Comparison of Sleep State Classification Performance Using Random Forests, Hidden Markov Models, and Non-homogeneous Hidden Markov Models

by

Mathieu Chalifour

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Statistics

Department of Mathematical and Statistical Sciences
University of Alberta

Abstract

In this work, the CF00N polysomnograph data of 75 patients, with ranging severeties of Obstructive Sleep Apnea (OSA), is presented and analyzed in terms of sleep state classification. The pre-processing and cleaning of each polysomnograph recording were performed in `R` (R Core Team, 2019) using independent component analysis. Then three sets (Renyi Entropy, Moments of Discrete Wavelet coefficient (DWC), Moments of Non-EEG signals) of statistical features, 40 in total, were extracted from the time, frequency, and time-frequency domain of every 30 second epoch using the C4M1 and non-EEG channels. A random forest feature selection was then performed selecting nine multivariate normal features to be used for comparison of classification performance against all three feature sets. Classification performance of sleep states for each patient's epochs was analyzed first using random forests with a 10-fold cross validation and then a leave-one-patient-out-cross-validation (LOPOCV). In the 10-fold cross validation, the mean (standard deviation) accuracy of the four feature sets was found to be 73.34%(0.07), for Renyi features, 75.26%(0.07), for DWC features, 60.64%(0.1), for non-EEG features, and 76.85%(0.06) for the final features. In the LOPOCV, the mean performance measures of the random forest was found to decrease and the variance increase each feature set when the testing and training data did not share epochs from the same patient. The mean accuracy results for the LOPOCV were 67.1%(7), for Renyi features, 74.5%(5.3), for DWC, 33.6%(6.3), for non-EEG features, and 72.2%(6.4) for the final features. The classification performance in the LOPOCV was further analyzed using a 2-way MANOVA, which found no significant difference between the

means of classification performance measures for the patient age and OSA group combinations.

Then, using the Renyi entropy and final features of each patient's epochs, hidden Markov models (HMMs) and non-homogeneous hidden Markov models (NHMMs) were fitted using 500 random starting points. The HMMs and NHMMs were fitted via the `R` library `depmixs4` (Visser & Speekenbrink, 2010). The mean classification accuracy using the Renyi features was 67.1%(7.9) for HMM and 68.9%(7.8) for the NHMM and, using the final features, 65.3%(8.7) for HMM and 67.6%(9.1) for NHMM. Again, 2-way MANOVA was employed, with the only significant difference found between the mean performance measures of the age and OSA groups for the NHMM that used the final features. Furthermore, the comparison between HMMs and NHMMs that used the Renyi features found that on average the NHMM accuracy was between 0.5% and 3.1% higher than HMM. When the HMMs and NHMMs used the final features, the NHMM accuracy was on average between 0.4% and 4.2% higher than HMM.

A comparison of classification accuracy for the random forest LOPOCV versus the HMM and NHMM found that, when using the Renyi features, the HMM and NHMM typically performed better than the random forest and, when using the final features the random forest performed better than the HMM and NHMM. The analysis of this thesis demonstrates that although the random forest, HMM and NHMM can be successful at classifying sleep states, the HMM and NHMM are superior, since the random forest lacks a model for the dependence structure between sleep states. The modelling of sleep state transitions captured by the HMM and NHMM can provide sleep experts with further insight to the underlying dynamics of sleep.

Preface

This thesis is an original work by Mathieu Chalifour. The clinical study, of which the CF00N data presented and analyzed in this thesis is a part, received ethics approval by the Health Research Ethics Board of the University of Alberta Pro00057638.

For my grandfather, Roland, without whose encouragement and advice this accomplishment would never have been possible.

For my son, Alaxandre, and sister, Shellaine, may you both one day reap the fruits of your ambition and hard work.

Acknowledgments

I would first like to thank my master's thesis supervisor, Dr. Giseon Heo, for her guidance and support. Furthermore, for her giving me the opportunity to be a graduate student at the University of Alberta. Next, I would like to thank my master's thesis co-supervisor, Dr. Adam Kashlak for his advice and help with this thesis. I would like to give special thanks to my father, Wade Douglas Chalifour, for all of his worldly wisdom, and my mother, Myrtle Cindy Shank, for her help editing the first draft of this thesis. Further thanks for editing go to my friend, Marshall Robert Michael Hann, whose grammatical editing skills are rivaled only by the rigour of a pure mathematician. However, any faults found in this thesis are my own and do not reflect at all on their contributions and aid. Lastly, my deepest gratitude goes to my life partner, Sarah Fecko. I could not have accomplished this feat without your encouragement, inspiration, and most of all your support.

Contents

List of Tables

List of Figures

Chapter 1

Sleep and Polysomnography

1.1 Introduction to Sleep

Long before the scientific study of sleep, there has always been a curiosity about the nature and functional purpose of sleep. According to Chokroverty (2017), people have always understood sleep is an essential part of life in the same way as death. Only beginning in the twentieth century have researchers begun to quench this curiosity with scientific evidence. This evidence has allowed researchers to define characteristics of healthy sleep and sleep related diseases. Evidence found by examining how changes in the physiological processes of sleep affect a patient's physical and mental health led researchers to find answers to the important question of what characterizes healthy sleep. Understanding the process of healthy sleep allows experts to identify irregularities in patient sleep architecture, diagnose sleep related disorders, and determine a course for treatment. Sleep architecture is defined in terms of two main components: the macro and the micro-structures. The macro-structures govern and define the main components of sleep. The micro-structures are events within certain sleep macro-structures.

1.2 Sleep macro-structures

There are six main sleep macro-structures: circadian rhythm, sleep type or state, sleep cycles, sleep latency, sleep efficiency, and wake after sleep onset. Circadian rhythm is the periodic cycle of sleep and wake periods throughout the day, commonly known as the body clock. Typically, sleep is done at night in response to external stimuli that help to develop the circadian rhythm as a person ages, like the sun going down. Table 1.1 shows the duration of sleep as a person ages, which decreases from birth to adolescence.

There are three main types of sleep: wakefulness, rapid eye-movement (REM), and non rapid eye-movement (NREM). NREM sleep accounts for about 75-85% of total sleep and REM the other 15-25%. However, Figure 1.A shows how the proportions of NREM and REM change as a person ages. By the age 12-13 years a person's REM and NREM sleep patterns have almost fully matured and the proportions of REM and NREM stay fairly consistent for the rest of the lifespan. According to American Academy of Sleep Medicine (AASM) standards (Iber et al., 2007), the NREM sleep is comprised of 3 distinct states: NREM 1 and NREM 2 are considered very light sleep and are thought of as transitioning states from wakefulness to the deep sleep, NREM 3. Wake and REM sleep are considered their own states, with REM being considered the active sleep state. The AASM standards (Iber et al., 2007) for scoring sleep states have been used in this research. To summarize, there are five distinct sleep states used for this research: wake, NREM 1, NREM 2, NREM 3 and REM. Prior to 2007, manual scoring of sleep states was done in accordance with the (Rechtschaffen & Kales, 1968) (R&K) guidelines, which splits the NREM 3 state into NREM 3 and NREM 4.

Sleep cycles are the cyclical pattern of the wakefulness, NREM, and REM sleep through out the night. Cycle duration increases as a person ages, starting around 45-60 minutes for infants and toddlers, then gradually increasing to 60-80 minutes for children. Sleep cycle duration, seen in Table 1.1, begins to reach its maximum of 90-120 minutes around thirteen years of age. The graphical representation of

Age	Sleep Duration	Number Daily Naps	Sleep Cycle Duration	Number of Sleep Cycles
Newborn	16-18 Hrs	4-6	45-50 mins	20-24
6 months	14.2 Hrs	3-5	45-50 mins	19
1 Yrs	13.9 Hrs	3-4	45-50 mins	19
2 Yrs	13.2 Hrs	1-2	50-60 mins	13-16
4 Yrs	11.8 Hrs	0	50-60 mins	12-14
5 Yrs	11.4	0	60-70 mins	10-11
12 Yrs	9.3	0	80-100 mins	6-7
13-50 Yrs	8 Hrs	0	90-120 mins	4-5
50+ Yrs	7.5 Hrs	1-2(60+ Yrs)	90-120 mins	4-5

Table 1.1: Changes to human sleep cycle over lifespan

sleep state progression, Figure 1.B, is called a hypnogram, which is an important tool used by sleep experts for examining irregularities in sleep cycles. Sleep latency is the amount of time needed to fall asleep. Sleep efficiency is the percent of time spent sleeping in relation the total amount of time spent in bed. Lastly, wake after sleep onset is the amount of time spent in the wake sleep state after falling asleep for the night.

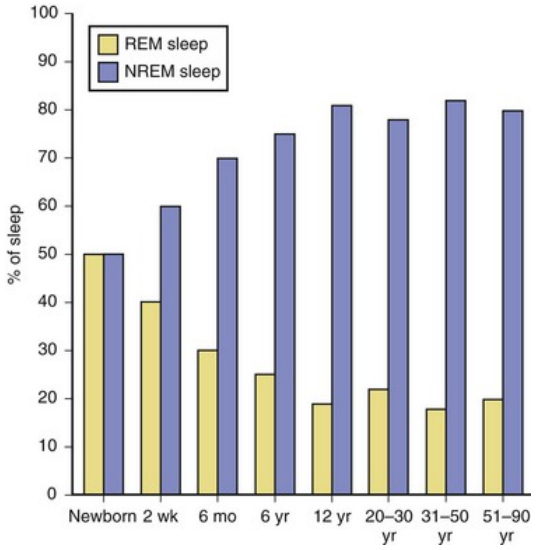


Figure 1.A: Proportions of REM vs NREM sleep across human lifespan. Image taken from Chokroverty (2017)

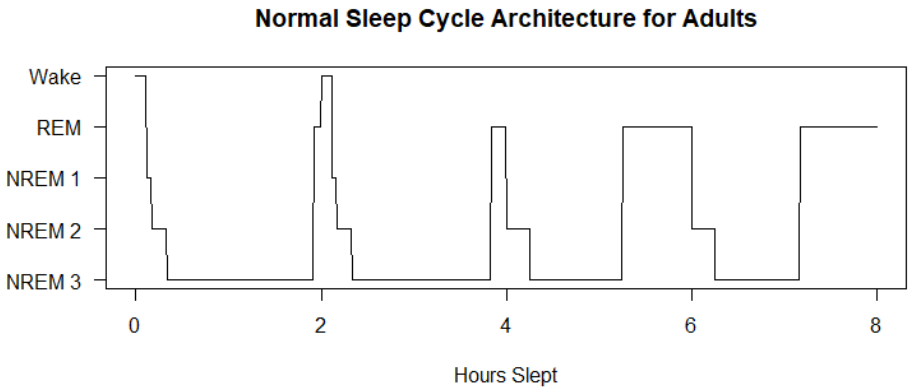


Figure 1.B: Hypnogram of normal sleep cycle for adults. Cycle duration approximately 120 minutes.

1.2.1 Sleep micro-structures

There are four primary sleep micro-structures: cyclical alternating pattern (CAP), arousals, sleep spindles, and K-complexes. The CAP is an electroencephalogram (brainwave) pattern that repeats in a cyclical manner, seen in Figure 1.C, typically during NREM sleep. Presence of a CAP is indicative of sleep instability. The pat-

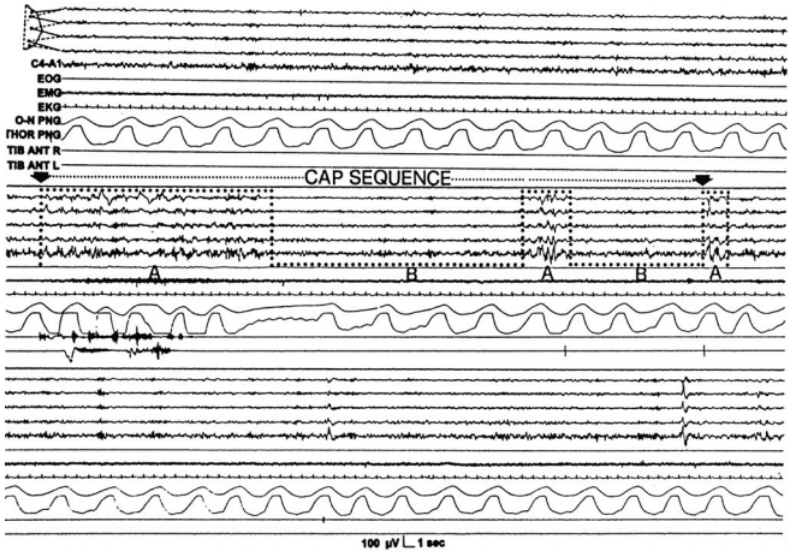


Figure 1.C: Example of polysomnograph data with presence of cyclic alternating . CAP sequence indicated by dotted line. Image taken from Terzano et al. (2001)

tern consists of a stable A phase, consisting of regular undisturbed sleep, and an unstable B phase. The B phase is typically accompanied with heart rate, blood

pressure, muscle tension, and respiratory activity all increased. An arousal "is an abrupt shift in [brain wave] frequency lasting from 3 seconds to 14 seconds" (Chokroverty, 2017, p. 10) without becoming fully awake. Patients must also be sleeping for at least 10 seconds prior to arousal. Sleep spindles and K-complexes are micro-structures specific to NREM 2 sleep. Sleep spindles "represent periods of time where the brain inhibits mental processing in order to keep the person in a tranquil state. By keeping the person in a tranquil state, the sleep cycle can continue and the person can transition to the next stage of deep sleep" (Cushner, Fish, & Wilson, 2019). K-complexes are produced in response to external stimuli while a person sleeps that work to suppress external stimuli to keep a person sleeping.

1.3 Sleep Polysomnography

Examining the physiological processes that comprise sleep macro-structures is done using sleep polysomnography (PSG). Chokroverty (2017, p. 6) describes in brief detail the development of PSG in sleep research and how it captured the interests of the scientific community. PSG was first used in 1875 by Richard Caton, an English physician who discovered Electroencephalogram (EEG) waves in dogs. Over five decades later, in 1929, Hans Berger, a German physician, discovered the alpha frequency band of EEG waves in humans. In 1937, American physiologists Alfred Lee Loomis, Edmund Newton Harvey, and Garret Hobart discovered there are different sleep states by examining the changes in EEG wave frequencies during sleep. Then, in 1953 at the University of Chicago, Eugene Aserinsky and Nathaniel Kleitman discovered Rapid Eye-Movement (REM) sleep. This discovery "electrified the scientific community and propelled sleep research to the forefront" (Chokroverty, 2017, p. 6).

Technological improvements of the last few decades have drastically improved sleep PSG, as it no longer only measures EEG wave forms. The main electrophysiological measurements taken during sleep PSG are the electrooculography (EOG), electromyography (EMG), and electrocardiography (ECG), which mea-

sures eye-movement, skeletal muscle tension, and heart activity, respectively. However, patients recommended for sleep PSG due to a sleep related breathing disorder will have additional measurements for respiratory activity and blood-oxygen saturation. The importance of sleep PSG in medicine cannot be overstated, as it is considered the gold standard for diagnosing sleep related breathing disorders. PSG is typically performed overnight in a laboratory setting with electrodes attached on the surface of the patient's skin prior to bedtime. The electrodes measure electrical impulses using a site electrode and a reference electrode, specifically the difference in electrical potential (μV) between the site and reference electrodes. EEG electrodes are placed on the scalp according to the International 10-20 system, seen in Figure 1.D, which is used to measure electrical activity of neurons in the brain. The reference electrodes are labelled A1 and A2, but are interchangeably referred to as M1 and M2, since they are placed on the mastoid. PSG channels are named using the site and reference electrodes, C4M1 refers to the PSG channel that measures the difference in electrical potential between electrodes C4 and M1 (A1 in Figure 1.D).

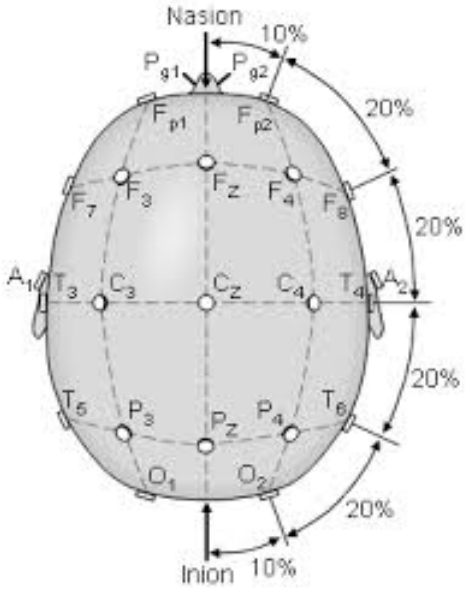


Figure 1.D: International 10-20 EEG electrode placements. Image taken from AASM manual (Iber et al., 2007)

EOG site electrodes are placed on the peripheral sides of the eyes and use the same reference electrodes as EEG. The EMG site electrodes are placed on the left

and right side just underneath the chin with the reference placed slightly above the chin. Figure 1.E shows the placement of EOG and EMG electrodes. Additional EMG electrodes can also be placed on the abdomen and legs in order to measure skeletal muscle activity tension in the lower half of the body. The ECG electrodes are attached to the patient abdomen in the chest area.

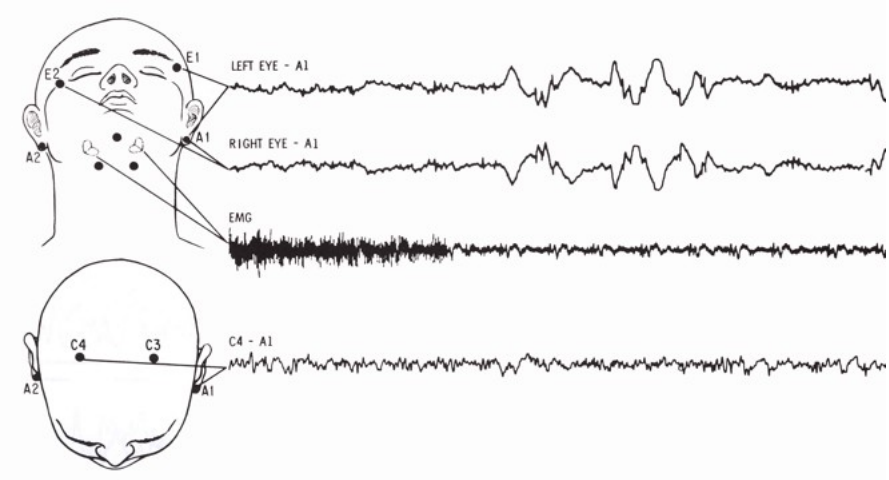


Figure 1.E: EMG and EOG electrode placement. Image from (Rechtschaffen & Kales, 1968)

Once complete, the PSG recording is then segmented into 30 second epochs and each epoch manually scored by the expert into one of the discrete sleep states, in accordance with AASM scoring criteria. Scoring of the epoch into one of the distinct sleep states is primarily done by identifying the dominant EEG wave frequency. Experts also use EMG, ECG, and EOG activity present to score the epoch. For example, in a REM state where it is believed the majority of active dreaming occurs, the EOG can detect the rapid movement of the eyes. During REM sleep the body enters a temporary paralysis to keep the dreamer from physically acting out. Furthermore, this paralysis is reflected in the EMG, as the skeletal muscle tension is drastically reduced. The author refers the interested reader to the AASM manual (Iber et al., 2007) for an in-depth explanation of the current medical sleep state scoring standards.

1.3.1 Goal and Overview of this Thesis

The goal of this thesis is to propose an automated solution for scoring the sleep states of individual patients. There are several approaches or classifiers proposed.

1. Random Forest classifier is trained using all but one patient data and then classifies the sleep states of the remaining patient.
2. Hidden Markov Models (HMMs) and Non-homogeneous Hidden Markov Models (NHMMs). Cluster and then classify sleep states of individual patients using only data from that individual.

The rest of this thesis is organized as follows. Chapter 2 reviews previous studies that propose an automated solution for sleep state classification. Chapter 3 begins with a brief introduction to the random forest methodology used in the literature discussed in Chapter 2. Then an introduction to HMMs and their mathematical framework. Along with estimation of model parameters with the Expectation-Maximization variant specific to HMMs and the decoding of the sleep states for individual patients using the Viterbi algorithm (Viterbi, 1967; Forney, 1973). The same is then done for the NHMM along with discussing the methodology used in the `depmixS4` (Visser & Speekenbrink, 2010) to estimate model parameters and the decoding of sleep states. Chapter 4 discusses the PSG data that was collected at the University of Alberta sleep laboratory and used for the research presented in this thesis. Then moves into the pre-processing, cleaning, artifact removal strategies, and the extraction of statistical features from the epochs of each individual patient's PSG. Chapter 5 is the feature selection process carried out using the Random Forest methodology, specifically using the `randomForest` (Liaw & Wiener, 2002) library in `R`. Chapter 6 provides the classification analysis of the random forest approach used in this thesis and compares classification performance with the respective literary sources. The analysis goes slightly deeper than what was done in the literature as the number of trees used in increased significantly and the cross-validation setup is further extended to model the classification of sleep states for individual patients. Chapter 7 is the HMM and NHMM classification analysis.

This is done for both models separately and then a comparison between them is conducted. The last part of the chapter explores the sleep dynamics between sleep states of individual patients by examining the transition probability matrix. Chapter 8 is the final chapter of this thesis, it provides the conclusion and a brief comparison of classification performance between HMM and NHMM versus random forest. Lastly, a discussion of possible improvements and future work

Chapter 2

Literature Review of Sleep State Classification

2.1 Sleep State Classification

2.1.1 Automated Sleep State Classification

Sleep state scoring is done effectively by the experts, yet it can also be a long and tedious process that's subject to human error. This leaves room for automated improvements to reduce scoring time and eliminate scorer subjectivity such as accurate automated sleep state classification algorithms. However, automated sleep state classification is not as simple as feeding the raw PSG signal to a machine to classify sleep states of the epochs, not yet anyway.

When dealing with any data, the first step is to clean and organize it for processing. The PSG data is cleaned via a digital filter, along with removal of interference artifacts that may also contaminate the PSG signals, both of which will be discussed in detail in chapter 4, then segmentation of the clean PSG data into 30 second epochs is performed. For the next step, there are two main approaches. The first being extraction of statistical features from each epoch of PSG data. This in-

cludes time domain features, such as statistical moments of the raw signals. There are also frequency domain features, such as spectral analysis to find parameters that characterize what frequencies dominate the epoch. Extraction of statistical features from the epochs is a very natural starting point, as sleep scoring done by experts uses the dominant wave frequency and micro-structure events present in EEG and other PSG signals.

The second and more modern approach is to transform or decompose the PSG data into the time-frequency domain and then extract statistical features. Time-frequency domain features are of significant interest, as they open the doors for researchers to consider many other feature types. The continuous wavelet transformation and entropy measures are very popular choices (Fraiwan et al., 2010, 2012; Boostani et al., 2017) and will also be discussed in chapter 4. Once a set of statistical features have been extracted from each epoch, the next major step is to choose an automated statistical classification algorithm.

2.1.2 Importance of Accurate and Reliable Results

The importance of accuracy for an automated sleep scoring algorithm cannot be overstated, especially for patients, since patients who are expected to have abnormal sleeping patterns and architecture may not be properly diagnosed and treated. Comparison of sleep state classification performance in the literature will be done using the overall classification accuracy and Cohen's kappa, κ statistic (Cohen, 1960), as both are commonly utilized. The classification accuracy will be measured with respect to the 5-state AASM or 6-state R&K scoring system used by the sleep experts to classify the epochs. κ is considered a more robust measure of rater agreement for nominal variables that corrects for pairwise agreements happening by random chance. This statistic is commonly preferred over the accuracy for evaluation of agreement between the classification of two or more field experts and for the evaluation of classification algorithm performance. Landis and Koch (1977) provide the first interpretations of the κ statistic, Table 2.1, found in the literature.

κ Statistic	Strength of Agreement
< 0	Poor
0 - 0.20	Slight
0.21 - 0.40	Fair
0.41 - 0.60	Moderate
0.61 - 0.8	Substantial
0.81 - 1.00	Almost Perfect

Table 2.1: Table of Cohen's κ interpretation taken from (Landis & Koch, 1977).

However, Landis and Koch (1977, p. 165) also state that "[although] these divisions are clearly arbitrary, they do provide useful 'benchmarks' for the discussion" of their specific example of comparing patient diagnosis of multiple sclerosis by two Neurologists. These benchmarks have been accepted and used in many fields to measure classification performance and agreement evaluation. A more modern measure of classification performance produced from a clustering algorithms is the adjusted Rand index (ARI) of Steinley (2004a). This measure will be used to assess the classification performance of the HMM and NHMM presented in this thesis. To the knowledge of the author, the ARI has not been found in any sleep state classification literature thus far, so for that reason a brief description of the ARI is presented in Chapter 7. For a detailed explanation the author refers the interested reader to Steinley (2004a) and McNicholas (2017).

Danker-Hopfe et al. (2008) examine the inter-rater reliability(IRR) using R&K and AASM guidelines using seven scorers from sleep labs in Austria and Germany, each with long standing credibility in sleep science. The scorers analyzed 72 PSG recordings from 56 healthy subjects and 16 patients with varying sleep disorders. Danker-Hopfe et al. (2008) reported that the IRR agreement using AASM sleep state scoring is 82.0%, which is slightly higher than the 80.6% for the R&K scoring system. "The κ statistics, $\kappa = 0.76$ for AASM and $\kappa = 0.68$ for R&K, show an increase in agreement between raters from R&K to AASM standards" (Danker-Hopfe et al., 2008, