$p\left(y|X^{(1:N)}\right) = \frac{p(y)}{p\left(X^{(1:N)}\right)} \prod_{i=1}^{N} p\left(x^{(i)}|y\right). \quad (3)$$

Bayes' rule can then be applied a second time, but now to the individual likelihood terms, to arrive at:

$$p\left(y|X^{(1:N)}\right) = \frac{p(y)}{p\left(X^{(1:N)}\right)} \prod_{i=1}^{N} \frac{p\left(x^{(i)}\right)}{p(y)} p\left(y|x^{(i)}\right). \quad (4)$$

Equation (4) contains many difficult to calculate terms that do not depend on y, namely $p\left(X^{(1:N)}\right)$ and $p\left(x^{(i)}\right)$. To get rid of these, we can express the equation as a score:

$$\nabla_y \log p\left(y|X^{(1:N)}\right) = \nabla_y \log p(y) +$$

$$\sum_{i=1}^{N} \left(\nabla_y \log p\left(y|x^{(i)}\right) - \nabla_y \log p(y)\right). \quad (5)$$

It is thus possible to estimate the posterior score using only the individual conditional scores and the prior score. Of remark are two properties of equation (5). Firstly, if there is only 1 signal ($N = 1$), then it reads as a simple identity. Secondly, the term inside the summation, $\left(\nabla_{\boldsymbol{y}} \log p\left(\boldsymbol{y}|\boldsymbol{x}^{(i)}\right) - \nabla_{\boldsymbol{y}} \log p(\boldsymbol{y})\right)$, can be interpreted as: what additional information do we learn about $\boldsymbol{y}$ from $\boldsymbol{x}^{(i)}$, that was not already in the prior?

Score-based diffusion modeling
The posterior score calculated using equation (5) can be used to draw samples from $p\left(\boldsymbol{y}|X^{(1:N)}\right)$ by leveraging score-based diffusion modeling [16]. This type of generative model has garnered a lot of attention recently, due to its ease of training, stability, and high-fidelity outputs. We make use of the unifying framework proposed by Karras *et al.* [39], which we will briefly introduce.

Starting with an easy to sample from distribution $\boldsymbol{y}_0 \sim \mathcal{N}(0, \sigma_{max}^2 I)$, we use the factorized posterior score to progressively move towards more likely outputs in M discrete steps, until we approximate $\boldsymbol{y}_M \sim p_{data}$. This 'movement' is described by the following ordinary differential equation:

$$d\boldsymbol{y} = -\dot{\sigma}(t)\sigma(t)\nabla_{\boldsymbol{y}} \log p\left(\boldsymbol{y}|X^{(1:N)}\right) dt, \qquad (6)$$

where $\sigma(t)$ is known as the noise schedule, which defines the noise level at time t, and $\dot{\sigma}(t)$ is the first derivative with respect to t. To link the M discrete steps to the continuous time t, we use the following time schedule as proposed by Karras *et al.*:

$$t_m = \begin{cases} \left(\sigma_{max}^{1/\rho} + \frac{m}{M-1}(\sigma_{min}^{1/\rho} - \sigma_{max}^{1/\rho})\right)^{\rho} & \text{if } m \neq M \\ 0 & \text{if } m = M \end{cases} \qquad (7)$$

We empirically choose $\sigma_{min} = 0.0001$, $\sigma_{max} = 40$, $\rho = 7$, and $M = 32$. The noise schedule is simply chosen to be $\sigma(t) = t$, $\dot{\sigma}(t) = 1$.

Learning the individual conditional scores
In order to make use of equations (5) and (6), we need estimates of the individual conditional scores. To that end, we employ denoising score-matching [40, 41]. In this framework, the scores are only estimated at chosen time steps t and corresponding noise levels $\sigma(t)$. This is because noise-level specific scores are easier to estimate than the generic scores. Following Tweedie's approximation [42], the noise-level specific score estimates can be written as:

$$\nabla_{\boldsymbol{y}} \log p\left(\boldsymbol{y}|\boldsymbol{x}^{(i)}\right) \approx s_{\theta^{(i)}}\left(\boldsymbol{y}, \boldsymbol{x}^{(i)}, \sigma\right)$$
$$\approx \left(D_{\theta^{(i)}}\left(\boldsymbol{y}, \boldsymbol{x}^{(i)}, \sigma\right) - \boldsymbol{y}\right)/\sigma^2, \qquad (8)$$

where $D_{\theta^{(i)}}$ is a denoising function implemented using a deep neural network parameterized by $\theta^{(i)}$ and specific to the signal with index i. To train the denoising networks, $D_{\theta^{(i)}}$, we require a dataset of ground truth hypnograms, $\boldsymbol{y}$, with simultaneously acquired signals, $X^{(1:N)}$. We will call the generating dataset distribution as:

$$X^{(1:N)}, \boldsymbol{y} \sim p_{data}^{(1:N)}. \qquad (9)$$

In practice, not all signals will be measured in each recording. For example, in one recording in the dataset, sensors A-B-C might have been used, while for another recording, sensors B-C-D might have been used. To overcome this issue, each denoising network is trained only on the subset of recordings where its sensor was actually applied. We will denote these subsets as:

$$x^{(i)}, y \sim p_{data}^{(i)} \qquad (10)$$

Since $\boldsymbol{y}$ is categorical, we train the denoising networks with the expected cross entropy loss J over a range of σ values:

$$J_i = -\mathbb{E}_{x^{(i)}, y \sim p_{data}^{(i)}} \mathbb{E}_{\boldsymbol{n} \sim \mathcal{N}(0, \sigma^2 I)} \mathbb{E}_{\sigma}\big[$$
$$\boldsymbol{y} \log\left(D_{\theta_i}\left(\boldsymbol{y} + \boldsymbol{n}, \boldsymbol{x}^{(i)}, \sigma\right)\right)\big]. \qquad (11)$$

Throughout the training process, the noise level σ is drawn randomly using a log-normal distribution: $ln(\sigma) \sim \mathcal{N}(0.2, 1.4^2)$. This biases the denoiser to minimize the loss for medium levels of noise.

In summary, the individual conditional scores are approximated using denoising neural networks. These denoising neural networks are trained on the subset of recordings where their corresponding signals were measured. The loss function is the cross entropy loss, where the input to the denoiser is not only the measured signal x, but also a noisy version of the ground truth hypnogram $\boldsymbol{y} + \boldsymbol{n}$ as well as the noise level σ.

Learning the prior scores
Next to the individual conditional scores, we also need to estimate the prior scores. The prior score shows up two separate times in equation (5). The first time as a 'global' prior, since it is counted at the start, and the second time as an 'individual' prior, since it is subtracted from an individual likelihood score:

$$\nabla_{\boldsymbol{y}} \log p\left(\boldsymbol{y}|X^{(1:N)}\right) = \underbrace{\nabla_{\boldsymbol{y}} \log p(\boldsymbol{y})}_{\text{global prior}} +$$
$$\sum_{i=1}^{N} \left(\underbrace{\nabla_{\boldsymbol{y}} \log p\left(\boldsymbol{y}|\boldsymbol{x}^{(i)}\right)}_{\text{individual likelihood}} - \underbrace{\nabla_{\boldsymbol{y}} \log p(\boldsymbol{y})}_{\text{individual prior}}\right). \qquad (12)$$

As we discussed in the previous section, the dataset used for training does not have all signals available for all recordings. This has some implications for the prior score estimation. Specifically, if an individual likelihood score carries no new information it is desirable for the individual prior to exactly cancel it out, such that:

$$\underbrace{\nabla_{\boldsymbol{y}} \log p\left(\boldsymbol{y}|\boldsymbol{x}^{(i)}\right)}_{\text{individual likelihood}} = \underbrace{\nabla_{\boldsymbol{y}} \log p(\boldsymbol{y})}_{\text{individual prior}} \text{ if } I(\boldsymbol{y}; \boldsymbol{x}^{(i)}) = 0. \qquad (13)$$

In order to achieve this behavior, we estimate each individual prior in a similar vein as equation (8) as:

$$\underbrace{\nabla_{\boldsymbol{y}} \log p(\boldsymbol{y})}_{\text{individual prior}} \approx s_{\theta^{(i)}}\left(\boldsymbol{y}, \boldsymbol{0}, \sigma\right)$$
$$\approx \left(D_{\theta^{(i)}}\left(\boldsymbol{y}, \boldsymbol{0}, \sigma\right) - \boldsymbol{y}\right)/\sigma^2, \qquad (14)$$

where we replaced the input signal $\boldsymbol{x}^{(i)}$ using a vector of zeroes. Additionally, we also supplement the training by augmenting the loss as specified in equation (11). With probability $p_{\text{augment}} = 0.5$ we partially replace the input signal $\boldsymbol{x}^{(i)}$ with some zeroes, and with probability $p_{\text{zero}} = 0.1$ we completely replace it with zeroes. This ensures that we can use each signal specific denoising network both as a conditional likelihood score estimator, as well as an individual prior estimator. Where the individual likelihoods and priors have been trained on the same subset of data.

Contrary to the individual priors, the intuition behind the global prior is that it should be trained on the largest set of possible hypnograms. To that end we train one separate global prior on the entire datset, since it is not constrained by sensor availability. The global prior is equal to:

$$\underbrace{\nabla_{\boldsymbol{y}} \log p(\boldsymbol{y})}_{\text{global prior}} \approx s_{\theta^{(0)}}\left(\boldsymbol{y}, \boldsymbol{0}, \sigma\right)$$
$$\approx \left(D_{\theta^{(0)}}\left(\boldsymbol{y}, \boldsymbol{0}, \sigma\right) - \boldsymbol{y}\right)/\sigma^2, \qquad (15)$$

Where $D_{\theta^{(0)}}$ is trained using the following loss:

$$J_0 = -\mathbb{E}_{\boldsymbol{0}, y \sim p_{data}^{(0)}} \mathbb{E}_{\boldsymbol{n} \sim \mathcal{N}(0, \sigma^2 I)} \mathbb{E}_{\sigma}\big[$$
$$\boldsymbol{y} \log\left(D_{\theta^{(i)}}\left(\boldsymbol{y} + \boldsymbol{n}, \boldsymbol{0}, \sigma\right)\right)\big], \qquad (16)$$

where $\boldsymbol{0}, y \sim p_{data}^{(0)}$ covers the entire dataset. In summary, the training of the prior networks are special cases of the conditional likelihood networks where the input signals x are (partially) set to zero. An overview of the training loop is given as pseudo-code in algorithm 1.

