Modeling the Impact of Inter-Rater Disagreement on Sleep Statistics Using Deep Generative Learning

Hans van Gorp, *Graduate Student Member, IEEE*, Merel M. van Gilst, Pedro Fonseca, Sebastiaan Overeem, and Ruud J.G. van Sloun, *Member, IEEE*

***Abstract*—Sleep staging is the process by which an overnight polysomnographic measurement is segmented into epochs of 30 seconds, each of which is annotated as belonging to one of five discrete sleep stages. The resulting scoring is graphically depicted as a hypnogram, and several overnight sleep statistics are derived, such as total sleep time and sleep onset latency. Gold standard sleep staging as performed by human technicians is time-consuming, costly, and comes with imperfect inter-scorer agreement, which also results in inter-scorer disagreement about the overnight statistics. Deep learning algorithms have shown promise in automating sleep scoring, but struggle to model inter-scorer disagreement in sleep statistics. To that end, we introduce a novel technique using conditional generative models based on Normalizing Flows that permits the modeling of the inter-rater disagreement of overnight sleep statistics, termed U-Flow. We compare U-Flow to other automatic scoring methods on a hold-out test set of 70 subjects, each scored by six independent scorers. The proposed method achieves similar sleep staging performance in terms of accuracy and Cohen's kappa on the majority-voted hypnograms. At the same time, U-Flow outperforms the other methods in terms of modeling the inter-rater disagreement of overnight sleep statistics. The consequences of inter-rater disagreement about overnight sleep statistics may be great, and the disagreement potentially carries diagnostic and scientifically relevant information about sleep structure. U-Flow is able to model this disagreement efficiently and can support further investigations into the impact inter-rater disagreement has on sleep medicine and basic sleep research.**

***Index Terms*—Automatic sleep staging, deep learning, generative ai, inter-rater disagreement, uncertainty.**

I. Introduction

THE diagnosis of many sleep disorders is generally supported by an overnight polysomnography (PSG). During such a PSG measurement, several physiological signals are recorded including, among others, electroencephalography (EEG), electrooculography (EOG), electromyography (EMG), and electrocardiography (ECG). These data are then visually inspected by a technician, and, following the American Academy of Sleep Medicine (AASM) scoring rules [1], each 30-second segment, known as an epoch, is annotated as one of five discrete sleep stages: Wake (W), Rapid Eye Movement (REM), or non-REM (NREM) stage 1-3. The resulting sequence of sleep stages is visually represented in a hypnogram. This hypnogram is presented to the physician in a sleep report together with overnight sleep statistics derived from the scored epochs. According to the AASM, these statistics include, among others, total sleep time, time spent in each sleep stage, sleep onset latency, and stage REM latency. Depending on the preferences of e.g. the manufacturer of a sleep diagnostic system they might also include the number of awakenings in both REM and NREM sleep. These sleep statistics are used in support of diagnosis and in basic sleep research.

Sleep staging, however, is not perfect and displays inter-rater disagreement for human technicians. In 2013, Rosenberg and Van Hout showed that individual scorers only tend to agree in 82.6% of epochs to a group consensus [2]. When scoring specific sleep stages, such as N3, the agreement of an individual scorer to a group consensus further dropped to 67.4%. More recently, it has also been shown that agreement between individual scorers

Manuscript received 3 April 2023; revised 4 July 2023; accepted 2 August 2023. Date of publication 10 August 2023; date of current version 7 November 2023. This work was supported by the IMPULSE framework of the Eindhoven MedTech Innovation Center (e/MTIC, incorporating Eindhoven University of Technology, Philips Research, and Sleep Medicine Center, Kempenhaeghe Foundation), including a PPS supplement from the Dutch Ministry of Economic Affairs and Climate Policy. *(Corresponding author: Hans van Gorp.)*

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Stanford Institutional Review Board for the SSC study, the ethics board of the Central Hospital of the University of Coimbra for the ISRUC study, the Committees of Protection of Persons (CPP) for the DOD healthy dataset under clinical trial number NCT03657329, and by the Institutional Review Board (IRB) at the University of Pennsylvania for the IS-RC study.

Hans van Gorp, Pedro Fonseca, and Ruud J.G. van Sloun are with the Department of Electrical Engineering, Eindhoven University of Technology, 5612AZ Eindhoven, The Netherlands, and also with the Philips Research, 5656AE Eindhoven, The Netherlands (e-mail: h.v.gorp@tue.nl; pedro.fonseca@philips.com; r.j.g.v.sloun@tue.nl).

Merel M. van Gilst and Sebastiaan Overeem are with the Department of Electrical Engineering, Eindhoven University of Technology, 5612AZ Eindhoven, The Netherlands, and also with the Sleep Medicine Centre Kempenhaeghe, 5591VE Heeze, The Netherlands (e-mail: m.m.v.gilst@tue.nl; s.overeem@tue.nl).

Output hypnograms for all explored methods on the hold-out test together with scripts to reproduce the evaluation metrics reported here are available at: https://github.com/HansvanGorp/U-Flow

This article has supplementary downloadable material available at https://doi.org/10.1109/JBHI.2023.3304010, provided by the authors.

Digital Object Identifier 10.1109/JBHI.2023.3304010

is approximately inversely proportional to the square root of the number of scorers [3].

As a result of this inter-rater disagreement, there will also be variability in the overnight sleep statistics. However, while the inter-rater disagreement of human technicians itself has widely been studied [2], [4], [5], its effect on sleep statistics has not. This is likely because multi-scorer hypnograms are not part of standard clinical practice, as these are both impractical and expensive. Nonetheless, the consequences of inter-rater disagreement on sleep diagnosis and basic sleep research may be great and could contain medically relevant information.

The recent explosion in computational power and deep learning (DL) [6] could provide the solution to the problem of costly multi-scoring of hypnograms. DL has already enabled many advancements in the field of automatic sleep stage scoring [7], [8], [9], [10], [11], [12], [13]. A recent review by Phan and Mikkelsen gives a thorough overview of recent methods [14]. The hypnodensity concept is especially interesting from the perspective of modeling the inter-rater disagreement [7]. In the hypnodensity concept, rather than having the automatic scoring algorithm output a hypnogram with a single sleep stage per epoch in the recording, it outputs a graph displaying a categorical distribution over all sleep stages for each epoch. It has been shown that such a hypnodensity graph matches well with the label distribution of a human panel [3], [7], [15]. In other words, the DL model which outputs a hypnodensity graph correctly learns to model the inter-rater disagreement on an epoch-based level.

However, it remains unclear how to extrapolate the hypnodensity approach to model the inter-rater disagreement of overnight sleep statistics. This is because the hypnodensity approach, which leverages the cross entropy loss, factorizes the hypnogram into a series of categorical distributions, which removes contextual information about dependencies between epochs. For example, a human scorer might score a series of epochs as all belonging to N3, while another scorer might score them all as belonging to N2. The assumed factorization is unable to model this all-or-nothing dependency between these epochs and will thus be unable to correctly estimate the inter-rater disagreement of the sleep statistics.

We hypothesize that to model the inter-rater disagreement of overnight sleep statistics correctly, a DL method needs to be able to provide multiple, valid, hypnograms given a single PSG, similar to a panel of human scorers. We leverage the fact that such a one-to-many relationship can be estimated efficiently using conditional generative neural networks. Generative neural networks can learn a mapping from an easy to sample distribution, e.g. Gaussian, to a more complex data distribution, where this mapping can also depend on a conditioning. For example, a conditional generative network can create a set of images that all fit a textual prompt. In our case, we leverage the conditional generative network to estimate the probability distribution of hypnograms given a measured PSG (i.e. $p(hypnogram|PSG)$). Several approaches have been proposed in literature, these include among others, conditional variational autoencoders [16], which are trained using the evidence lower bound, conditional generative adversarial networks [17], which are trained indirectly using an adversarial loss, conditional diffusion models [18], [19], [20], which are trained using denoising score-matching, and conditional normalizing flows [21], [22], which are trained using the exact log-likelihood.

In this work, we introduce U-Flow, an automatic sleep scoring algorithm based on conditional normalizing flows. U-Flow is able to learn how one PSG measurement is associated with multiple different hypnograms due to the human inter-rater disagreement, enabling us to estimate the inter-rater disagreement of overnight sleep statistics. We summarize our main contributions as follows:

- We propose U-Flow, a novel sleep staging algorithm based on conditional generative modeling.
- We show how the uncertainty of overnight sleep statistics predicted by an automatic sleep staging model can be compared to a human panel that scored the same recording.
- We found that joint modeling of the full hypnogram outperforms both factorized (cross entropy) based methods and Monte-Carlo dropout in its ability to capture uncertainty of overnight statistics.

II. Methods

A single PSG, when scored by a panel of human scorers will result in a set of slightly different hypnograms, depending on the experience and biases of each scorer. Fig. 1 shows an example of a PSG scored by a panel of six human technicians. From each hypnogram separately, overnight sleep statistics, such as total sleep time, can then be calculated. The uncertainty or spread in the estimation of the overnight sleep statistics by the human panel can be visualized by plotting the empirical cumulative distribution function (eCDF), which is shown all the way on the right of Fig. 1. The eCDF is constructed by a series of step functions, with each step located at an observed sample. In this example, the steps are located at 436 minutes, 444 minutes, etc, for the human panel. If we had access to an infinite number of scorers, the eCDF would approach the 'true' CDF. Plotting an eCDF is useful as several statistical measures, such as mean, variance, skewness, and the presence of outliers can easily be visually inferred.

In this manuscript, we argue for a similar approach using neural networks that also output a set of hypnograms given a PSG. The prediction pipeline for the hypnograms and the eCDF of each overnight statistic is then the same between the human panel and the automated method. It is then possible to evaluate the performance of the automated method based on two different categories. Firstly, the quality of the hypnograms themselves can be evaluated by comparing the majority-voted hypnogram of the human panel to that of the automated method in terms of accuracy, Cohen's kappa, and the per-class F1-scores. Secondly, the estimation of the inter-rater disagreement about the overnight sleep statistics can be evaluated by comparing the distribution as estimated by the automated method against the human panel using the kullback-leibler divergence, Kolmogorov-Smirnov metric, and the Wasserstein distance.

The rest of this section will be structured as follows. First, we will introduce the datasets that were used in this study and the preprocessing performed on them. Second, we will explore several posterior sampling methods that enable us to create multiple valid hypnograms from a single PSG measurement.

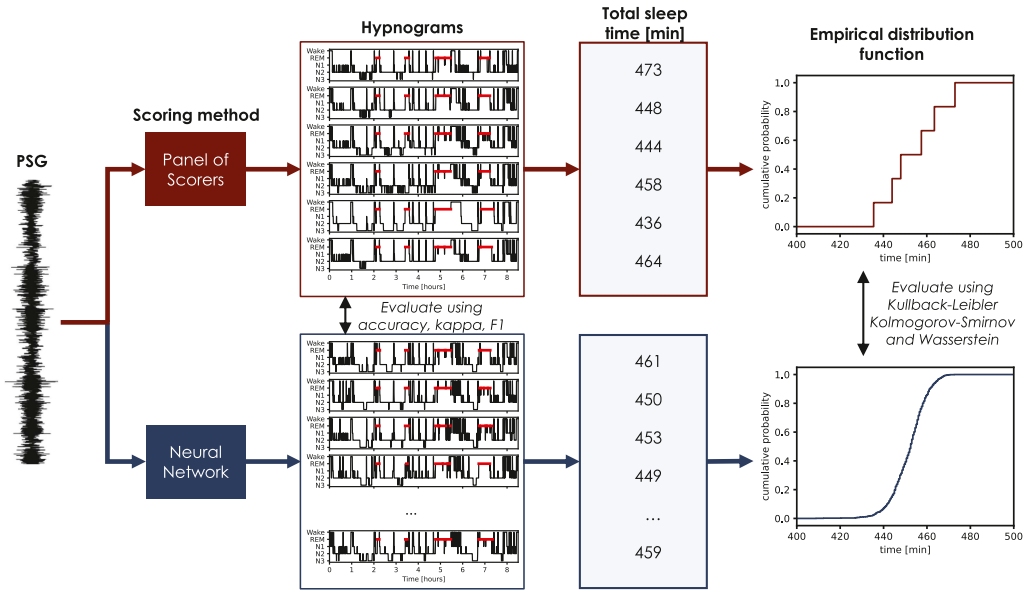

Fig. 1. Prediction and evaluation pipeline. Due to human inter-rater disagreement, a single PSG will result in a set of different hypnograms, and thus a set of estimates for each overnight sleep statistic, such as total sleep time. We show how automated methods can be employed in a similar way as a human panel. We evaluate the performance of the automated method by comparing the hypnograms in terms of accuracy, Cohen's kappa, and the F1-scores, and comparing the distribution of overnight sleep statistics using the kullback-leibler divergence, the Kolmogorov-Smirnov metric, and the Wasserstein distance.

TABLE I
DATASETS USED IN THIS STUDY

Dataset		#Recordings	#Scorings	Split
SSC	[23], [24]	661	1	validation (20%)
				train (80%)
ISRUC	[26]	110	2	train (100%)
DOD	[25]	80	5	train (100%)
IS-RC	[27]	70	6	test (100%)

'#Scorings' denotes the number of scorings per individual recording.

Third, we will introduce U-Flow, our approach to sample a hypnogram based conditionally on the PSG. Fourth, we cover our implementation of a U-Net baseline. The, we explain how both methods were trained. Last, we introduce the metrics that we used to determine how well the inter-rater disagreement of the overnight statistics was captured by the model.

A. Datasets

We make use of four different datasets: the Stanford Sleep Cohort (SSC) [23], [24], the Dreem Open Datasets (DOD) [25], the Institute of Systems and Robotics dataset (ISRUC) [26], and the Inter-Scorer Reliability Cohort (IS-RC) [27], see Table I . Together, the datasets comprise 921 overnight PSG recordings covering 7261 hours.

The complete SSC comprises thousands of overnight sleep recordings. Here we make use of the subset of SSC that is available through the National Sleep Research Resource [28], which consists of 772 PSGs of participants with varying sleep-related disorders, including sleep-disordered breathing, insomnia, and restless legs syndrome. We made use of this dataset in order to include a large amount and variety of sleep recordings in the training set. Each PSG recording in the dataset was associated with only a single ground truth hypnogram as scored by a human expert using the Rechtschaffen and Kales (R&K) rules [29]. The R&K scoring was harmonized to the AASM standard by merging S4 and S3 into N3. Additionally, S1 was kept unchanged as N1 and S2 as N2. REM and Wake were also not changed. We had to exclude 111 out of the 772 available PSGs because not all selected channels were available (see Section II-B). Of the remaining 661 PSGs, 529 (80%) were randomly chosen to be included in the training set and the remaining 132 (20%) were used in the validation set, which was used to monitor for over-fitting during training.

DOD consists of two separate subsets: DOD-H contains 25 PSG recordings of healthy participants, and DOD-O contains 55 recordings of obstructive sleep apnea (OSA) patients. Each recording in both DOD-O and DOD-H is scored by 5 different sleep experts using the AASM guidelines. All 80 recordings of both DOD-H and DOD-O are used in the training set in order to make the model learn that a single recording can result in multiple different, but valid, hypnograms.

ISRUC consists of three subsets. We only make use of subsets I and III as these contain PSGs scored by 2 different human experts, and thus enable the model to learn the uncertainty in scoring. The annotators followed the AASM guidelines, version unspecified. Subset I contains recordings of 100 participants with a variety of disorders and subset III contains 10 healthy participants. All 110 recordings of ISRUC are used in the training set.

IS-RC contains 70 PSGs of middle-aged women (51.1 ± 4.2 years) suspected of sleep-disordered breathing. Each PSG was scored by 10 human experts from 5 different sleep centers (2 experts per center) following the AASM guidelines. However, only the scorings from 3 centers (and thus 6 scorings per subject) are available. Because IS-RC was scored by a variety of experts coming from different sleep centers, we use it exclusively as a

hold-out test set. For one recording in the IS-RC set, only 5 out of 6 of the scorings were available, on which the subsequent analyses for that subject were performed.

B. Channel Selection and Preprocessing

We use the following signals from the PSG as input to all neural networks: EEG (C3-M2, C4-M1, O2-A1, O1-A2), EOG (LOC-A2, ROC-A1), EMG (EMG1-EMG2), and ECG. Because sampling frequencies differ between datasets by non-integer factors (SSC 256 Hz, DOD 250 Hz, ISRUC 200 Hz, IS-RC 128 Hz) we employ polyphase filtering to resample all channels of all datasets to the lowest available sampling frequency of 128 Hz. This subsampling does not impact relevant sleep staging information present in the EEG and EOG signals, where the AASM recommends the usage of a low pass filter at 35 Hz, and only minorly impacts the EMG and ECG signal, where the low pass filter is recommended to be at 100 Hz.

No additional signal validation to check for segments of bad quality or loss of signal was performed on top of what might already have been done by the original authors of each dataset. We cropped the IS-RC recordings to only contain epochs where the scoring from all scorers was available, on average trimming 93 minutes at the start of each recording and 122 minutes at the end. Since there is no lights-on/lights-off information available, we assume that scoring spans this period. This cropping equalizes the calculation of sleep statistics such as sleep onset latency between the six scorers, where we now start counting from the same moment in time. All PSG recordings were then zero-padded to a length of $7 * 2^8 = 1792$ epochs for implementation purposes. This additional padding was solely added for training purposes, and excluded when computing performance such as accuracy or Cohen's kappa.

As a final preprocessingxvv step, the data is rescaled using the following procedure [7]:

$$\tilde{\boldsymbol{x}} = \text{sign}(\boldsymbol{x}) \cdot \log\left(\frac{|\boldsymbol{x}|}{P_{95}(\boldsymbol{x})} + 1\right), \tag{1}$$

where $\boldsymbol{x}$ is the input signal, $\tilde{\boldsymbol{x}}$ is the scaled output, and $P_{95}(\boldsymbol{x})$ is the 95th magnitude percentile. We perform this scaling in the log-domain, as it is close to linear around 1, but helps push up very low signal values, and push down very high signal values. Moreover, scaling is done based on the 95th-percentile instead of the maximum value (100th-percentile), so as to be less sensitive to outliers in the ExG signals.

C. Posterior Sampling

Since a single PSG measurement can result in multiple hypnograms and multiple estimates of the overnight statistics, from the point of view of the automatic method, we have what is known as *aleatoric* uncertainty [30]. This can be modeled as a conditional probability distribution over all possible outputs: $p(\boldsymbol{h}|\boldsymbol{x})$, where $\boldsymbol{x}$ is the PSG and $\boldsymbol{h}$ is the hypnogram. Posterior sampling can then be leveraged as $\boldsymbol{h} \sim p(\boldsymbol{h}|\boldsymbol{x})$ to yield a set of plausible hypnograms given the input PSG. Similar to the panel of human scorers, the overnight sleep statistics can be calculated separately for each hypnogram $s = f_{stat}(\boldsymbol{h})$, which then allows us to calculate the eCDF for each overnight sleep statistic $P(s|\boldsymbol{x})$.

We implement three methods for posterior sampling: factorized sampling from the hypnodensity graph, Monte-Carlo (MC) dropout, and sampling from the joint distribution, see Fig. 2. We will now describe each of these methods.

1) Factorized Sampling: Current DL methods for automatic sleep staging are generally trained using cross entropy as a loss function. These methods use a softmax activation function to output a categorical distribution over sleep stages for each epoch. The hypnodensity graph is the visualization of all these categorical distributions over time [7]. More formally, a factorization of categorical distributions over epochs is assumed as:

$$p(\boldsymbol{h}_{1:T}|\boldsymbol{y}_{1:T}) = \prod_{t=1}^{T}\prod_{l=1}^{L} y_{t,l}^{[h_t=l]}, \tag{2}$$

where T is the number of epochs in the recording, L is the number of sleep staging labels, h_t is the sleep stage for the epoch at time t, and $y_{t,l}$ is the softmax output of the neural network at time t and for label l. The expression $[h_t = l]$ evaluates to 1, if $h_t = l$, and is 0 otherwise, thereby selecting only the relevant softmax outputs given the hypnogram $\boldsymbol{h}_{1:T}$. Cross entropy-based methods are however lacking in the sense that modeling the inter-rater disagreement of the overnight statistics can be difficult, because we need a set of discrete hypnograms for that. Taking the cross entropy loss to its logical conclusion, factorized posterior sampling is leveraged from (2), see the left column of Fig. 2.

2) MC Dropout and Bayesian Neural Networks: In Bayesian neural networks, the learned parameters of the network are not points, but rather distributions [31], [32], [33]. During each pass through the network, the parameters for that pass are sampled from the learned distributions, resulting in different outputs each time, even for the same input data, see the central column Fig. 2. The simplest form of such a Bayesian network is MC dropout [34]. In MC dropout, random connections in the neural network are set to zero by sampling from a Bernoulli distribution. While often only employed during the training phase to combat over-fitting, MC dropout can also be employed during inference to yield diverse outputs. It has been shown that MC dropout outperforms other Bayesian approaches [32], [33]. While MC dropout has been used in sleep staging before to quantify uncertainty per epoch [11], it has, to the best of our knowledge, not yet been employed to generate diverse hypnograms and estimate the uncertainty of summarizing sleep statistics. We employ MC dropout on the same network architecture as is used for the factorized approach as a means of posterior sampling, see the middle column of Fig. 2.

3) Joint Modeling: Modeling the joint distribution $p(\boldsymbol{h}_{1:T}|\boldsymbol{x}_{1:T})$ directly, instead of its factorized form, is achieved here by employing conditional generative networks. Conditional generative approaches take randomly sampled latent variables $z \sim N(0, I)$ and map them to the desired signal, where this mapping is conditioned on some context. In our case, the desired signal would be the hypnogram and the context would be the PSG, see the right column Fig. 2. Several approaches have been proposed in literature, these include among others, conditional

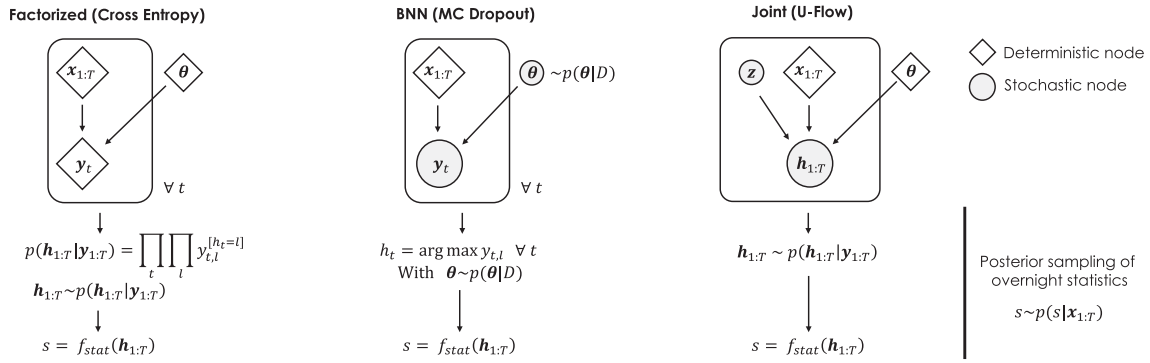

Fig. 2. Graph models of posterior sampling of the overnight statistics. Shown here are factorized, MC dropout, and joint modeling methods.

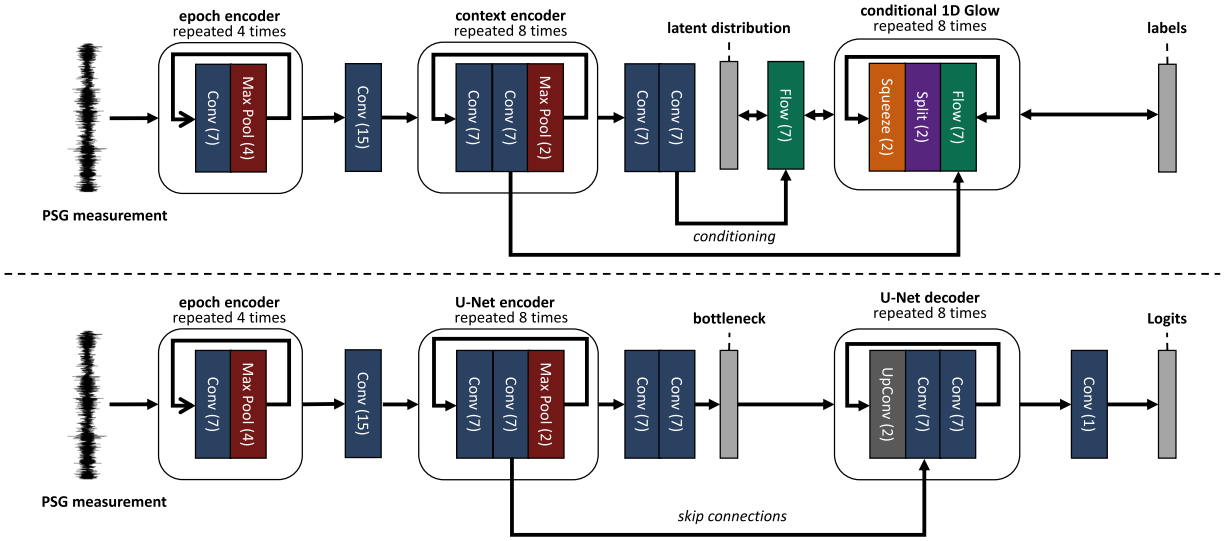

Fig. 3. Architecture of the proposed U-Flow network (top) and the U-Net baseline (bottom).

variational autoencoders [16], conditional generative adversarial networks [17], conditional diffusion models [18], [19], [20], and conditional normalizing flows [21], [22]. To the best of our knowledge, none of these approaches have been applied to sleep staging. Here we choose to make use of conditional normalizing flows, as these methods are efficient to sample from and can be trained on the exact likelihood function $p(\boldsymbol{h}_{1:T}|\boldsymbol{x}_{1:T})$.

D. U-Flow (Joint)

Normalizing Flows (NFs) are a type of generative neural network that learn to transform a latent base distribution, often modelled as multivariate normal distribution, into a more complex data distribution, by using a series of learned invertible mappings. Commonly used architectures for NFs include RealNVP [35] and Glow [36]. We refer the reader to the review paper by Kobyzev et al. for a more thorough introduction and overview of Normalizing Flows [37].

1) Architecture: Here, we adopt a Glow network into a 1D structure (instead of the original 2D structure aimed at image generation), termed U-Flow. U-Flow takes a base distribution $p_Z(\boldsymbol{z}_{1:T}) \sim \mathcal{N}(0, I)$ and maps it to the hypnogram $\boldsymbol{h}_{1:T}$. Additionally, this mapping is conditioned on a learned context $\boldsymbol{c}_{0:8}$ extracted from the PSG signal $\boldsymbol{x}$ at different resolution levels, where each resolution level corresponds to a halving in size, see Fig. 3. U-Flow can then be described as:

$$\boldsymbol{z}_{1:T} = f_{Glow}(\boldsymbol{h}_{1:T}|\boldsymbol{c}_{0:8}), \quad \boldsymbol{h}_{1:T} = f^{-1}_{Glow}(\boldsymbol{z}_{1:T}|\boldsymbol{c}_{0:8}). \quad (3)$$

The learned conditioning vector is created by a convolutional neural network that consists of two parts, an epoch encoder and a context encoder. The epoch encoder takes the full PSG measurement as input, which is of length $\#epochs \cdot 30s \cdot F_s$. Through a series of convolutional layers of kernel size 7, max pooling of kernel size 4, and a final convolution of kernel size and stride 15, the epoch encoder compresses the input to a length of $\#epochs$.

The epoch encoding is then fed into the context encoder, which consists of 8 blocks, each with two convolutional layers of kernel size 7 followed by a max pooling operation of size 2. The output of the max pooling operation at each resolution level r is used as part of the conditioning vector $\boldsymbol{c}_r$. Each convolutional layer is followed by a ReLU activation function

Algorithm 1: U-Flow Training.

Require: dataset $\mathcal{D}$, initial model parameters $\boldsymbol{\theta}$, number of iterations N, learning rate α, epoch encoder $f^{\boldsymbol{\theta}}_{Epoch}(.)$, context encoder $f^{\boldsymbol{\theta}}_{Context}(.)$, normalizing flow model $f^{\boldsymbol{\theta}}_{Glow}(.)$

// loop over dataset iterations
for $n = 0$ **to** $N - 1$ **do**
  // loop over mini-batches
  foreach $\boldsymbol{x}_{batch}, \boldsymbol{h}_{batch}$ *in* $\mathcal{D}$ **do**
    // encode PSG data
    $\boldsymbol{c} \leftarrow f^{\boldsymbol{\theta}}_{Context}(f^{\boldsymbol{\theta}}_{Epoch}(\boldsymbol{x}_{batch}))$
    // map to latent space
    $\boldsymbol{z}, \log|\det J| \leftarrow f^{\boldsymbol{\theta}}_{Glow}(\boldsymbol{h}_{batch}, \boldsymbol{c})$
    // calculate loss and update parameters
    $\mathcal{L} \leftarrow |\boldsymbol{z}|_2^2 - \log|\det J|$
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} \mathcal{L}$

Return: trained model parameters $\boldsymbol{\theta}$

Algorithm 2: U-Flow Inference.

Require: trained model parameters $\boldsymbol{\theta}$, PSG data $\boldsymbol{x}$, number of samples M, epoch encoder $f^{\boldsymbol{\theta}}_{Epoch}(.)$, context encoder $f^{\boldsymbol{\theta}}_{Context}(.)$, normalizing flow model $f^{\boldsymbol{\theta}}_{Glow}(.)$

// encode PSG data
$\boldsymbol{c} \leftarrow f^{\boldsymbol{\theta}}_{Context}(f^{\boldsymbol{\theta}}_{Epoch}(\boldsymbol{x}))$
// create emtpy list to store outputs
$\boldsymbol{h}_{list} \leftarrow (empty\ list)$
// loop over samples
for $m = 0$ **to** $M - 1$ **do**
  // sample using inverse flow model
  $\boldsymbol{z} \sim \mathcal{N}(0, I)$
  $\boldsymbol{h} \leftarrow {f^{\boldsymbol{\theta}}_{Glow}}^{-1}(\boldsymbol{z}, \boldsymbol{c})$
  // store
  $\boldsymbol{h}_{list}[m] \leftarrow \boldsymbol{h}$

Return: list of sampled hypnograms $\boldsymbol{h}_{list}$

and a dropout layer with probability 0.5 for an element to be zeroed out. Moreover, all convolutions (except the last epoch encoder convolution) make use of padding = 'same' (e.g. a padding of 3 zeroes in the case of a kernel of size 7).

In contrast to the context encoder that progressively lowers in size, the Glow model progressively increases in size using 8 levels. We modify the standard squeezing and splitting operations to work on the 1D hypnogram. Each flow consists of 6 steps that are conditioned on the $\boldsymbol{c}_r$ of the same resolution. The coupling network is implemented using three convolutional layers of kernel size 7, followed by a ReLU and activation normalization. See the supplemental material for a more thorough explanation of these three building blocks (squeeze, split, and flow).

2) Loss Function: U-Flow is trained using the exact negative log-likelihood of the hypnogram as given by the change of variables formula:

$$\begin{aligned}\mathcal{L}_{f_{flow}} &= -\log p_H(\tilde{\boldsymbol{h}}_{1:T}|\boldsymbol{x}_{1:T}) \\ &= -\log(p_Z(f_{Glow}(\tilde{\boldsymbol{h}}_{1:T}|\boldsymbol{c})) \cdot |\det J f_{Glow}(\tilde{\boldsymbol{h}}_{1:T}|\boldsymbol{c})|) \\ &= |\boldsymbol{z}|_2^2 - \log|\det J f_{Glow}(\tilde{\boldsymbol{h}}_{1:T}|\boldsymbol{c})|, \end{aligned} \tag{4}$$

where $\det J$ is the determinant of the Jacobian of the Glow model, which accounts for the change of probability density, and $\tilde{h}_t$ is the one-hot encoded ground truth hypnogram. Note how no factorization over epochs is applied in this loss function, rather the loss function takes the entire joint distribution into account. Because the change of variables formula is defined in continuous space, but the hypnograms are discrete, we apply dequantization using triangular noise:

$$\tilde{\boldsymbol{h}}_{deq} = 0.5\tilde{\boldsymbol{h}} + 0.25\boldsymbol{u}_1 + 0.25\boldsymbol{u}_2, \tag{5}$$

with $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ i.i.d. samples from the uniform distribution. We provide pseudo-code for the training loop in algorithm 1.

3) Inference: In inference, the conditioning of the U-Flow model from the PSG is paired with 1024 i.i.d. sampled latent vectors $\boldsymbol{z}$. The Glow model is then run in the generative direction for each conditioning-latent code pair, resulting in 1024 different hypnograms, all belonging to the same PSG, see algorithm 2 for the accompanying pseudo-code.

E. U-Net (Factorized and MC Dropout)

1) Architecture: We create a U-Net of similar structure to U-Flow. The U-Net decoder also consists of 8 blocks. Each block consists of an upconvolutional layer of size 2, whose output is concatenated with the skip connections of the encoder. This is used as input for 2 convolutions of kernel size 7 with a ReLU activation function and a dropout layer with probability 0.5 for an element to be zeroed out. As a final step, a convolutional layer of kernel size 1 is used to output the logits $\boldsymbol{\phi}$ (log-probability) of each sleep stage, which can be mapped to the probabilities $\boldsymbol{y}$ using the softmax activation function.

2) Loss Function: The U-Net is trained using cross entropy, which assumes a factorization of categorical distributions over the epochs:

$$\mathcal{L}_{\text{cross entropy}} = \sum_{t=1}^{T} \tilde{\boldsymbol{h}}_t \cdot \log(\boldsymbol{y}_t). \tag{6}$$

3) Inference: During inference, U-Net can be used in two modes: factorized sampling or MC dropout. In the case of factorized sampling, one forward pass through the network is performed, yielding the logits of each class. Gumbel sampling is used to sample from the categorical distribution specified by the logits [38]. This is done using 1024 i.i.d. samples of Gumbel noise, $\boldsymbol{\epsilon}$, to get 1024 different hypnograms:

$$\boldsymbol{h}_t = \arg\max \boldsymbol{\phi}_t + \boldsymbol{\epsilon}_t \quad \forall t. \tag{7}$$

In MC dropout mode, all dropout layers are enabled even during inference. Now, 1024 individual passes are performed through the network using the same input PSG data. Because the dropout layers will zero out different connections each time, this method will result in 1024 different hypnograms.

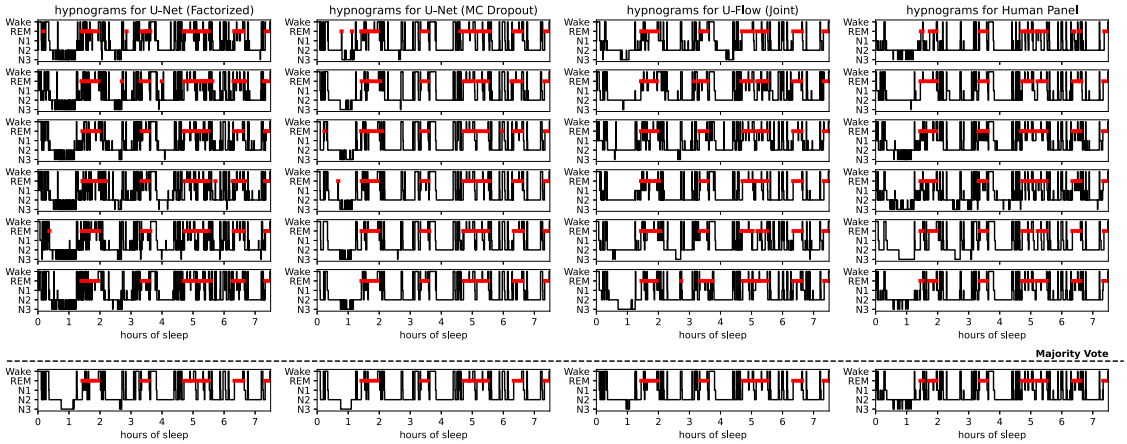

Fig. 4. Example results for the PSG in the test set where U-Flow has median performance in terms of accuracy. Shown here are 6 samples (out of 1024) from each DL method, as well as the 6 ground truth hypnograms made by the panel of human scorers.

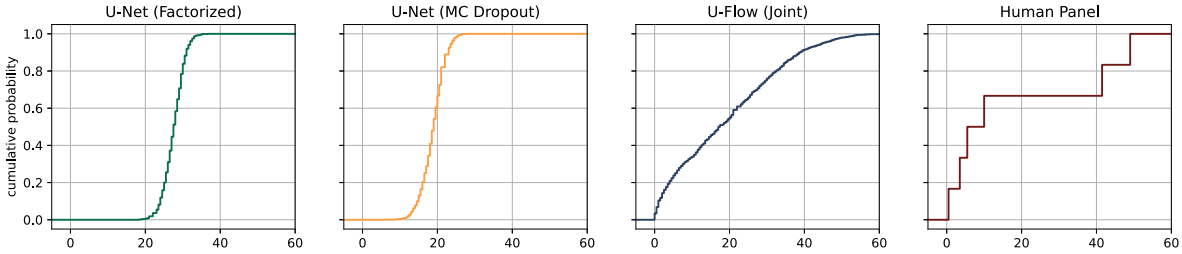

Fig. 5. Empirical distribution functions for 'time in N3' calculated from the hypnograms shown in Fig. 4.

F. Training Strategy

Both U-Flow and U-Net are trained with their respective loss function for 200 dataset iterations using mini-batches of two whole-night PSGs. We use the Adam optimizer [39] with parameters: $lr = 10^{-4}$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. After 100 dataset iterations the learning rate is lowered to 10^{-5} to aid with convergence.

G. Overnight Statistics Metrics

1) Kullback-Leibler Divergence: To calculate the Kullback-Leibler divergence, we first fit a normal distribution to the sampled overnight statistics for each recording for each method by calculating the sample mean and standard deviation. Then the Kullback-Leibler divergence can be calculated as:

$$KL\left(\mathcal{N}_1, \mathcal{N}_2\right) = \log\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{\sigma_1^2 + (\mu_1 - \mu_2)^2}{2\sigma_2^2} - \frac{1}{2}, \quad (8)$$

where $\mathcal{N}_1$ is the normal distribution as estimated from the human panel, and $\mathcal{N}_2$ the normal distribution as estimated from one of the DL methods. Moreover, μ_1 and μ_2 are the means and σ_1 and σ_2 the standard deviations of the two normal distributions. For numerical stability, division by zero is counteracted by clipping the minimum values of the standard deviations at 10^{-2}. Note that the Kullback-Leibler divergence is calculated for each recording for each overnight statistic separately.

2) Kolmogorov-Smirnov Metric: The Kolmogorov-Smirnov metric can be calculated from the eCDFs for each recording for overnight statistic as:

$$KS\left(F_1(s), F_2(s)\right) = \sup_s |F_1(s) - F_2(s)|, \quad (9)$$

where $F_1(s)$and $F_2(s)$ are the eCDFs of the human panel and DL method, respectively. Moreover, $\sup_s$ is the supremum function which returns the maximum absolute difference between the two eCDFs. A visual representation of the Kolmogorov-Smirnov metric can be found in the supplementary material.

3) Wasserstein Distance: The Wasserstein distance, also known as the earth mover's distance or the Kantorovich–Rubinstein metric, expresses how much effort is required to change one probability function into another. We here make use of the 1-Wasserstein distance, which can be calculated efficiently by using the cumulative distributions:

$$W_1\left(F_1(s), F_2(s)\right) = \int_{\mathcal{R}} |F_1(s) - F_2(s)|\, ds. \quad (10)$$

A visual representation of the Wasserstein distance can also be found in the supplementary material.

III. Results

A. Qualitative Example

A qualitative example where U-Flow achieves median performance in terms of accuracy is shown in Fig. 4. This figure also shows the hypnograms for the same subject as scored by the human panel and the two U-Net approaches: factorized and MC dropout sampling. Note that we only show 6 out of the 1024

TABLE II
METRICS ON THE PREDICTED DISTRIBUTIONS OF OVERNIGHT STATISTICS OF EACH MODEL COMPARED TO THE HUMAN PANEL

	U-Net (Factorized)			U-Net (MC Dropout)			Stanford [7] (Factorized)			U-Sleep [8] (Factorized)			U-Flow (Joint)		
	KL	WD	KS	KL	WD	KS	KL	WD	KS	KL	WD	KS	KL	WD	KS
Total Sleep Time [min]	41.1	23.7	0.83	32.5	21.5	0.80	59.2	17.3	0.75	107	14.0	0.78	**1.34**	*13.3*	**0.53**
Sleep Efficiency [%]	41.1	5.04	0.83	32.5	4.57	0.80	59.2	3.62	0.75	107	2.91	0.78	**1.34**	*2.77*	**0.53**
Sleep Onset Latency [min]	9.27	6.21	0.43	12.0	7.76	0.56	42.1	*3.55*	0.49	699	4.02	0.52	*2.11*	5.82	*0.41*
REM latency [min]	*2.20*	23.7	0.47	2.76	24.8	0.61	28.1	*15.6*	0.49	5.73	29.9	0.47	3.08	18.9	*0.46*
Wake After Sleep Onset [min]	33.5	26.0	0.83	14.2	20.3	0.73	33.2	16.7	0.73	50.4	*11.8*	0.71	**1.08**	13.8	**0.51**
REM awakenings [-]	1.30	2.70	0.55	1.02	1.63	0.39	1.71	2.65	0.49	3.20	1.88	0.48	**0.66**	*1.26*	**0.31**
NREM awakenings [-]	14.0	23.0	0.95	5.37	5.74	*0.55*	5.04	7.41	0.56	5.81	*5.34*	0.56	**1.38**	7.55	0.56
Time in N1 [min]	13.2	14.9	0.56	111	25.5	0.84	34.7	16.6	0.63	70.5	16.4	0.65	**0.78**	**12.7**	**0.42**
Time in N2 [min]	68.8	46.9	0.74	72.1	38.9	0.65	124	35.0	0.60	231	38.0	0.67	**1.57**	**25.8**	**0.44**
Time in N3 [min]	87.8	36.9	0.68	187	35.8	0.68	332	30.3	0.60	343	36.9	0.71	**2.31**	**19.6**	**0.41**
Time in REM [min]	81.2	15.3	0.63	29.6	16.0	0.69	84.2	16.3	0.67	124	17.2	0.75	**1.84**	*13.1*	**0.47**

Mean values over the hold-out test set are shown. KL: kullback-leibler divergence. KS: kolmogorov-smirnov metric. WD: wasserstein distance. If the best value was significant it is shown in bold, otherwise it is italic.

hypnograms sampled with the DL methods. The figure shows that sampling from a factorized distribution as learned by a cross entropy loss function leads to too many state transitions. This is because contextual information about dependencies between epochs is lost through factorization. On the other hand, MC dropout leads to too conservative estimates of the amount of state transition. Both of these sampling methods, however, greatly underestimate the amount of uncertainty or inter-rater disagreement for the overnight statistic 'time in N3', which is shown in the eCDFs of the same recording, illustrated in Fig. 5. Here it can be seen that the eCDFs of these two methods form very sharp s-bends at around 20 and 30 minutes. The actual values for 'time in N3' as calculated from the human panel range from 0 to 50 minutes, a spread in scoring that our proposed U-Flow was much better able to capture. The eCDFs for all statistics for this example can be found in the supplementary material.

B. Comparison With Other Approaches

We compared U-Flow quantitatively to the factorized and MC dropout U-Net baseline. Additionally, we also compared against two automatic scoring algorithms from the literature, the Stanford staging algorithm [7] and U-Sleep [8], both of which are factorized approaches.

C. Uncertainty Estimation of Overnight Statistics

For each recording in the hold-out test, we calculated 1024 and 6 samples of each overnight statistic from the provided hypnograms of the DL models and the human panel, respectively. We first compared the predicted samples to the ground truth samples by modeling each statistic for each recording as a normal distribution following the observed sample mean and standard deviation. The normal distributions of the DL model and human panel were then compared in terms of Kullback-Leibler divergence. The mean Kullback-Leibler divergence over the hold-out test set for each statistic is shown in Table II , where lower is better.

To test for statistical significance, a paired student's t-test was applied between the best and second-best performing method for each overnight sleep statistic. We choose a significance level of $\alpha = 0.05$ and applied a Bonferroni correction coefficient of 11 due to repeated testing over the 11 overnight sleep statistics. This resulted in eight significant differences between the methods, each with U-Flow achieving the lowest Kullback-Leibler divergence. The differences in Kullback-Leibler divergence for the sleep- and REM onset latency were not statistically significant.

We also evaluated the predicted samples using non-parametric metrics on the eCDFs, namely the Wasserstein distance, and the Kolmogorov-Smirnov metric, see Table II. We again test for statistical significance using the same procedure. This resulted in three and seven statistically significant differences, for the Wasserstein and Kolmogorov-Smirnov metrics, respectively. All statistically significant differences were in favor of the proposed U-flow model, indicating that it was better able to model the inter-rater disagreement of the human panel than the other automatic scoring algorithms.

D. Evaluating the Hypnograms

To compare the hypnograms of the automatic scoring algorithms with those of the human panel we used majority voting. The majority vote of the 1024 samples from each model was compared to that of 6 human scorers. For the human scorers, ties were handled by using the scoring of the most reliable scorer, following the procedure proposed by Guillot et al. [25]. We compared the majority voted hypnograms in terms of accuracy, Cohen's kappa, and the per-class F1-scores on an epoch-by-epoch basis, the resulting averages across the subjects are shown in Table III .

We again tested for statistical significance using the paired student's t-test with a significance level of $\alpha = 0.05$ using a Bonferroni correction coefficient of 7. The majority voting results are highly comparable between methods, and the only significant difference was found to be the F1 score for the N1 stage. Per-subject results can be found in the supplementary material.

TABLE III
ACCURACY, COHEN'S KAPPA, AND PER CLASS F1 SCORES

	U-Net (Factorized)	U-Net (MC Dropout)	Stanford [7] (Factorized)	U-Sleep [8] (Factorized)	U-Flow (Joint)
Accuracy	77.7	79.5	*84.1*	83.3	83.2
Cohen's kappa	0.667	0.692	*0.751*	0.747	0.735
F1 - Wake	0.795	0.804	0.857	*0.869*	0.844
F1 - N1	0.290	0.279	0.385	**0.515**	0.404
F1 - N2	0.816	0.835	*0.892*	0.866	0.877
F1 - N3	0.496	0.504	*0.588*	0.549	0.482
F1 - REM	0.855	0.884	0.857	*0.902*	0.878

Mean values over the hold-out test set are shown. The metrics are computed by comparing the majority vote of each model against the panel of scorers. If the best value was significant it is shown in bold, otherwise it is italic.

IV. DISCUSSION

In this manuscript, we proposed to model the relationship between PSG and hypnograms as a joint conditional probability function. This joint modeling approach, which we implemented using the U-Flow architecture, allows for the creation of a set of valid hypnograms from a single PSG measurement, similar to how a panel of human scorers would have scored the PSG. Going beyond previously proposed cross entropy approaches, our method enables the modeling of the inter-rater disagreement not only on a per epoch level but also at the level of overnight sleep statistics.

A. Hypnodensity

Since the introduction of the hypnodensity concept in 2018, thinking probabilistically about the relationship between sleep recording and hypnogram has been gaining traction. Our approach is an extension of this line of thinking and fully compatible with the hypnodensity concept. To get a hypnodensity graph using our approach, a similar strategy to getting the hypnodensity of a human panel can be applied. Namely, summing the occurrence of each sleep stage divided by the total number of scorers or samples, in case of a human panel or U-Flow model, respectively. However, unlike the original hypnodensity estimator, our new approach also permits sampling valid hypnograms from the learned probability distribution, thereby enabling assessment of downstream uncertainty of overnight sleep statistics derived from hypnograms. Therefore, in this manuscript, we focused on comparing the uncertainty estimates of the overnight sleep statistics. Furthermore, to evaluate sleep staging performance we compared the majority-voted hypnograms, rather than comparing the hypnodensity graphs, as the former is still the gold standard in clinical practice and the outcome of the PSG that clinical experts look at in daily practice.

While the hypnodensity approach has recently become more popular, it has not yet led to probabilistic thinking about the overnight sleep statistics. For example, whereas Bakker et al. [3] argue in favor of hypnodensity, they recommend calculating the overnight statistics as deterministic point estimates (e.g., sleep onset latency as the first epoch with a sleep probability > 0.5). Probabilistic thinking about the overnight statistics for human panels has been explored in literature before, especially in terms of (transformed) Gaussians [27], [40]. In the present study, we extend this line of thinking, for the first time, to both human panels and automatic scoring methods.

B. Inter-Rater Disagreement

Acquiring robust estimates of the impact of inter-rater disagreement on overnight sleep statistics is important, as variability in the overnight statistics could impact the diagnostic process. For example, disagreement about the time to REM sleep could impact a narcolepsy diagnosis [41]. Moreover, the number of awakenings may impact the physician's perception of how much sleep is fragmented and potentially has a large impact on the assessment of an abnormal sleep pattern. Additionally, basic sleep research findings that correlate overnight statistics to clinically relevant outcomes are also impacted by inter-rater disagreement. For instance, it has been found that increased time in N1 and reduced time in REM are associated with worsening cognitive performance in older men over time [42], whereas a lower percentage of time in N3 is associated with hypertension in both men and women [43]. However, it has been shown that time in N1 and time in N3 are particularly sensitive to inter-rater disagreement [44], possibly limiting the strength of these findings. This is not limited to N1 and N3, and many more examples of basic sleep research studies on overnight statistics can be found, all possibly confounded by ambiguity in the scoring of sleep [45], [46], [47], [48], [49].

C. Generalizability

Whereas most of the recordings in the training set (74%) only had a single hypnogram as ground truth available, U-Flow still extrapolated well to the multi-scored hold-out test set. This is similar to findings reported in the literature, where this type of extrapolation was also found [3], [7]. Moreover, none of the six scorers in the hold-out test set were present in the training set, but U-Flow still managed to both qualitatively and quantitatively model their inter-rater disagreement. Whereas the idea of a scoring 'bias' can be hard to quantify, this suggests that U-Flow learned to cover a multitude of scoring 'biases'. Recent work by Huijben et al. explores the phenomenon of extrapolating to more scorers, showing that an automatic scoring method converges to the labeling distribution found in the training set, even if each training example only has a single 'hard' label [50].

D. Limitations

A potential limitation of this study is the relatively homogeneous nature of the hold-out test set, which comprises 70 recordings of middle-aged women suspected of sleep-disordered breathing. Further investigation into larger cohorts, with different disorders, age groups, and disease severities, is required to evaluate their effect on the performance of the proposed method. It has for example been observed that sleep-disordered

breathing and narcolepsy can negatively impact the inter-rater agreement of human scorers [51], [52]. Additionally, it has been observed that the inter-rater agreement of subjects submitted to an intensive care unit is also relatively poor [53].

As the first application of conditional generative modeling for sleep staging, no extensive hyperparameter search or tuning was performed. These hyperparameters include, among others, the number of training iterations, kernel sizes, and the number of posterior samples. The tuning of these hyperparameters could lead to an increase in the performance of the methods we implemented (U-Net and U-Flow) and bring their majority voting results even closer to the performance of the literature baselines. However, the theoretical difference in loss functions and sampling principles between U-Flow and the (literature) baselines remains the same. Which we have shown to work better for the purpose of estimating the inter-rater disagreement of the overnight sleep statistics.

E. Future Work

This work opens up several avenues for future research. Firstly, only the usage of conditional normalizing flows was explored. We leave the investigation of other conditional generative models to future work. Especially the exploration of score-based diffusion models would be an interesting avenue of research here, as these have gained recent popularity due to their stable training mechanics and ability to generate high-fidelity outputs [18], [19], [20]. Moreover, future work could extend our approach to the estimation of overnight sleep statistics calculated from other means than the hypnograms, such as the apnea-hypopnea index. The scoring of disordered breathing events could likewise be modeled using a joint conditional generative approach. We also leave to future work the scoring of "sub-epoch events", such as arousals.

Additionally, the proposed joint modeling technique could be applied to surrogate measurements that assess sleep stages in an indirect way, such as photoplethysmography, suprasternal pressure, or actigraphy. These methods are advantageous over conventional PSG, as they are more comfortable to wear for subjects, cheaper to use, and can be used for longitudinal and ambulatory studies. The disadvantage of surrogate sleep trackers is their lower performance, especially in the case of fragmented sleep [54], [55]. One possible reason for this lower performance is the lower amount of information about sleep stages present in these surrogate measures. We thus have a large uncertainty about which sleep stages to assign to each of the epochs. Our method, which allows for the quantification of this uncertainty on both a per-epoch level and on the level of overnight sleep statistics, would be well suited for this task. By accurately estimating these uncertainties, U-Flow could aid with the acceptance of such surrogate sleep trackers in clinical practice. As it would allow physicians and other sleep experts to better interpret the data from such surrogate trackers, taking their uncertainty, due to e.g. low signal quality, into account.

Moreover, the quantification of uncertainty has potential clinical uses, such as the detection of sleep disorders linked to the obfuscation of standard sleep patterns. It would for example be interesting to investigate whether diseases such as Parkinson's lead to a larger degree of uncertainty about the overnight sleep statistics. Last, but not least, the proposed method allows to more cheaply study the effects of inter-rater disagreement in sleep staging on sleep diagnosis and basic sleep research. See for example the effects that are mentioned in Section IV-B, such as the physician's perception of a patient or the linking of time in N3 to hypertension.

It remains an open question how to best present the outcomes of our method in standard clinical practice. It would be unreasonable to expect physicians to sift through a multitude of hypnograms that all fit with the sleep measurement. We hypothesize that it would be best to present them with the majority-voted hypnogram from all the individual realizations, e.g. the last row of Fig. 4, plus an optional hypnodensity graph. Epochs with high variation between the different realizations can then also be marked as uncertain. For the overnight statistics, the uncertainty can be expressed visually using the cumulative distribution functions as presented in Fig. 5. Alternatively, the uncertainty can be expressed using the mean plus/minus the standard deviation or the median and the interquartile range. That way, it becomes easily interpretable how much variation and uncertainty there is for each of the overnight statistics. Future work should explore which presentation method is preferred by clinical experts and its effect on the quality of care.

Acknowledgment

At the time of writing, H.G., P.F., and R.J.G.S. 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. P.F. reports personal fees from Philips Research during the conduct of the study; personal fees from Philips Research, outside the submitted work. S.O. received an unrestricted research grant from UCB Pharma and participated in advisory boards for UCB Pharma, Jazz Pharmaceuticals, and Bioproject, all paid to the institution and all unrelated to the present work.

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*. Westchester, IL, USA: American Academy of Sleep Medicine, 2007.

[2] R. S. Rosenberg and S. Van Hout, "The american academy of sleep medicine inter-scorer reliability program: Sleep stage scoring," *J. Clin. Sleep Med.*, vol. 9, pp. 81–87, Jan. 2013.

[3] J. P. Bakker et al., "Scoring sleep with artificial intelligence enables quantification of sleep stage ambiguity: Hypnodensity based on multiple expert scorers and auto-scoring," *Sleep*, vol. 46, Jul. 2023, Art. no. zsac154.

[4] H. Danker-hopfe et al., "Interrater reliability for sleep scoring according to the Rechtschaffen & Kales and the new AASM standard," *J. Sleep Res.*, vol. 18, no. 1, pp. 74–84, 2009.

[5] Y. J. Lee, J. Y. Lee, J. H. Cho, and J. H. Choi, "Interrater reliability of sleep stage scoring: A meta-analysis," *J. Clin. Sleep Med.*, vol. 18, no. 1, pp. 193–202, 2022.

[6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[7] J. B. Stephansen et al., "Neural network analysis of sleep stages enables efficient diagnosis of narcolepsy," *Nature Commun.*, vol. 9, Dec. 2018, Art. no. 5229.

[8] M. Perslev, S. Darkner, L. Kempfner, M. Nikolic, P. J. Jennum, and C. Igel, "U-Sleep: Resilient high-frequency sleep staging," *NPJ Digit. Med.*, vol. 4, Apr. 2021, Art. no. 72.

[9] H. Phan, K. Mikkelsen, O. Y. Chen, P. Koch, A. Mertins, and M. de Vos, "SleepTransformer: Automatic sleep staging with interpretability and uncertainty quantification," *IEEE Trans. Biomed. Eng.*, vol. 69, no. 8, pp. 2456–2467, Aug. 2022.

[10] E. Eldele et al., "An attention-based deep learning approach for sleep stage classification with single-channel EEG," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 809–818, 2021.

[11] L. Fiorillo, P. Favaro, and F. D. Faraci, "DeepSleepNet-Lite: A simplified automatic sleep stage scoring model with uncertainty estimates," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 2076–2085, 2021.

[12] A. Supratak and Y. Guo, "TinySleepNet: An efficient deep learning model for sleep stage scoring based on raw single-channel EEG," in *Proc. IEEE 42nd Annu. Int. Conf. Eng. Med. Biol. Soc.*, 2020, pp. 641–644.

[13] M. Radha et al., "A deep transfer learning approach for wearable sleep stage classification with photoplethysmography," *NPJ Digit. Med.*, vol. 4, Sep. 2021, Art. no. 135.

[14] H. Phan and K. Mikkelsen, "Automatic sleep staging of EEG signals: Recent development, challenges, and future directions," *Physiol. Meas.*, vol. 43, Apr. 2022, Art. no. 04TR01.

[15] 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, 2023, Art. no. zsad028.

[16] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, vol. 28.

[17] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[18] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2256–2265.

[19] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *Proc. Int. Conf. Learn. Representations*, 2020.

[20] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," 2022, *arXiv:2204.06125*.

[21] C. Winkler, D. Worrall, E. Hoogeboom, and M. Welling, "Learning likelihoods with conditional normalizing flows," 2019, *arXiv:1912.00042*.

[22] A. Abdelhamed, M. A. Brubaker, and M. S. Brown, "Noise flow: Noise modeling with conditional normalizing flows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 3165–3173.

[23] O. Andlauer et al., "Nocturnal rapid eye movement sleep latency for identifying patients with narcolepsy/hypocretin deficiency," *JAMA Neurol.*, vol. 70, pp. 891–902, Jul. 2013.

[24] H. Moore IV et al., "Design and validation of a periodic leg movement detector," *PLoS One*, vol. 9, no. 12, pp. 1–30, 2014.

[25] A. Guillot, F. Sauvet, E. H. During, and V. Thorey, "Dreem open datasets: Multi-scored sleep datasets to compare human and automated sleep staging," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 9, pp. 1955–1965, Sep. 2020.

[26] S. Khalighi, T. Sousa, J. M. Santos, and U. Nunes, "ISRUC-Sleep: A comprehensive public dataset for sleep researchers," *Comput. Methods Prog. Biomed.*, vol. 124, pp. 180–192, 2016.

[27] S. T. Kuna et al., "Agreement in computer-assisted manual scoring of polysomnograms across sleep centers," *Sleep*, vol. 36, no. 4, pp. 583–589, 2013.

[28] G.-Q. Zhang et al., "The national sleep research resource: Towards a sleep data commons," *J. Amer. Med. Inform. Assoc.*, vol. 25, pp. 1351–1358, Oct. 2018.

[29] A. Rechtschaffen and A. Kales, *A Manual for Standardized Terminology, Techniques and Scoring System for Sleep Stages in Human Subjects*. Los Angeles, CA, USA: Brain Research Institute. US Dept. of Health, Education, and Welfare, 1968.

[30] 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, Jun. 2022, Art. no. zsac134.

[31] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural network," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, pp. 1613–1622.

[32] F. K. Gustafsson, M. Danelljan, and T. B. Schon, "Evaluating scalable Bayesian deep learning methods for robust computer vision," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 318–319.

[33] I. Osband, Z. Wen, M. Asghari, M. Ibrahimi, X. Lu, and B. Van Roy, "Epistemic neural networks," 2021, *arXiv:2107.08924*.

[34] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1050–1059.

[35] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real NVP," in *Proc. Int. Conf. Learn. Representations*, 2017.

[36] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," in *Proc. Neural Inf. Process. Syst.*, 2018.

[37] I. Kobyzev, S. J. D. Prince, and M. A. Brubaker, "Normalizing flows: An introduction and review of current methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 3964–3979, Nov. 2021.

[38] I. A. M. Huijben, W. Kool, M. B. Paulus, and R. J. G. van Sloun, "A review of the gumbel-max trick and its extensions for discrete stochasticity in machine learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 2, pp. 1353–1371, Feb. 2023.

[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015, *arXiv:1412.6980*.

[40] N. M. Punjabi, N. Shifa, G. Dorffner, S. Patil, G. Pien, and R. N. Aurora, "Computer-assisted automated scoring of polysomnograms using the somnolyzer system," *Sleep*, vol. 38, pp. 1555–1566, Oct. 2015.

[41] M. J. Sateia, "International classification of sleep disorders," *Chest*, vol. 146, no. 5, pp. 1387–1394, 2014.

[42] "Relationships between sleep stages and changes in cognitive function in older men: The MrOS sleep study," *Sleep*, vol. 38, pp. 411–421, 2015.

[43] S. Javaheri, Y. Y. Zhao, N. M. Punjabi, S. F. Quan, D. J. Gottlieb, and S. Redline, "Slow-wave sleep is associated with incident hypertension: The sleep heart health study," *Sleep*, vol. 41, 2017, Art. no. zsx179.

[44] M. Younes et al., "Reliability of the American academy of sleep medicine rules for assessing sleep depth in clinical practice," *J. Clin. Sleep Med.*, vol. 14, no. 2, pp. 205–213, 2018.

[45] R. Lopez et al., "Rapid eye movement sleep duration during the multiple sleep latency test to diagnose hypocretin-deficient narcolepsy," *Sleep*, vol. 46, 2022, Art. no. zsac247.

[46] B. D. Yetton, E. A. McDevitt, N. Cellini, C. Shelton, and S. C. Mednick, "Quantifying sleep architecture dynamics and individual differences using Big Data and Bayesian networks," *PLoS One*, vol. 13, pp. 1–27, 2018.

[47] X. Chen, H. Liu, Y. Wu, K. Xuan, T. Zhao, and Y. Sun, "Characteristics of sleep architecture in autism spectrum disorders: A meta-analysis based on polysomnographic research," *Psychiatry Res.*, vol. 296, 2021, Art. no. 113677.

[48] J. Roche, V. Gillet, F. Perret, and F. Mougin, "Obstructive sleep apnea and sleep architecture in adolescents with severe obesity: Effects of a 9-month lifestyle modification program based on regular exercise and a balanced diet," *J. Clin. Sleep Med.*, vol. 14, no. 6, pp. 967–976, 2018.

[49] K. Shahveisi, A. Jalali, M. R. Moloudi, S. Moradi, A. Maroufi, and H. Khazaie, "Sleep architecture in patients with primary snoring and obstructive sleep apnea," *Basic Clin. Neurosci.*, vol. 9, no. 2, 2018, Art. no. 147.

[50] I. A. M. Huijben, L. W. A. Hermans, A. C. Rossi, S. Overeem, M. M. van Gilst, and R. J. G. van Sloun, "Interpretation and further development of the hypnodensity representation of sleep structure," *Physiol. Meas.*, vol. 44, Jan. 2023, Art. no. 015002.

[51] R. Norman, I. Pal, C. Stewart, J. Walsleben, and D. Rapoport, "Interobserver agreement among sleep scorers from different centers in a large dataset," *Sleep*, vol. 23, pp. 901–908, Nov. 2000.

[52] X. Zhang et al., "Process and outcome for international reliability in sleep scoring," *Sleep Breathing*, vol. 19, pp. 191–195, Mar. 2015.

[53] R. Elliott, S. McKinley, P. Cistulli, and M. Fien, "Characterisation of sleep in intensive care using 24-hour polysomnography: Anobservational study," *Crit. Care*, vol. 17, Mar. 2013, Art. no. R46.

[54] B. M. Wulterkens et al., "It is all in the wrist: Wearable sleep staging in a clinical population versus reference polysomnography," *Nature Sci. Sleep*, vol. 13, pp. 885–897, Jun. 2021.

[55] E. D. Chinoy et al., "Performance of seven consumer sleep-tracking devices compared with polysomnography," *Sleep*, vol. 44, 2020, Art. no. zsaa291.

Supplementary Material for: Modeling the Impact of Inter-Rater Disagreement on Sleep Statistics using Deep Generative Learning

KOLMOGOROV-SMIRNOV METRIC

The Kolmogorov-Smirnov (KS) metric can be calculated as follows:

$$KS\left(F_1(s), F_2(s)\right) = \sup_s |F_1(s) - F_2(s)|, \tag{9}$$

where $\sup_s$ is the supremum function which returns the maximum absolute difference between the two eCDFs. Visually this can be expressed as:

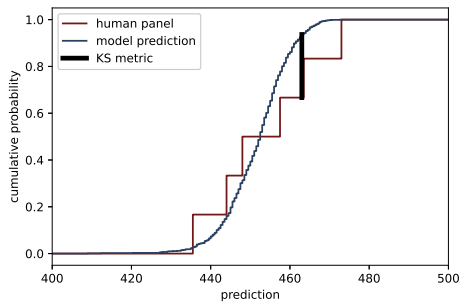

Fig. 6. Visual representation of the KS metric calculated from two eCDFs. The KS metric is calculated as the maximum distance in terms of cumulative probability between the two eCDFs.

WASSERSTEIN DISTANCE

The Wasserstein distance is calculated as the total area between the eCDFs as follows:

$$W_1\left(F_1(s), F_2(s)\right) = \int_{\mathcal{R}} |F_1(s) - F_2(s)|\ ds. \tag{10}$$

Visually this can be expressed as:

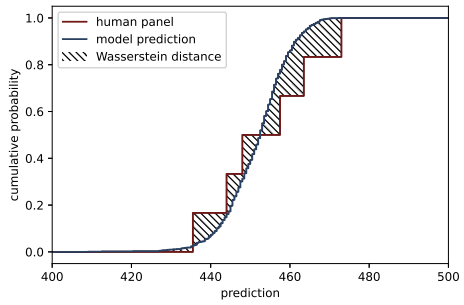

Fig. 7. Visual representation of the Wasserstein distance calculated from two eCDFs. The Wasserstein distance is calculated as the total area between the two eCDFs.

PER SUBJECT PERFORMANCE

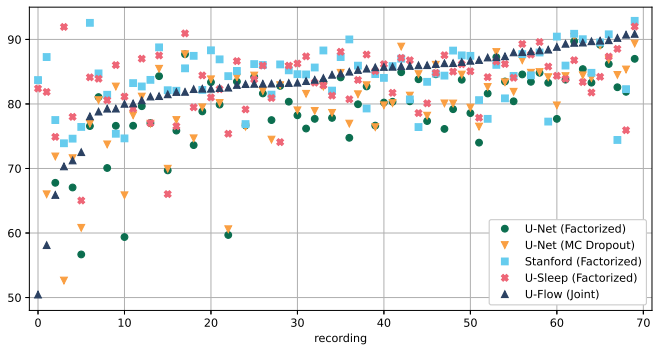

Fig. 8. Individual accuracy per recording. Note that the recordings have been sorted according to the accuracy of the proposed joint modeling method (U-Flow).

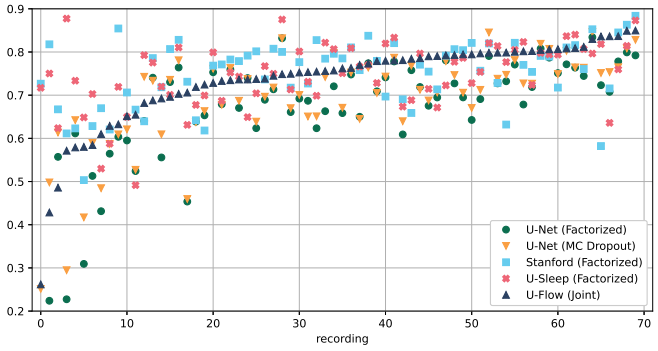

Fig. 9. Individual Cohen's kappa for per recording. Note that the recordings have been sorted according to the kappa values of the proposed joint modeling method (U-Flow). Cohen's kappa can be interpreted as: $\mathbf{0-0.10}$ no agreement, $\mathbf{0.11-0.20}$ slight agreement, $\mathbf{0.21-0.40}$ fair agreement, $\mathbf{0.41-0.60}$ moderate agreement, $\mathbf{0.61-0.80}$ substantial agreement, $\mathbf{0.81-0.99}$ near perfect agreement, $\mathbf{1}$ perfect agreement.

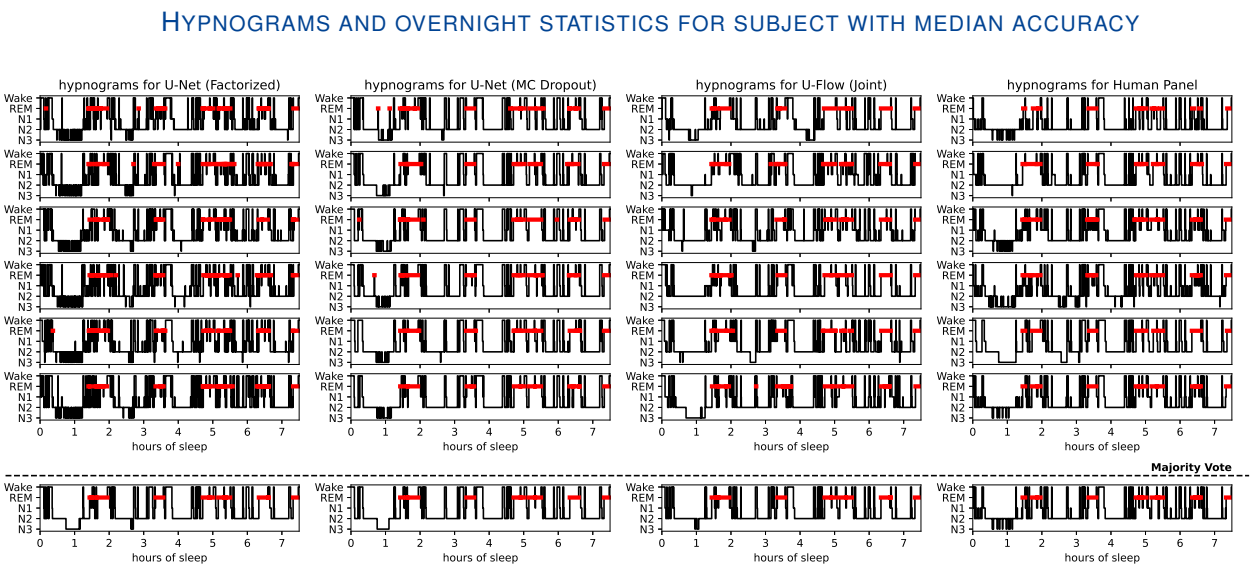

Fig. 10. Hypnograms for the recording with the median accuracy for U-Flow compared to the human panel.

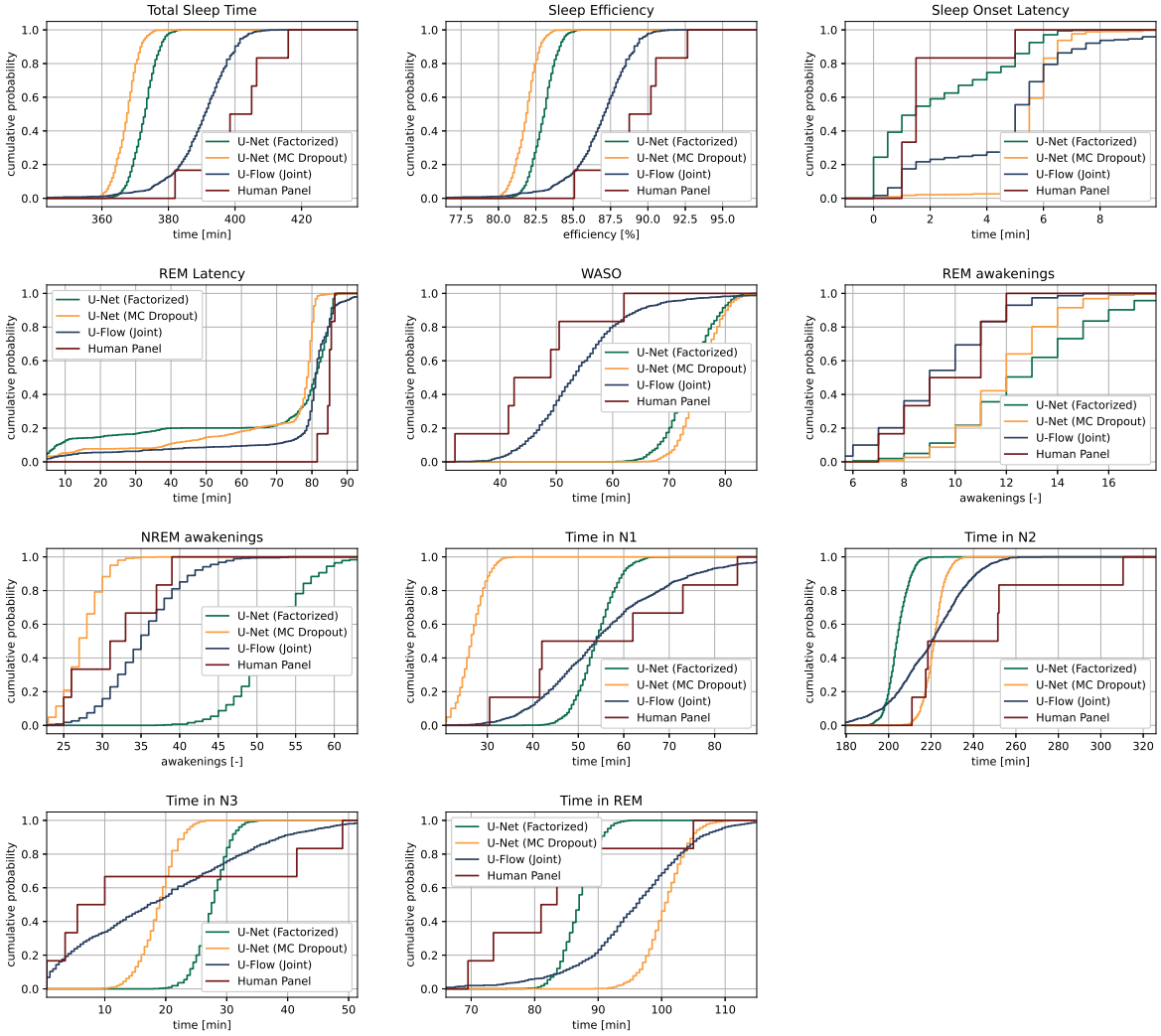

Fig. 11. Empirical distribution functions of each overnight statistic for the recording with the median accuracy for U-Flow compared to the human panel.

BUILDING BLOCKS OF THE GLOW MODEL

In this section we will elucidate all the building blocks of the 1D Glow model used by U-Flow, as detailed in Fig. 3 of the main manuscript. We will use the following notation: let $r \in [0, 8]$ denote the current resolution of the 1D Glow model, let $\boldsymbol{h}_r$ denote the current hidden vector, and let $\boldsymbol{c}_r$ denote the associated context vector which was created from the PSG measurement by the epoch and context encoder. Furthermore, the hidden vector $\boldsymbol{h}_r \in \mathcal{R}^{Ch \times L}$ is of length L with Ch being the number of channels. With slight abuse of notation, we will drop r whenever it is not currently relevant. Lastly, we will index the hidden vector as $\boldsymbol{h}[ch, l]$ to denote the element at channel index ch and lenght index l.

Flow

At each resolution level r, six steps of flow are applied. Each step of such a flow can be split into three operations: an activation normalization layer (actnorm), a squeeze layer, and a coupling layer.

Actnorm: In the actnorm layer, a scale and translate ($\boldsymbol{s}, \boldsymbol{t} \in \mathcal{R}^{Ch}$) is learned per channel, so that for each channel separately, the following forward and reverse operators are applied:

$$\tilde{\boldsymbol{h}}[ch, :] = \text{actnorm}(\boldsymbol{h}[ch, :]) = s_{ch} \cdot \boldsymbol{h}[ch, :] + t_{ch} \tag{11}$$

$$\boldsymbol{h}[ch, :] = \text{actnorm}^{-1}(\tilde{\boldsymbol{h}}[ch, :]) = \left(\tilde{\boldsymbol{h}}[ch, :] - t_{ch}\right) / s_{ch}. \tag{12}$$

The log-determinant of such an operation can be calculated in a straightforward manner as:

$$\log |\det J \text{ actnorm}| = L \cdot \sum_{ch} \log |s_{ch}|. \tag{13}$$

In other words, it is equal to summing the log of the individual scales, and multiplying that by the current length of the hidden vector. To make calculating the log-determinatnt easier, we do not learn s_{ch} directly, but rather $\log |s_{ch}|$.

Invertible Convolution: Kingma and Dhariwal introduced the invertible convolution in their original Glow paper, in order to permute the hidden vector along the channel dimension. This is useful, because the main learned component of a normalizing flow, the coupling layer, only applies a transformation to half of the channel dimensions. By applying an invertible convolution along the channel dimension, each channel gets a chance to be transformed.

The invertible convolution can be interpreted as a matrix multiplication along the channel dimension as:

$$\tilde{\boldsymbol{h}}[l, :] = \text{inv. conv.}(\boldsymbol{h}[l, :]) = W \boldsymbol{h}[l, :] \tag{14}$$

$$\boldsymbol{h}[l, :] = \text{inv. conv.}^{-1}(\tilde{\boldsymbol{h}}[l, :]) = W^{-1} \tilde{\boldsymbol{h}}[l, :], \tag{15}$$

Where $W \in \mathcal{R}^{Ch \times Ch}$ is a learned matrix. Calculating the log determinant and/or inverse of this learned matrix scales with $O(Ch^3)$. However, since the channel dimension in our 1D Glow model never grows large, it is still computationally cheap to calculate them. If computational cost really would become an issue, one can use the LU decomposition as proposed by Kingma and Dhariwal. Furthermore, during sampling, W^{-1} never changes again, so it can simply be cached in memory.

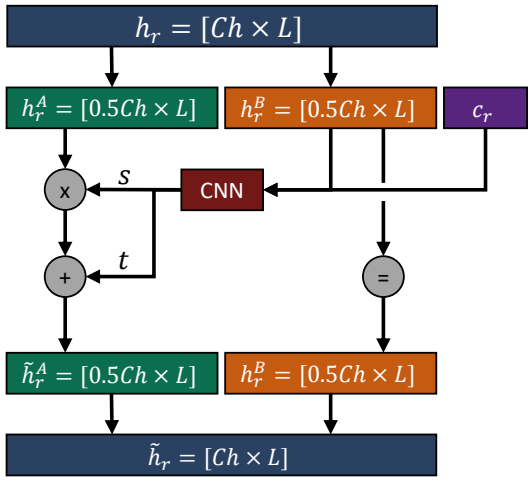

Fig. 12. Illustration of a coupling layer. The input vector gets split along the channel dimension. Part B, together with the associated context information, goes through a CNN, that outputs a scale and translate parameter. Part A gets multiplied and shifted by these. This operation is invertible, since we never have to reverse through the CNN.

Coupling layer: The coupling layer is where the Glow model really 'learns'. To learn interesting non-linear transformations, the coupling layer leverages standard deep neural network architectures, such as convolutional neural networks (CNN). However, CNNs are typically not invertible, nor is it possible to calculate the log-determinant for them. In order to circumvent this issue, the coupling layer first splits the input along the channel dimension into two equal parts, A and B. Then, part B, together with the relevant contextual information $\boldsymbol{c}_r$ is fed into such a CNN. This CNN then produces a scale and translate vector, which is applied element-wise to part A. Laslty, parts A and B are concatenated together again. Mathematically, this all can be denoted as:

$$\boldsymbol{s}, \boldsymbol{t} = \text{CNN}(\boldsymbol{h}_r^B, \boldsymbol{c}_r) \tag{16}$$

$$\tilde{\boldsymbol{h}}_r^A = \text{affine}(\boldsymbol{h}_r^A) = \boldsymbol{s} \cdot \boldsymbol{h}_r^A + \boldsymbol{t} \tag{17}$$

$$\boldsymbol{h}_r^A = \text{affine}^{-1}(\tilde{\boldsymbol{h}}_r^A) = \left(\tilde{\boldsymbol{h}}_r^A - \boldsymbol{t}\right) / \boldsymbol{s}. \tag{18}$$

$$\log |\det J \text{ affine}| = \sum_{l} \sum_{ch} \log |s[ch, l]|. \tag{19}$$

Notice how even in the reverse process, we never have to revert through the CNN, as part B and the contextual information coming from the PSG stay fixed. By then stacking many coupling layers and invertible convolutions after each other (six in our case), all parts of the hidden vector get a chance to be both part A and part B. In our case, the CNN is implemented using three convolutions of kernel size 7, where the first two are followed by activation normalization and ReLU activation. The CNN outputs a translate vector $\boldsymbol{t}$ and a scale vector $\boldsymbol{s}$. To ensure that the scaling is stable during inference, see equation (18), we want to avoid it getting close to zero. Which we ensure by applying a sigmoid activation function on it, and then adding a small amount of bias (0.01 in our case). See Fig.12 for an illustration.

Split

The split and squeeze operation together form the normalizing flow analog of the subsampling operation. We cannot use a standard subsampling operation, since the normalizing flow architecture has to be invertible, and the upsampling operation has to be the exact inverse.

In the split operation, similar to what is done in the coupling layer, the current hidden vector is split along the channel dimension into a part A and a part B. Then, part A is split of from the rest of the model, and already appended to the latent vector $\boldsymbol{z}$, while part B is carried forward and goes through the rest of the model. The hidden vector is thus reduced in size by a factor 2, ensuring that we can use very deep normalizing flow architectures, without computational complexity growing out of hand. During the reverse process, we simply take part A out of the latent vector $\boldsymbol{z}$, and concatenate it again with B.

The intuition behind the split operation is that the hidden vector $\boldsymbol{h}$ contains both complex data structures, which are difficult to normalize and cast into Gaussians, and simple data structures, which are easy to normalize. The Flow blocks that precede the splitting operation should thus align the elements of the hidden vector in such a way as to put the already normalized information into part A, which are already put into the latent vector $\boldsymbol{z}$, while the rest is put in part B, to be handeld by the deeper layers.

Squeeze

Whereas the splitting operation takes care of reducing the number of elements in the hidden vector, the squeezing operation takes care of reducing the spatial resolution of the hidden vector. This is analog to maxpooling or other subsampling operations in a CNN, such that the deeper layers of the model have a larger field of view.

In a 1D squeeze layer, the elements along the length axis L are indexed, all the odd-numbered indices are left in place, while all the even-numbered indices are taken out, and concatenated channel-wise on top of the odd indices, see Fig.13 for a visual illustration of this process. In other words, the squeeze layer trades of resolution along the length axis for resolution along the channel axis.

By now applying the 1d splitting and squeezing operation together, we effectively reduce the number of elements of the hidden vector by a factor 2 and decrease the resolution along the length dimension by a factor 2. They thus forms the normalizing flow analog of a pooling operation.

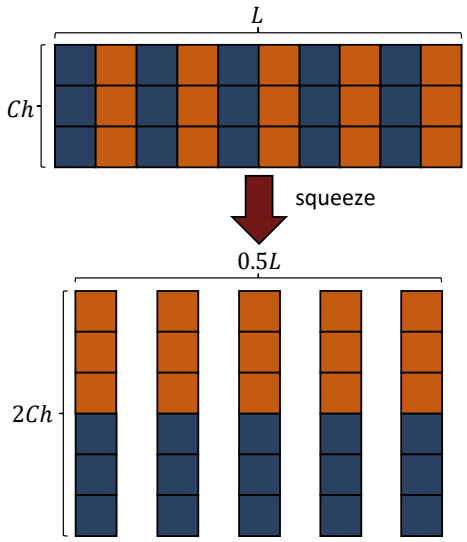

Fig. 13. Illustration of a squeeze layer, where spatial resolution gets cut in half by doubling the channel resolution.