Algorithm 1: Training a single score network

Require: signal index $i \in [0, 1, \ldots, N]$, Densoising function $D_{\theta^{(i)}}$, dataset sampler $p_{data}^{(i)}$, noise sampling scheme p_σ, optimizer $opt()$, probabilities p_{augment} and p_{zero}

1 **while** *not converged* **do**
   // expectation through Monte-Carlo
2   sample $x^{(i)}, y \sim p_{data}^{(i)}$
3   sample *augment* $\sim p_{\text{augment}}$, *zero* $\sim p_{\text{zero}}$
4   sample $\sigma \sim p_\sigma$
5   sample $n \sim \mathcal{N}(0, \sigma^2 I)$
   // augment the data
6   **if** *augment* **then**
7       $k \sim u(1, |x^{(i)}|), l \sim u(1, |x^{(i)}|)$
8       $x^{(i)}[k : l] = 0$
9   **if** *zero* **then**
10       $x^{(i)} = 0$
   // Loss calculation and optimizer step
11     $y_{noisy} = y + n$
12     $y_{denoised} = D_{\theta^{(i)}}\left(y_{noisy}, x^{(i)}, \sigma\right)$
13     $J_i = -\sum y \log y_{denoised}$
14     $opt(J_i, D_{\theta^{(i)}})$
Return: $D_{\theta^{(i)}}$

Algorithm 2: Sampling from an FSDM

Require: Measured signals $X = [x_1, x_2, \ldots, x_N]$, Denoising functions $D = [D_{\theta^{(0)}}, \ldots, D_{\theta^{(N)}}]$, noise schedule $\sigma(t)$, time $t_{m \in \{0, 1, \ldots, M\}}$, projection function $\tau()$

// Factorized score calculation
1 **Function** FS(y_{nosiy}, σ):
2     $y_{denoised} = D_{\theta^{(0)}}(y_{nosiy}, \sigma, 0)$
3   **foreach** $x^{(i)} \in X$ **do**
4   likelihood $= D_{\theta^{(i)}}(y_{nosiy}, \sigma, x^{(i)})$
5   individual prior $= D_{\theta^{(i)}}(y_{nosiy}, \sigma, 0)$
6       $y_{denoised} = y_{denoised} + \lambda($ likelihood $-$
7   individual prior $)$
8     $y_{denoised} = \tau(y_{denoised})$
9   score $= (y_{denoised} - y_{nosiy})/\sigma^2$
10   **return** score

// Main sampling algorithm
11 sample $y_0 \sim \mathcal{N}(0; \sigma(t_0)^2 I)$
12 **for** $m = 0$ **to** $M - 1$ **do**
   // Gradient step
13     $dy = -\sigma(t_m)$ FS($y_m, \sigma(t_m)$)
14     $y_{m+1} = y_m + (t_{m+1} - t_m)dy$
   // Second order correction
15   **if** $\sigma(t_{m+1}) \neq 0$ **then**
16       $dy' = -\sigma(t_{m+1})$ FS($y_{m+1}, \sigma(t_{m+1})$)
17       $y_{m+1} = y_m + 0.5(t_{m+1} - t_m)(dy + dy')$
Return: y_M

Sampling from an FSDM

We can now rewrite equation (5) to use the estimated scores from equations (8), (14), and (15):

$$D_{all}\left(y, X^{(1:N)}, \sigma\right) = D_{\theta^{(0)}}\left(y, 0, \sigma\right) +$$
$$\lambda \sum_{i=1}^{N} \left(D_{\theta^{(i)}}\left(y, x^{(i)}, \sigma\right) - D_{\theta^{(i)}}\left(y, 0, \sigma\right)\right), \quad (17)$$

where D_{all} is the combined denoising function. Additionally, we have introduced a weighting term λ that specifies the importance of the likelihood terms with respect to the prior, which is common practice in both Bayesian inference and diffusion guidance [43, 44]. We empirically choose $\lambda = 1/N$, which gives rise to the desirable property that adding the same signal once or many times leads to the same posterior score estimate.

In practice, combining score estimates from multiple models can lead to instability in the sampling process, as the current estimate at a time-step can 'fall off the manifold'. A lot of research has been done on this effect for image restoration and multiple solutions have been found [45–48]. However, these methods assume there is some (partially) sampled data, coupling the diffusion process via a likelihood function, which is not the case for our setup as the hypnograms are completely unknown a-prior. We thus propose a new manifold projection step suited for categorical data. Since we know that on the denoised end-estimate manifold, all classes should count up to a total of probability 1, we use:

$$\tau(y) = y / \sum_{j=1}^{5} y_j. \quad (18)$$

In other words, we re-normalize the hypnodensity to follow the hypnodensity manifold constraint.

After applying the manifold projection step, we can use the end-estimate together with Tweedie's formula to get the posterior score estimate as:

$$p\left(y|X^{(1:N)}\right)\Big|_\sigma \approx \left(\tau\left(D_{all}\left(y, X^{(1:N)}, \sigma\right)\right) - y\right)/\sigma^2. \quad (19)$$

Following [39] we employ Heunn's second order method to solve the ODE as specified in equation (6) using the posterior score estimate of equation (19). This leads to the sampling process as specified in algorithm 2. A visual overview of the FSDM rule and the evolution of samples over time is shown in Fig.8.

To generate the final hypnograms that are shown to the user and which were compared to the ground truth, we sample 64 times from the FSDM algorithm. This results in 64 different realizations of the posterior distribution $y \sim p_\theta(y|X)$, i.e., hypnograms that are likely given the input data. The end result is then calculated as the majority vote of these hypnograms per epoch:

$$\hat{h} = \arg\max \mathbb{E}_{y \sim p_\theta(y|X)}[y], \quad (20)$$

where the arg max is applied along the first dimension of $y \in \mathcal{R}^{5 \times E}$, resulting in a hypnogram with categorical sleep stages $\hat{h} \in [W, N1, N2, N3, R]^E$. Additionally, the 64 samples are used to separately calculate the overnight sleep statistics, similar to previous work [35]. The final value for each overnight statistic for each recording is then calculated as the median of the individual realizations:

$$stat = \text{median}_{y \sim p_\theta(y|X)}[f_{stat}(y)], \quad (21)$$

where f_{stat} refers to the function that calculates the overnight statistic of interest from a hypnogram, e.g. total sleep time or wake afer sleep onset.

Information

The factorized combination rule from equations (5) and (17) allows for the evaluation of how much each individual measurement source contributes to the end-result. This can be seen through the lens of 'information' as defined by Caticha 2011: *"Information is what forces a change of rational beliefs"* [49]. In our case, the 'rational belief' can be interpreted as the prior score, whereas the amount of change is expressed as the difference between the likelihood score and the prior score.
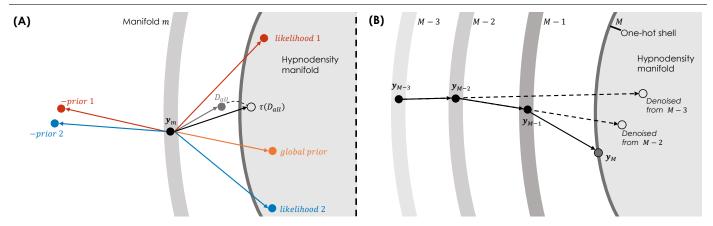
Figure 8: Visualization of the sampling process for an FSDM model. **(A)** From a current point $\boldsymbol{y}_m$ we estimate two likelihoods, two priors, and one global prior. Combining them all leads to a denoised estimate outside the hypnodensity manifold, which is corrected using a projection step $\tau()$. **(B)** Evolution of a sample over the last three time-steps. The end-estimate progressively moves from the hypnodensity manifold to the one-hot shell, which is congruent with how human scores score one-hot hypnograms.

There are different domains and distance functions that we could use to compare the likelihood and the prior in order to express the amount of information gain. We here choose to follow recent literature on hypnodensity, which proposes the use of the cosine distance between the two vectors of class probabilities at each epoch [50, 51].

To express the amount of information, we calculate the expected cosine distance between likelihood and prior over the entire sampling trajectory:

$$\boldsymbol{b}_i = \mathbb{E}_{\boldsymbol{y}_0}\left[\frac{1}{M}\sum_{m=1}^{M}\left(\text{cos. dist.}\left(D_{\theta^{(i)}}\left(\boldsymbol{y}_m, \boldsymbol{x}^{(i)}, \sigma(t_m)\right),\right.\right.\right.$$
$$\left.\left.\left. D_{\theta^{(i)}}\left(\boldsymbol{y}_m, \boldsymbol{0}\ , \sigma(t_m)\right)\right)\right)\right], \quad (22)$$

where we take the expectation over different initial states of the sampling process $\boldsymbol{y}_0 \sim \mathcal{N}(0; \sigma(t_0)^2 I)$. Additionally, $\boldsymbol{b}_i$ is the information for the signal with index i, and 'cos. dist.' is the cosine distance between the two hypnodensities as estimated by the prior and likelihood denoising functions. This resulting information will thus be of similar length as the hypnogram and take values between 0 and 1, i.e. $\boldsymbol{b}_i \in [0, 1]^E$. An information of 1 means that the likelihood and prior completely disagreed over the entire sampling process, and a 0 means that they always agreed, in which case one could just as easily not measured the signal at all.

Demographic details of the datasets

The overnight recording data used in this study came from the SOMNIA [17] and HealthBed datasets [18]. Each recording consisted of a full PSG following the AASM setup [1], which was subsequently scored by a human technician. Additionally, each recording in the SOMNIA set was accompanied by at least one additional surrogate sensor modality, such as a supranational notch sensor or additional electromyograms. We included a total of 1947 recordings obtained in the period between 2017-01-01 and 2023-10-10. 1851 recordings came from the SOMNIA set and 96 recordings came from the HealthBed set. We did not include any pediatric recordings. No other exclusion criteria were applied. We randomly split the recordings into 1347 train, 100 validation, and 500 test recordings. Table 1 show the demographic data for each random split, including whether a PAP device was used during the night. The distribution of primary sleep disorder groups is shown in Table 2. Note that many subjects had multiple primary sleep disorders, thus the columns of the table count up to more than the total number of recordings in each split. To avoid very small groups, the diagnostic categories in Table 2 were created by merging similar diagnosis together, e.g. the narcolepsy category includes both subjects with narcolepsy type 1 and type 2. We provide a full breakdown per diagnosis and to what diagnostic group it relates in the supplemental material.

Table 1: Demographic parameters for the two datasets. '#' refers to number, and 'std.' refers to the standard deviation.

	Parameter		Total	Train	Val	Test
SOMNIA [17]	Recordings	[#]	1851	1281	97	473
	Female	[#]	710	499	30	181
		[%]	38.4	39.0	30.9	38.3
	Age	[mean]	51.0	50.7	52.5	51.5
		[std.]	15.7	16.1	15.7	14.7
	BMI	[mean]	25.9	25.8	25.9	26.4
		[std.]	8.2	8.2	8.9	8
	PAP usage	[#]	241	163	12	66
		[%]	13.0	12.7	12.4	14.0
HealthBed [18]	Recordings	[#]	96	66	3	27
	Female	[#]	60	44	2	14
		[%]	62.5	66.7	66.7	51.9
	Age	[mean]	36.0	35.9	33.7	36.5
		[std.]	13.5	13.3	12.7	14.1
	BMI	[mean]	24.3	24.0	23.7	25.2
		[std.]	3.2	2.9	3.8	3.7

Table 2: Primary sleep disorder diagnoses over the three splits. Note that many subject had multiple primary sleep diagnoses.

Diagnosis	Total	Train	Val	Test
Insomnia disorders	613	418	29	166
Obstructive sleep apnea	1037	698	59	280
Central sleep apnea	42	26	2	14
Treatment emergent-central sleep apnea	6	6	0	0
Hypoventilation	8	6	0	2
Narcolepsy	31	21	0	10
Other hypersomnolence-disorders	54	40	3	11
Insufficient sleep syndrome	66	52	3	11
Circadian rythm disorder	46	34	4	8
NREM Parasomnias	115	85	5	25
REM sleep behavior disorder	122	84	8	30
Other REM Parasomnias	55	47	2	6
Other Parasomnias	45	31	3	11
RLS/PLMD	268	198	10	60
Other movement disorders	58	37	3	18
Other sleep disorders	16	11	2	3
No primary sleep diagnosis-and/or normal variants	99	76	5	18
Healthy	96	66	3	27

Table 3: Overview of the signals extracted from the datasets. We clustered the signals into groups such as EEG and RIP belts. The same preprocessing parameters were used for all the signals that are in the same group. 'HP' and 'LP' denote the high-pass and low-pass filter respectively where we show the cut-off frequency in Hz.

Signal group	Signals	Unit	Scale	HP	LP	Total	Train	Val	Test
EEG	F3-M2, F4-M1, C3-M3, C4-M1, O1-M2, O2-M1	V	10^4	0.3	49	11681	8081	600	3000
EOG	E1-M2, E2-M2	V	10^4	0.3	49	3886	2689	200	997
EMG chin	Chin1-Chin2, Chin1 Chin3, Chin2-Chin3	V	10^4	10	49	5838	4038	300	1500
ECG	ECG	V	10^3	0.3	49	1947	1347	100	500
RIP belts	Abdomen, Thorax	V	10^{-2}	0.1	15	3892	2692	200	1000
Thermistor	Thermistor	V	10^4	0.1	15	1706	1184	88	434
Nasal cannula	Nasal cannula	cmH2O	1	0.03	49	1706	1184	88	434
PAP flow	PAP flow	cmH2O	10	0.03	49	241	163	12	66
Suprasternal notch	Suprasternal notch	V	10	0.03	49	289	199	18	72
Esophageal pressure	Esophageal pressure	mmHg	10^{-1}	0.03	49	97	65	8	24
Snore microphone	snore microphone	V	10^3	10	49	1947	1347	100	500
Finger PPG	Finger PPG	V	10^{-2}	0.3	49	1976	1364	101	511
SpO2	SpO2	%	10^{-2}	-	-	1944	1344	100	500
EMG FDS	FDS L, FDS R	V	10^4	10	49	508	368	20	120
EMG legs	Leg L, Leg R	V	10^4	10	49	3894	2694	200	1000
EMG SCM	SCM L, SCM R	V	10^4	10	49	296	206	24	66
Instantaneous heart rate	ECG, PPG	Bpm	$1/60$	-	-	3923	2711	201	1011
Instantaneous breath rate	RIP Belts, Thermistor, Nasal cannula, PAP flow, Suprasternal notch, Esophageal pressure	Brpm	$1/60$	-	-	8163	5642	426	2095

Signal extraction

A total of 36 signals were extracted from the recordings of the datasets. These were grouped into 18 clusters of similar type. For example, the signals F3-M2, F4-M1, C3-M3, F4-M1, O1-M2, and O2-M1 were all grouped into the EEG type, and the signals from the abdominal belt and the thoracic belt were grouped into a common Respiratory Inductance Plethysmography (RIP) belt type. Table 3 shows each of the 36 signals and their corresponding type.

The PAP flow signal was measured during overnight recordings where the subject used a PAP device, which was the case for 241 recordings, see Table 1. Subjects were allowed to bring their personal PAP device, resulting in a large variety of types and manufacturers. The types of PAP included in the study were continuous PAP (208), automatic PAP (23), adaptive servo ventilation (2), and Bi-Level PAP(8). Typically, these devices allow for some type pressure or flow readout. To homogenize this readout between the different devices, a common third-party sensor was attached to the breathing tube of the PAP device, called the pneumo flow (Braebon, Canada). This readout is used in the study as the 'PAP flow' signal. For details regarding how all the other signals were measured, we refer the reader to the SOMNIA protocol paper [17].

Preprocessing

We applied a common preprocessing pipeline to all signals as shown in Fig. 9. We will briefly describe each preprocessing operation. First, each of the signals is scaled by a constant value in order to bring its approximate magnitude around 1. The scaling factor is chosen specific to each signal type and is shown in Table 3. We for example scale all the EEG channels by a factor 10^4, making it so that an amplitude of $100\mu V$ corresponds to the value 1 and the slow-wave amplitude threshold of $75\mu V$ corresponds to a value of 0.75 [1]. This scaling enables faster training of the neural networks.

Second, we identify missing values in the signals as those locations where they are exactly equal to 0. We linearly interpolate these values by using neighbouring sample values. This interpolation is performed in order to reduce filtering artifacts that can occur

by the large jumps in magnitude that these missing values cause. After all the filtering operations, the samples with the missing values are reset to exact zeroes, this enables the neural networks to understand which values are missing in the signal.

Third, we resample all signals to 128 Hz using a polyphase filter, except for the SpO2 signal which was up-sampled from 32 Hz to 128 Hz using a sample and hold scheme. After resampling, we apply a low-pass and a high-pass filter, both implemented using fifth order butterworth filter. The high-pass and low-pass filter settings are signal type specific and reflect the recommendations of the AASM manual [1] with some minor adjustments to the low-pass filter settings.

After filtering we reset the samples values of the missing value indices back to zero. We then clip the signals between -5 and 5. For example, the EEG signals are clipped between $-500\mu V$ and $+500\mu V$. Lastly, we zero-pad all the signals to a common length of 1792 sleep epochs, or $1792 \cdot 30 \cdot 128 = 6,881,280$ samples. This zero-padding was solely done for implementation purposes, as it allows to stack the signals of different nights, and the zero-padded segment was not used to calculate any of the overnight statistics or performance metrics such as accuracy and Cohen's kappa.

In the case that the extracted signal was a cardiac or respiratory signal, such as ECG, finger PPG, or RIP Belt, we also extracted the instantaneous heart-rate (IHR) or the instantaneous breath-rate (IBR). This extraction was performed by extracting the peaks of the cardiac pulses and the breaths using the automatic multi-scale peak detection algorithm [52]. We used the following settings: window length = 600 seconds, window overlap = 120 seconds, and maximum scale = 5 seconds for cardiac signals, while maximum scale = 60 seconds for respiratory signals. After peak detection, we convert the peaks to the inter-beat and inter-breath interval length using a sample and hold scheme. To reduce the impact of artifacts, biologically implausible intervals are removed from the sequence. We remove all cardiac inter-beat intervals outside the range of [0.3s - 2s], and we remove all respiratory inter-breath intervals outside the range of [1s - 30s]. As a final step, the intervals are converted into the IHR or IBR. While typically expressed in beats
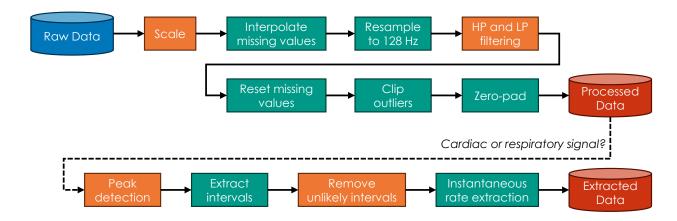
Figure 9: The preprocessing pipeline is the same for all the signals. The orange blocks are set by signal-type specific parameters, e.g. the cutoff frequencies of the filters can be different for different signal types, see Table 3. If a signal is a cardiac or respiratory signal we also extract the instantaneous heart-rate or breathing-rate.

per minute (Bpm) or breaths per minute (Brpm), we scale the signals by 1/60 to get the beats/breaths per second, resulting in a better magnitude for use by the neural networks. Fig. 10 shows an example of how we extract the peaks of a finger PPG signal, which we then convert to inter-beat intervals, to subsequently convert to the IHR.

Neural network architecture

Our method is agnostic to the exact architecture used for each denoising neural network. In this manuscript, we leverage the DDPM++ model as implemented by Karras *et al.* [39], and modified to work on 1D timeseries in our previous work on EOG-driven sleep staging [53]. See Fig. 11 for a visualization of the model architecture. In this section, we highlight the most important modifications made to the original DDPM++ model, see the supplemental material for a complete overview of the model implementation.

Firstly, The model takes as input not only the current sample point y_{m-1} and noise level $\sigma(t_m)$, but also a conditioning created from the measured signal as $c^{(i)} = \text{enc}(x^{(i)}$. This conditioning is of the same size as y_{m-1} and appended channel-wise to it as input to the DDPM++ model. The conditioning networks, enc(), are implemented using the ResNet blocks that make up the backbone of the DDPM++ model. Using 5 levels, with 2 ResNet blocks per level, and a final strided convolution, these conditioning networks compress the input signals $x^{(i)} \in \mathcal{R}^{1792 \cdot 30 \cdot 128}$ to conditioning vectors $c^{(i)} \in \mathcal{R}^{1792 \times 16}$, i.e. a length of 1792 with 16 channels. Note that the ResNet blocks typically use a timestep embedding, but these are not added to the epoch encoder. This speeds up the sampling process, as the epoch encoder only needs to be run once, and its output context vector can be cached.

Secondly, the DDPM++ model makes use of self-attention, to which we added positional encoding using sine-cosine embedding, creating a transformer encoder layer. The transformer architecture enables the model to learn the temporal relations within and between the signals and hypnograms.

Thirdly, we adapted DDPM++ to work on 1D time-series, using 1D convolutions of kernel size 7 with 32 channels. We used 4 resolution levels with a down-sampling stride of 4. We applied a transformer layer at all resolution levels.

Data availability

The SOMNIA data [17] and HealthBed data [18] used in this study are available from the Sleep Medicine Centre Kempenhaeghe upon reasonable request. The data can be requested by presenting a scientific research question and by fulfilling all the regulations concerning the sharing of the human data. The details of the agreement will depend on the purpose of the data request and the entity that is requesting the data (e.g. research institute or corporate). Each request will be evaluated by the Kempenhaeghe Research Board and, depending on the request, approval from independent medical ethical committee might be required. Access to data from

Figure 10: Example of IHR extraction from a finger PPG signal. The red dots denote the peaks found by the automatic multi-scale peak detection algorithm.

outside the European Union will further depend on the expected duration of the activity; due to the work required from a regulatory point of view, the data is less suitable for activities that are time critical, or require access in short notice. Specific restrictions apply to the availability of the data collected with sensors not comprised in the standard PSG set-up, since these sensors are used under license and are not publicly available. These data may however be available from the authors with permission of the licensors. For inquiries regarding availability, please contact Merel van Gilst (M.M.v.Gilst@tue.nl).

REFERENCES

[1] C. Iber, S. Ancoli-Israel, A. Chesson, and S. F. Quan, *The AASM manual for the scoring of sleep and associated events: rules, terminology and technical specifications*, American Academy of Sleep Medicine, Westchester, IL, USA, 2007.

[2] C. van der Woerd, H. van Gorp, S. Dujardin, M. Sastry, H. Garcia Caballero, F. B. van Meulen, S. van den Elzen, S. Overeem, and P. Fonseca, "Studying sleep: towards the identification of hypnogram features that drive expert interpretation," *Sleep*, vol. 47, no. 3, pp. zsad306, 12 2023.

[3] H. Phan and K. Mikkelsen, "Automatic sleep staging of EEG signals: recent development, challenges, and future directions," *Physiological Measurement*, vol. 43, no. 4, pp. 04TR01, April 2022.

[4] P. Fonseca et al., "A computationally efficient algorithm for wearable sleep staging in clinical populations," *Scientific Reports*, vol. 13, no. 1, pp. 9182, June 2023.

Figure 11: Overview of a the neural network used for each denoiser $D_{\theta^{(i)}}\left(\boldsymbol{y}_m, \boldsymbol{x}^{(i)}, \sigma(t_m)\right)$. The signal, $\boldsymbol{x}^{(i)}$, is used as the inital input to the network and encoded into a context vector. This is then stacked with the sample at the current timestep $\boldsymbol{y}_m$ and fed though a U-Net structure. At the end, a hypnodensity is given as output through the use of a softmax activation function. The current noise variance, $\sigma(t_m)$, is additionally embedded into a timestep embedding, which is added inside the ResNet layers of the U-Net encoder and U-Net decoder. To avoid needing to run the Epoch Encoder M times, the timestep embedding is not added to its ResNet layers.

[5] J. P. Bakker, M. Ross, R. Vasko, A. Cerny, P. Fonseca, J. Jasko, E. Shaw, D. P. White, and P. Anderer, "Estimating sleep stages using cardiorespiratory signals: validation of a novel algorithm across a wide range of sleep-disordered breathing severity," *Journal of Clinical Sleep Medicine*, vol. 17, no. 7, pp. 1343–1354, July 2021.

[6] H. Zhai, Y. Yan, S. He, P. Zhao, and B. Zhang, "Evaluation of the accuracy of contactless consumer sleep-tracking devices application in human experiment: A systematic review and meta-analysis," *Sensors*, vol. 23, no. 10, May 2023.

[7] B. M. Wulterkens, P. Fonseca, L. W. A. Hermans, M. Ross, A. Cerny, P. Anderer, X. Long, J. P. van Dijk, N. Vandenbussche, S. Pillen, M. M. van Gilst, and S. Overeem, "It is all in the wrist: Wearable sleep staging in a clinical population versus reference polysomnography," *Nature and Science of Sleep*, vol. 13, pp. 885–897, June 2021.

[8] N. Sridhar, A. Shoeb, P. Stephens, A. Kharbouch, D. B. Shimol, J. Burkart, A. Ghoreyshi, and L. Myers, "Deep learning for automated sleep staging using instantaneous heart rate," *npj Digital Medicine*, vol. 3, no. 1, pp. 106, Aug 2020.

[9] G. Garcia-Molina and J. Jiang, "Interbeat interval-based sleep staging: work in progress toward real-time implementation," *Physiological Measurement*, vol. 43, no. 2, pp. 025004, 2022.

[10] L. Cerina, S. Overeem, G. B. Papini, J. P. van Dijk, R. Vullings, F. B. van Meulen, M. Ross, A. Cerny, P. Anderer, and P. Fonseca, "A sleep stage estimation algorithm based on cardiorespiratory signals derived from a suprasternal pressure sensor," *Journal of Sleep Research*, p. e14015, 2023.

[11] J. F Carter, Jj Jorge, Oj Gibson, and Lj Tarassenko, "Sleepvst: Sleep staging from near-infrared video signals using pretrained transformers," *arXiv preprint arXiv:2404.03831*, 2024.

[12] E. D. Chinoy, J. A. Cuellar, K. E. Huwa, J. T. Jameson, C. H. Watson, S. C. Bessman, D. A. Hirsch, A. D. Cooper, S. P. A. Drummond, and R. R. Markwald, "Performance of seven consumer sleep-tracking devices compared with polysomnography," *Sleep*, vol. 44, no. 5, 12 2020.

[13] M. Perslev, S. Darkner, L. Kempfner, M. Nikolic, P. J. Jennum, and C. Igel, "U-sleep: resilient high-frequency sleep staging," *npj Digital Medicine*, vol. 4, no. 1, pp. 72, April 2021.

[14] L. Fiorillo, G. Monachino, J. van der Meer, M. Pesce, J. D. Warncke, M. H. Schmidt, C. L. A. Bassetti, A. Tzovara, P. Favaro, and F. D. Faraci, "U-sleep's resilience to aasm guidelines," *NPJ digital medicine*, vol. 6, no. 1, pp. 33, 2023.

[15] R. Thapa, B. He, M. R. Kjaer, H. Moore, G. Ganjoo, M. Mignot, and J. Zou, "SleepFM: Multi-modal representation learning for sleep across brain activity, ECG and respiratory signals," in *Forty-first International Conference on Machine Learning*, 2024.

[16] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations*, November 2021.

[17] M. M. van Gilst et al., "Protocol of the somnia project: an observational study to create a neurophysiological database for advanced clinical sleep monitoring," *BMJ open*, vol. 9, no. 11, November 2019.

[18] F. B. van Meulen, A. Grassi, L. van den Heuvel, S. Overeem, M. M. van Gilst, J. P. van Dijk, H. Maass, M. J. H. van Gastel, and P. Fonseca, "Contactless camera-based sleep staging: The healthbed study," *Bioengineering*, vol. 10, no. 1, 2023.

[19] American Academy of Sleep Medicine, *International classification of sleep disorders*, American Academy of Sleep Medicine, Darien, IL, USA, 3rd ed, text revision edition, 2023.

[20] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[21] F. Andreotti, H. Phan, and M. De Vos, "Visualising convolutional neural network decisions in automatic sleep scoring," in *CEUR Workshop Proceedings*. CEUR Workshop Proceedings, 2018, pp. 70–81.

[22] M. Dutt, S. Redhu, M. Goodwin, and C. W. Omlin, "Sleepxai: An explainable deep learning approach for multi-class sleep stage identification," *Applied Intelligence*, vol. 53, no. 13, pp. 16830–16843, 2023.

[23] H. Phan, K. B. Mikkelsen, O. Chen, P. Koch, A. Mertins, and M. de Vos, "Sleeptransformer: Automatic sleep staging

with interpretability and uncertainty quantification," *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 8, pp. 2456–2467, 2022.

[24] I. A. M. Huijben, S. Overeem, M. M. van Gilst, and R. J. G. van Sloun, "Attention on sleep stage specific characteristics," in *46th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2024.

[25] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[26] A. Brink-Kjaer, K. M. Gunter, E. Mignot, E. During, P. Jennum, and H. B. D. Sorensen, "End-to-end deep learning of polysomnograms for classification of rem sleep behavior disorder," in *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2022, pp. 2941–2944.

[27] H. van Gorp, I. A. M. Huijben, P. Fonseca, R. J. G. van Sloun, S. Overeem, and M. M. van Gilst, "Certainty about uncertainty in sleep staging: a theoretical framework," *Sleep*, vol. 45, no. 8, June 2022.

[28] R. S. Rosenberg and S. Van Hout, "The American Academy of Sleep Medicine inter-scorer reliability program: sleep stage scoring," *Journal of Clinical Sleep Medicine*, vol. 9, no. 1, pp. 81–87, January 2013.

[29] H. Zhu, C. Fu, F. Shu, H. Yu, C. Chen, and W. Chen, "The effect of coupled electroencephalography signals in electrooculography signals on sleep staging based on deep learning methods," *Bioengineering*, vol. 10, no. 5, pp. 573, 2023.

[30] H. van Gorp, M. M. van Gilst, S. Overeem, S. Dujardin, A. Pijpers, B. van Wetten, P. Fonseca, and R. J. G. van Sloun, "Single-channel eog sleep staging on a heterogeneous cohort of subjects with sleep disorders," *Physiological Measurement*, 2024.

[31] L. Cerina, G. B. Papini, P. Fonseca, S. Overeem, J. P. van Dijk, and R. Vullings, "Extraction of cardiac-related signals from a suprasternal pressure sensor during sleep," *Physiological Measurement*, vol. 44, no. 3, pp. 035002, 2023.

[32] J. Xie, P. Fonseca, J. P. van Dijk, S. Overeem, and X. Long, "Assessment of obstructive sleep apnea severity using audio-based snoring features," *Biomedical Signal Processing and Control*, vol. 86, pp. 104942, 2023.

[33] M. Radha, P. Fonseca, A. Moreau, M. Ross, A. Cerny, P. Anderer, X. Long, and R. M. Aarts, "A deep transfer learning approach for wearable sleep stage classification with photoplethysmography," *npj Digital Medicine*, vol. 4, no. 1, pp. 135, Sep 2021.

[34] J. F. van der Aar, D. A. van den Ende, P. Fonseca, F. B. van Meulen, and M. M. van Gilst, "Deep transfer learning for automated single-lead eeg sleep staging with channel and population mismatches," *Frontiers in Physiology*, vol. 14, pp. 1287342, 2024.

[35] H. van Gorp, M. M. van Gilst, P. Fonseca, S. Overeem, and R. J. G. van Sloun, "Modeling the impact of inter-rater disagreement on sleep statistics using deep generative learning," *IEEE Journal of Biomedical and Health Informatics*, pp. 1–12, August 2023.

[36] Y. Song, P. Dhariwal, M. Chen, and I. Sutskever, "Consistency models," in *Proceedings of the 40th International Conference on Machine Learning*. 23–29 Jul 2023, vol. 202 of *Proceedings of Machine Learning Research*, pp. 32211–32252, PMLR.

[37] Y. Song and P. Dhariwal, "Improved techniques for training consistency models," in *The Twelfth International Conference on Learning Representations*, 2024.

[38] J. Kohler, A. Pumarola, E. Schönfeld, A. Sanakoyeu, R. Sumbaly, P. Vajda, and A. Thabet, "Imagine flash: Accelerating emu diffusion models with backward distillation," *Facebook preprint*, April 2024.

[39] T. Karras, M. Aittala, T. Aila, and S. Laine, "Elucidating the design space of diffusion-based generative models," in *Advances in Neural Information Processing Systems*, December 2022, vol. 35, pp. 26565–26577.

[40] P. Vincent, "A connection between score matching and denoising autoencoders," *Neural Computation*, vol. 23, no. 7, pp. 1661–1674, 2011.

[41] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, Eds. 2020, vol. 33, pp. 6840–6851, Curran Associates, Inc.

[42] B. Efron, "Tweedie's formula and selection bias," *Journal of the American Statistical Association*, vol. 106, no. 496, pp. 1602–1614, 2011.

[43] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," in *Advances in Neural Information Processing Systems*, 2021, vol. 34, pp. 8780–8794.

[44] J. Ho and T. Salimans, "Classifier-free diffusion guidance," in *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

[45] J. Song, A. Vahdat, M. Mardani, and J. Kautz, "Pseudoinverse-guided diffusion models for inverse problems," in *International Conference on Learning Representations*, 2023.

[46] H. Chung, J. Kim, M. T. Mccann, M. L. Klasky, and J. C. Ye, "Diffusion posterior sampling for general noisy inverse problems," *arXiv preprint arXiv:2209.14687*, 2022.

[47] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations*, 2020.

[48] T. S. W. Stevens, F. C. Meral, J. Yu, I. Z. Apostolakis, J. L. Robert, and R. J. G. Van Sloun, "Dehazing ultrasound using diffusion models," *IEEE Transactions on Medical Imaging*, 2024.

[49] A. Caticha, "Entropic inference," in *AIP Conference Proceedings*. American Institute of Physics, 2011, vol. 1305, pp. 20–29.

[50] L. Fiorillo, D. Pedroncelli, V. Agostini, P. Favaro, and F. D. Faraci, "Multi-scored sleep databases: How to exploit the multiple-labels in automated sleep scoring," *Sleep*, vol. 46, no. 5, pp. zsad028, 2023.

[51] P. Anderer, M. Ross, A. Cerny, R. Vasko, E. Shaw, and P. Fonseca, "Overview of the hypnodensity approach to scoring sleep for polysomnography and home sleep testing," *Frontiers in Sleep*, vol. 2, April 2023.

[52] F. Scholkmann, J. Boss, and M. Wolf, "An efficient algorithm for automatic peak detection in noisy periodic and quasi-periodic signals," *Algorithms*, vol. 5, no. 4, pp. 588–603, 2012.

[53] H. van Gorp, M. M. van Gilst, P. Fonseca, S. Overeem, and R. J. G. van Sloun, "Single-channel eog sleep staging on a heterogeneous cohort of subjects with sleep disorders," *Physiological measurement*, pp. 1–14, 2024.

[54] J. B. Stephansen et al., "Neural network analysis of sleep stages enables efficient diagnosis of narcolepsy," *Nature Communications*, vol. 9, no. 1, pp. 5229, December 2018.

[55] Y. Wu and K. He, "Group normalization," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[56] D. Hendrycks and K. Gimpel, "Gaussian error linear units (gelus)," *arXiv preprint arXiv:1606.08415*, 2016.

[57] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 648–656.

[58] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

Author contributions

H.G. was involved with conceptualization, data processing, model development, and writing of the manuscript. M.G., P.F., S.O., R.S. were involved with conceptualization, reviewing, and supervision. M.G., J.D., F.M., S.O. were involved with data collection.

Competing interests

At the time of writing, H.G. and P.F. were employed and/or affiliated with Royal Philips, a commercial company and manufacturer of consumer and medical electronic devices, commercializing products in the area of sleep diagnostics and sleep therapy. Philips had no role in the study design, decision to publish, or preparation of the manuscript.

SUPPLEMENTAL MATERIAL - ADDITIONAL QUANTITATIVE RESULTS

Table 4: Hold-out test set results for all models. '#' refers to the number of recordings in the hold-out test set where the signal was available. We here show the average accuracy over the recordings, in terms of 5-,4-,3-, and 2-class sleep staging.

Signal(s)	#	Accuracy [%] W/N1/N2/N3/REM	W/N1-N2/N3/REM	W/NREM/REM	Wake/Sleep
All PSG electrodes	497	85.9	89.8	93.4	96.4
Left EEG electrodes	500	85.5	89.5	93.3	96.4
F3-M2 (EEG)	500	85.6	89.5	93.1	96.2
C3-M2 (EEG)	500	85.3	89.4	93.2	96.2
O1-M2 (EEG)	500	83.4	87.9	92.6	96.1
Recommended PSG electrodes	497	86.3	90.2	93.6	96.5
Right EEG electrodes	500	85.8	89.8	93.3	96.5
F4-M1 (EEG)	500	85.5	89.4	93.0	96.1
C4-M1 (EEG)	500	85.5	89.6	93.2	96.3
O2-M1 (EEG)	500	83.8	88.2	92.7	96.2
E2-M2 (EOG)	497	85.0	89.2	93.2	96.2
E1-M2 (EOG)	500	84.3	88.8	92.9	95.9
Chin1-Chin2 (EMG)	500	74.9	80.9	88.1	92.7
Chin2-Chin3 (EMG)	500	74.9	80.9	88.0	92.6
Chin1-Chin3 (EMG)	500	74.5	80.6	87.8	92.5
HSAT expanded	434	79.0	84.3	90.8	94.4
HSAT reduced	434	78.3	83.6	90.2	94.1
Nasal cannula	434	76.5	81.9	89.0	93.4
Finger PPG	500	75.1	80.8	88.0	92.4
Thoracic belt	500	76.3	82.7	89.7	93.6
HSAT reduced	434	77.6	82.9	89.6	93.8
Nasal cannula	434	76.5	81.9	89.0	93.4
IHR from finger PPG	500	71.8	77.6	84.9	90.0
HSAT expanded	434	78.5	84.0	90.6	94.1
HSAT reduced	434	77.7	83.1	89.9	93.5
thermistor	434	72.9	79.2	87.0	91.6
ECG	500	76.9	82.2	89.1	93.2
Thoracic belt	500	76.3	82.7	89.7	93.6
HSAT expanded	66	78.2	83.5	90.5	93.7
HSAT reduced	66	76.4	81.2	88.5	92.1
PAP flow	66	69.5	74.4	83.1	87.8
Finger PPG	500	75.1	80.8	88.0	92.4
Thoracic belt	500	76.3	82.7	89.7	93.6
HSAT reduced	65	74.9	80.1	87.0	92.0
IBR from PAP flow	65	69.2	74.8	83.0	89.3
IHR from finger PPG	500	71.8	77.6	84.9	90.0
Left Leg and SCM	33	71.0	77.0	85.1	90.7
Left Leg (EMG)	500	66.9	72.2	81.2	88.5
Left SCM (EMG)	33	66.2	72.4	80.5	89.5
Right Leg and SCM	33	70.3	76.3	84.4	90.2
Right Leg (EMG)	500	66.7	72.0	80.8	88.2
Right SCM (EMG)	33	67.0	73.1	81.7	89.7
Left Leg and FDS	60	67.4	72.8	81.8	88.7
Left Leg (EMG)	500	66.9	72.2	81.2	88.5
Left FDS (EMG)	60	63.7	69.5	78.8	88.2
Right Leg and FDS	60	68.0	73.1	82.0	88.6
Right Leg (EMG)	500	66.7	72.0	80.8	88.2
Right FDS (EMG)	60	63.2	69.1	78.2	87.9
Abdominal belt	500	76.4	83.0	89.9	93.6
Snore microphone	500	72.1	78.0	85.9	91.9
IHR from ECG	500	70.1	75.8	83.8	89.1
IBR from RIP thorax	500	70.0	76.1	83.9	89.7
IBR from RIP abdomen	500	69.9	76.1	84.0	89.7
SpO2	500	68.7	74.7	82.8	89.1
IBR from nasal cannula	434	66.9	73.3	81.5	87.5
IBR from Thermistor	434	65.4	72.0	80.8	87.4
Suprasternal notch	72	63.1	69.5	78.4	86.0
IBR from esophageal pressure	24	59.0	67.9	76.7	83.2
IBR from Suprasternal notch	72	58.5	65.3	74.5	82.2
Esophageal pressure	24	55.5	62.1	72.5	80.7

Table 5: Hold-out test set results for all models. '#' refers to the number of recordings in the hold-out test set where the signal was available. We here show the average kappa over the recordings, in terms of 5-,4-,3-, and 2-class sleep staging.

Signal(s)	#	Cohen's Kappa			
		W/N1/N2/N3/REM	W/N1-N2/N3/REM	W/NREM/REM	Wake/Sleep
All PSG electrodes	497	0.793	0.826	0.853	0.858
Left EEG electrodes	500	0.789	0.822	0.850	0.863
F3-M2 (EEG)	500	0.791	0.823	0.846	0.855
C3-M2 (EEG)	500	0.787	0.821	0.848	0.857
O1-M2 (EEG)	500	0.760	0.795	0.835	0.850
Recommended PSG electrodes	497	0.799	0.832	0.856	0.864
Right EEG electrodes	500	0.794	0.827	0.851	0.863
F4-M1 (EEG)	500	0.790	0.822	0.844	0.850
C4-M1 (EEG)	500	0.791	0.824	0.848	0.857
O2-M1 (EEG)	500	0.764	0.800	0.837	0.851
E2-M2 (EOG)	497	0.784	0.820	0.850	0.858
E1-M2 (EOG)	500	0.776	0.815	0.845	0.852
Chin1-Chin2 (EMG)	500	0.630	0.677	0.737	0.716
Chin2-Chin3 (EMG)	500	0.631	0.678	0.735	0.716
Chin1-Chin3 (EMG)	500	0.624	0.672	0.731	0.709
HSAT expanded	434	0.697	0.740	0.801	0.793
HSAT reduced	434	0.686	0.731	0.791	0.783
Nasal cannula	434	0.661	0.705	0.767	0.762
Finger PPG	500	0.640	0.685	0.746	0.735
Thoracic belt	500	0.657	0.708	0.775	0.765
HSAT reduced	434	0.676	0.719	0.777	0.774
Nasal cannula	434	0.661	0.705	0.767	0.762
IHR from finger PPG	500	0.597	0.637	0.693	0.688
HSAT expanded	434	0.687	0.733	0.797	0.787
HSAT reduced	434	0.674	0.720	0.785	0.774
thermistor	434	0.603	0.650	0.717	0.702
ECG	500	0.669	0.711	0.773	0.772
Thoracic belt	500	0.657	0.708	0.775	0.765
HSAT expanded	66	0.678	0.722	0.797	0.778
HSAT reduced	66	0.652	0.690	0.759	0.735
PAP flow	66	0.562	0.594	0.661	0.634
Finger PPG	500	0.640	0.685	0.746	0.735
Thoracic belt	500	0.657	0.708	0.775	0.765
HSAT reduced	65	0.626	0.666	0.719	0.699
IBR from PAP flow	65	0.538	0.580	0.634	0.579
IHR from finger PPG	500	0.597	0.637	0.693	0.688
Left Leg and SCM	33	0.575	0.624	0.699	0.716
Left Leg (EMG)	500	0.526	0.558	0.621	0.632
Left SCM (EMG)	33	0.502	0.546	0.594	0.681
Right Leg and SCM	33	0.565	0.613	0.689	0.700
Right Leg (EMG)	500	0.523	0.556	0.616	0.629
Right SCM (EMG)	33	0.517	0.561	0.627	0.679
Left Leg and FDS	60	0.532	0.566	0.633	0.647
Left Leg (EMG)	500	0.526	0.558	0.621	0.632
Left FDS (EMG)	60	0.482	0.510	0.555	0.615
Right Leg and FDS	60	0.540	0.569	0.639	0.647
Right Leg (EMG)	500	0.523	0.556	0.616	0.629
Right FDS (EMG)	60	0.476	0.503	0.544	0.609
Abdominal belt	500	0.661	0.715	0.779	0.770
Snore microphone	500	0.597	0.639	0.692	0.714
IHR from ECG	500	0.571	0.609	0.670	0.659
IBR from RIP thorax	500	0.564	0.611	0.662	0.626
IBR from RIP abdomen	500	0.563	0.612	0.664	0.625
SpO2	500	0.542	0.589	0.642	0.624
IBR from nasal cannula	434	0.521	0.568	0.616	0.563
IBR from Thermistor	434	0.497	0.545	0.597	0.536
Suprasternal notch	72	0.464	0.499	0.553	0.573
IBR from esophageal pressure	24	0.419	0.473	0.523	0.474
IBR from Suprasternal notch	72	0.394	0.433	0.473	0.431
Esophageal pressure	24	0.373	0.410	0.466	0.483

Table 6: Hold-out test set results for all models. '#' refers to the number of recordings in the hold-out test set where the signal was available. We show the average of each metric over the recordings. We here show the per sleep stage F1 scores.

Signal(s)	#	Wake	N1	N2	N3	REM
All PSG electrodes	497	0.885	0.588	0.881	0.834	0.882
Left EEG electrodes	500	0.888	0.599	0.876	0.821	0.871
F3-M2 (EEG)	500	0.882	0.593	0.878	0.833	0.869
C3-M2 (EEG)	500	0.883	0.596	0.874	0.826	0.872
O1-M2 (EEG)	500	0.878	0.574	0.857	0.767	0.857
Recommended PSG electrodes	497	0.890	0.598	0.883	0.845	0.881
Right EEG electrodes	500	0.886	0.599	0.879	0.833	0.871
F4-M1 (EEG)	500	0.876	0.591	0.879	0.834	0.868
C4-M1 (EEG)	500	0.882	0.598	0.877	0.828	0.872
O2-M1 (EEG)	500	0.878	0.572	0.859	0.787	0.858
E2-M2 (EOG)	497	0.887	0.577	0.865	0.828	0.876
E1-M2 (EOG)	500	0.881	0.582	0.855	0.821	0.874
Chin1-Chin2 (EMG)	500	0.766	0.352	0.778	0.677	0.813
Chin2-Chin3 (EMG)	500	0.767	0.353	0.778	0.683	0.810
Chin1-Chin3 (EMG)	500	0.760	0.342	0.775	0.677	0.807
HSAT expanded	434	0.833	0.434	0.811	0.724	0.845
HSAT reduced	434	0.824	0.393	0.804	0.718	0.832
Nasal cannula	434	0.806	0.375	0.787	0.702	0.811
Finger PPG	500	0.787	0.366	0.775	0.689	0.798
Thoracic belt	500	0.813	0.433	0.783	0.689	0.828
HSAT reduced	434	0.818	0.398	0.799	0.712	0.819
Nasal cannula	434	0.806	0.375	0.787	0.702	0.811
IHR from finger PPG	500	0.755	0.351	0.744	0.665	0.757
HSAT expanded	434	0.832	0.429	0.807	0.700	0.844
HSAT reduced	434	0.821	0.386	0.800	0.694	0.833
thermistor	434	0.759	0.331	0.762	0.640	0.774
ECG	500	0.821	0.399	0.786	0.699	0.829
Thoracic belt	500	0.813	0.433	0.783	0.689	0.828
HSAT expanded	66	0.820	0.373	0.814	0.669	0.853
HSAT reduced	66	0.788	0.320	0.798	0.670	0.834
PAP flow	66	0.709	0.239	0.733	0.591	0.784
Finger PPG	500	0.787	0.366	0.775	0.689	0.798
Thoracic belt	500	0.813	0.433	0.783	0.689	0.828
HSAT reduced	65	0.748	0.267	0.793	0.662	0.798
IBR from PAP flow	65	0.641	0.155	0.747	0.626	0.742
IHR from finger PPG	500	0.755	0.351	0.744	0.665	0.757
Left Leg and SCM	33	0.784	0.239	0.721	0.595	0.727
Left Leg (EMG)	500	0.709	0.226	0.698	0.598	0.689
Left SCM (EMG)	33	0.757	0.237	0.668	0.574	0.614
Right Leg and SCM	33	0.774	0.234	0.713	0.585	0.723
Right Leg (EMG)	500	0.705	0.220	0.696	0.597	0.680
Right SCM (EMG)	33	0.754	0.232	0.673	0.543	0.644
Left Leg and FDS	60	0.727	0.235	0.698	0.614	0.712
Left Leg (EMG)	500	0.709	0.226	0.698	0.598	0.689
Left FDS (EMG)	60	0.696	0.257	0.672	0.620	0.578
Right Leg and FDS	60	0.727	0.242	0.706	0.604	0.714
Right Leg (EMG)	500	0.705	0.220	0.696	0.597	0.680
Right FDS (EMG)	60	0.693	0.248	0.663	0.617	0.580
Abdominal belt	500	0.816	0.442	0.781	0.694	0.831
Snore microphone	500	0.771	0.362	0.749	0.668	0.729
IHR from ECG	500	0.732	0.346	0.729	0.637	0.740
IBR from RIP thorax	500	0.691	0.199	0.737	0.666	0.736
IBR from RIP abdomen	500	0.688	0.199	0.737	0.664	0.738
SpO2	500	0.697	0.181	0.722	0.634	0.712
IBR from nasal cannula	434	0.639	0.179	0.712	0.651	0.709
IBR from Thermistor	434	0.610	0.157	0.701	0.610	0.698
Suprasternal notch	72	0.671	0.232	0.662	0.561	0.596
IBR from esophageal pressure	24	0.568	0.129	0.657	0.579	0.607
IBR from Suprasternal notch	72	0.541	0.112	0.639	0.566	0.587
Esophageal pressure	24	0.614	0.117	0.569	0.522	0.529

SUPPLEMENTAL MATERIAL - ADDITIONAL QUALITATIVE RESULTS

We here show a qualitative example for each of the signal(s) as shown in tables 6 and 4. We show the most typical example for each, defined as the recording where it achieved median performance in terms of accuracy.

Figure 12: Some additional qualitative examples.

Figure 13: Some additional qualitative examples.

Figure 14: Some additional qualitative examples.

Figure 15: Some additional qualitative examples.

Figure 16: Some additional qualitative examples.

Figure 17: Some additional qualitative examples.

Figure 18: Some additional qualitative examples.

SUPPLEMENTAL MATERIAL - SAMPLES FROM THE PRIOR

Figure 19: We here show samples taken from the global prior score network (without the use of any measurement data).

SUPPLEMENTAL MATERIAL - INFORMATION GAIN VS PERFORMANCE

We here show the same experiment as shown in Fig. 6 from the manuscript, but now for Cohen's kappa instead of accuracy.

Figure 20: Quantitative results for information gain. **(A)** The average information gain per sensor over all test recordings shows a clear linear correlation with respect to Cohen's kappa. **(B)** Reducing the usefulness of the ECG signal by removing segments or adding noise reduces down-stream accuracy and information gain. The linear relationship as fitted on the data from (A) still provides a good fit here.

Table 7: Specific diagnoses and how they were clustered. Subjects could have multiple sleep disorders within the same cluster. For example, all subjects who had a pediatric obstructive sleep apnea diagnosis also had an adult obstructive sleep apnea diagnsosis. *Note that this table spans two pages.*

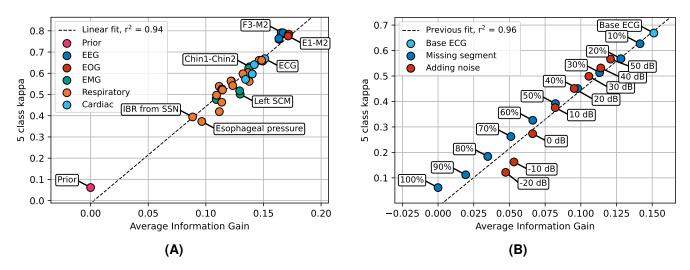| Diagnosis	#	Cluster	#
Chronic insomnia disorder	217		
Psychophysiological insomnia	206		
Idiopathic insomnia	11		
Paradoxical insomnia	23		
Inadequate sleep hygiene	55	Insomnia disorders	613
Behavioral insomnia of childhood	0		
Insomnia due to (another) mental disorder	69		
Insomnia due to (a) medical condition	86		
Other insomnia disorder	31		
Obstructive sleep apnea, adult	1037	Obstructive Sleep Apnea	1037
Obstructive sleep apnea, pediatric	4		
Central sleep apnea with Cheyne-Stokes breathing	19		
Central sleep apnea due to a medical disorder without Cheyne-Stokes breathing	9	Central Sleep Apnea	42
Central sleep apnea due to a medication or substance	3		
Primary central sleep apnea	11		
Treatment emergent central sleep apnea	6	Treatment emergent central sleep apnea	6
Obesity hypoventilation syndrome	1		
Idiopathic central alveolar hypoventilation	1		
Sleep related hypoventilation due to a medication or substance	1	Hypoventilation	8
Sleep related hypoventilation due to a medical disorder	0		
Sleep related hypoxemia disorder	5		
Narcolepsy type 1	25		
Narcolepsy type 1 due to a medical condition	1		
Narcolepsy type 2	5	Narcolepsy	31
Narcolepsy type 2 due to a medical condition	1		
Idiopathic hypersomnia	8		
Idiopathic hypersomnia with normal sleep time	14		
Idiopathic hypersomnia with long sleep time	8		
Kleine-Levin syndrome	1		
Hypersomnia due to a medical disorder	11	Other Hypersomnolence Disorders	54
Hypersomnia secondary to Parkinson disease	4		
Residual hypersomnia in OSA patients with adequately treated OSA	2		
Hypersomnia associated with a psychiatric disorder	7		
Hypersomnia associated with mood disorder	1		
Hypersomnia associated with a conversion disorder or somatic symptom disorder	1		
Insufficient sleep syndrome	66	Insufficient sleep syndrome	66
Delayed sleep-wake phase disorder	33		
Advanced sleep-wake phase disorder	2		
Irregular sleep-wake rhythm disorder	1	Circadian rhythm disorders	46
Shift work disorder	9		
Circadian sleep-wake disorder NOS	1		

Table 7: Specific diagnoses and how they were clustered. Subjects could have multiple sleep disorders within the same cluster. For example, all subjects who had a pediatric obstructive sleep apnea diagnosis also had an adult obstructive sleep apnea diagnososis. *Note that this table spans two pages.*

Diagnosis	#	Cluster	#
Confusional arousals	74		
Sleepwalking	48		
Sleep terrors	24	NREM parasomnias	115
Sleep related abnormal sexual behaviors	2		
Sleep related eating disorder	1		
REM sleep behavior disorder	122	RBD	122
Recurrent isolated sleep paralysis	11		
Nightmare disorder	39	REM parasomnias other than RBD	55
Sleep related hallucinations	12		
Parasomnia overlap disorder	7		
Exploding head syndrome	1		
Sleep enuresis	2	Other parasomnias	45
Parasomnia due to a medical disorder	4		
Parasomnia, unspecified	31		
Restless legs syndrome	185	RLS/PLMD	268
Periodic limb movement disorder	114		
Sleep related leg cramps	2		
Sleep related bruxism	27		
Sleep related rhythmic movement disorder	5	Other movement disorders	58
Propriospinal myoclonus at sleep onset	2		
Sleep related movement disorder, unspecified	18		
Sleep starts (hypnic jerks)	5		
Other sleep disorder	5		
Sleep related epilepsy	2		
Sleep related headache	1	Other	16
Sleep related laryngospasm	4		
Sleep related gastroesophageal reflux	3		
Sleep disorder due to sedative, hypnotic or anxiolytic	1		
No primary sleep diagnosis	45		
Short sleeper	2		
Snoring	31	No primary sleep diagnosis and/or normal variants	99
Catathrenia	7		
Long sleeper	14		
Healthy	96	Healthy	96

DETAILS REGARDING THE NETWORK ARCHITECTURE

We leveraged the DDPM++ model as implemented by Karras *et al.* [39], and modified to work on 1D timeseries in our previous work on EOG-driven sleep staging [53]. See Fig. 21 for an overview. The neural network architecture used in this work differs from [53] in two aspects. Firstly, there is an additional input between the epoch encoder and U-Net encoder in order to add the $\boldsymbol{y}_{m-1}$ of the previous output (in order to solve the ODE). Secondly, there is an additional embedding in ResNets to add the current diffusion timestep, a common practice in score-based diffusion models [39]. We will now discuss each neural network component.

Epoch encoder

Because the signals and the hypnograms had different sampling frequencies (128 Hz vs. 1/30 Hz), we first needed to downsample the input signal before we could use the U-Net structure of our model. To that end, we employed a context encoder, which downsampled the signals from $\mathbb{R}^{1792\cdot30\cdot128\times1}$ to $\mathbb{R}^{1792\times16}$, i.e. a context encoding of length number of epochs with 16 channels.

The context encoder worked as follows. First, a convolution of kernel size 1 expanded the number of channels from 1 to 16. Then, a series of two ResNets was employed to extract meaningful features from the input signal (see the ResNet section for further details). This pattern was repeated 5 times with 4 downsampling operations between the 5 blocks. Each downsampling operation used a kernel of [1,1,1,1] and a stride of 4, to effectively downsample the input by a factor of 4. At the end of the epoch encoder, another convolution of kernel and stride 15 was used, thus compressing the signals to a feature map of size $\mathbb{R}^{1792\times16}$ which was used as input to the U-Net encoder.

Stacking

After the epoch encoder, the feature map is concatenated channel wise with the previous estimate of the diffusion step. Following [39], we apply input scaling to the previous estimate of $\boldsymbol{y}_m$ as:

$$\tilde{\boldsymbol{y}}_m = \frac{1}{\sqrt{\sigma_{data}^2 + \sigma_t^2}} \cdot \boldsymbol{y}_m, \tag{23}$$

Where σ_{data} was estimated from data as $\sigma_{data} = 0.3160$ and σ_t is the current variance of the diffusion ODE. The stacking then results in an input of $\mathbb{R}^{1792\times(16+5)} = \mathbb{R}^{1792\times21}$ for the U-Net encoder.

Note on prior networks

When we are using the network as a prior network, no input conditioning data is used. Thus, we skip the epoch encoder and the stacking operation, and we only input the previous diffused hypnogram $\tilde{\boldsymbol{y}}_m$ into the rest of the network.

U-Net encoder

The U-Net encoder first employed a convolution of kernel size 1 to increase the channel size from 21 to 32. Then, a Transformer layer together with two ResNet blocks was employed (see the Transformer layer section for further details). After each ResNet block, a skip connection was added to the U-Net decoder at the same resolution. This pattern of a transformer with two ResNets was repeated 4 times with 3 downsampling operations in between. Again, a kernel of [1,1,1,1] and a stride of 4 was used in the downsampling operations. The number of channels was left the same throughout the network, at a fixed 32 channels. Note that in the original DDPM++ implementation [16], an attention layer was added after each ResNet in the encoder. However, to bring down the computational complexity of our method and to make the encoder symmetric with the decoder, we employed only a single transformer layer at the start of each resolution level in the U-Net encoder.

Bottleneck

In the bottleneck, the feature map was of its smallest size, namely $\mathbb{R}^{28\times32}$. Here, one transformer layer sandwiched between two ResNet blocks was used to learn the highest-level features of the hypnogram.

U-Net decoder

The decoder followed a mirrored structure to the encoder. The skip connections from the corresponding resolution levels were concatenated to the inputs of each ResNet block. These connections allowed the feature maps to skip the downward path of the 'U' and enabled the model to learn both high-and-low level features of the hypnogram. The upsampling operation of the decoder was implemented using a transposed convolution with the same filter of [1,1,1,1].

As a final step toward creating the $\boldsymbol{y}_m^{denoised}$, the U-Net decoder employed a convolution of kernel size 1 to map the input to 5 channels, where each channel corresponded to one of the five sleeps stages. A softmax activation function was then used to map each channel to a class probability. This creates a 'hypnodensity', a soft version of the hypnogram where each epoch is partially associated with each sleep stage according to some probability [54].

ResNet

The ResNet, or Residual Network, was repeated throughout the architecture. It consists of two group normalization layers and two convolutions in an alternating pattern. Group normalization, as described by [55], applies a learned normalization across groups of channels, enabling faster training. In our case, each group consisted of 4 channels. The 1D convolutions of the ResNet each used a kernel of size 7 and zero-padding set to 'same'. Each convolution was followed by SiLU (Sigmoid Linear Unit) activation [56]. Additionally, a spatial dropout layer was added before the second convolution, which drops out entire channels during training with a probability of 10%. Spatial dropout is a better regularizer for convolutional neural networks, since neighbouring samples are often highly correlated [57]. Finally, a residual connection was added to help combat vanishing gradient problems. To limit the magnitude of the signals, scaling with a factor of $skip\ scale = \sqrt{0.5}$ was applied.

In the case that the ResNet was part of the epoch encoder, the additive timestep embedding was equal to zero, and we effectively do not add it. Otherwise. an additive timestep embedding is generated from the current noise level of the diffusion process to tell it about what kind of noise level to expect in $\boldsymbol{y}_{m-1}$, which has been found to be helpful in the score-based diffusion literature. This additive timestep embedding is explained in the sequel.

Timestep embedding

Following [39], the current noise level of the diffusion process, σ_t, is given as an additional input to the network in a scaled and embedded form. To that end, it is first scaled as follows:

$$\tilde{\sigma}_t = 0.25 \log(\sigma_t). \tag{24}$$

After this scaling, a sine-cosine embedding scheme was used that embedded the noise level as follows:

$$\boldsymbol{c} = [0, 1, \ldots, C/2 - 1]^T, \tag{25}$$

$$\boldsymbol{f} = 1000\hat{}(-\boldsymbol{c}/(C/2 - 1)), \tag{26}$$

$$\boldsymbol{z} = [\cos(\tilde{\sigma}_t \cdot \boldsymbol{f}),\ \sin(\tilde{\sigma}_t \cdot \boldsymbol{f})]^T, \tag{27}$$

where C is the number of channels in the ResNet, which is equal to 32 throughout the U-Net. Subsequently, a multilayer perceptron (MLP) was applied of 2 layers, with 8 hidden nodes each and SiLU activation. Then, for each ResNet separately, a local linear layer was applied to increase the noise level embedding to 32 again. As a final step it is broadcasted into the input length of the feature map at the ResNet and added to it element-wise.

Transformer

The original transformer architecture is a sequence-to-sequence model composed of both an encoder and a decoder [58]. Where each element consists of a scaled dot-product attention layer and an element-wise feed-forward network. Additionally, positional encoding is added at the start of the encoding and decoding stacks. We adapt the transformer architecture to be suited for our network.
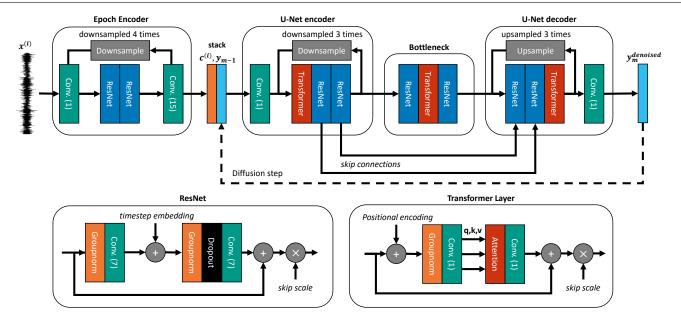
Figure 21: Overview of a the neural network used for each denoiser $D_{\theta^{(i)}}\left(\boldsymbol{y}_m, \boldsymbol{x}^{(i)}, \sigma(t_m)\right)$. The signal, $\boldsymbol{x}^{(i)}$, is used as the inital input to the network and encoded into a context vector. This is then stacked with the sample at the current timestep $\boldsymbol{y}_m$ and fed though a U-Net structure. At the end, a hypnodensity is given as output through the use of a softmax activation function. The current noise variance, $\sigma(t_m)$, is additionally embedded into a timestep embedding, which is added inside the ResNet layers of the U-Net encoder and U-Net decoder. To avoid needing to run the Epoch Encoder M times, the timestep embedding is not added to its ResNet layers.

Firstly, we did not use the decoder, since it is used to generate new sequence in an auto-regressive manner. Secondly, since we embedded the layers within a larger convolutional neural network, there was no need for separate element-wise feed-forward networks. lastly, because the attention layers operated at different time scales, we added positional encoding to each of them.

The positional encoding was also implemented using sine-cosine embedding. The encoding scheme used is similar to the timestep embedding, with some differences. Namely, we here create a full matrix embedding instead of only a vector embedding, no MLP is applied, and the sine and cosine terms are interleaved.

In the transformer layer positional encoding scheme, a positional encoding matrix is added element-wise to the input sequence of the transformer. To that end, the input sequence $\mathbf{S}$ and positional encoding matrix $\mathbf{P}$ should be of the same size: $\mathbf{S}, \mathbf{P} \in \mathbb{R}^{L \times C}$, where L is the length of the input sequence and C is the number of channels (32 in our case). The positional encoding matrix for the transformer layers is given by:

$$\mathbf{P}_{(l,2c)} = \sin\left(l \cdot 1000^{-2c/C}\right)$$
$$\mathbf{P}_{(l,2c+1)} = \cos\left(l \cdot 1000^{-2c/C}\right), \tag{28}$$

with $l \in [0, 1, \ldots, L-1]$ and $c \in [0, 1, \ldots, C/2-1]$. This type of encoding enables the transformer to exploit information about both the absolute and relative positions of samples along the night.

Each of the transformer layers used scaled dot-product self-attention. While the attention mechanism can be implemented using multiple attention-heads for added complexity, we here only made use of a single head. In scaled dot-product self-attention, three linear projections are applied to transform the sequence to a query, key, and value matrix:

$$\mathbf{Q} = \mathbf{S}\mathbf{W}_Q, \ \mathbf{K} = \mathbf{S}\mathbf{W}_K, \ \mathbf{V} = \mathbf{S}\mathbf{W}_V, \tag{29}$$

where $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{C \times C}$ are learned linear projection weights and $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{L \times C}$ are the query, key, and value matrices, respectively. These linear projections can be implemented efficiently by a single convolutional layer of kernel size 1 and output channel size of $3C$, as its output can be split along the channel dimension into the three separate components.

Following a database analogy, the queries are going to look for matching keys and propagate the associated values to the output,

where each individual query, key, and value are found along the rows of their respective matrices. This process is defined by the scaled dot-product self-attention mechanism:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{C}}\right)\mathbf{V}, \tag{30}$$

where $\mathbf{K}^T$ denotes the transpose of the key matrix. Moreover, $\mathbf{Q}\mathbf{K}^T \in \mathbb{R}^{L \times L}$ denotes the attention map. To ensure that the magnitudes in the attention map do not grow too large, it is scaled down by a factor of $1/\sqrt{C}$. Additionally, a softmax activation is applied along the rows of the attention map in order to ensure that the attention sums to 1.

After the scaled dot-product attention layer, another linear projection using a 1D convolution was applied. Similar to the ResNet, a residual connection was applied with a scaling of $skip\ scale = \sqrt{0.5}$.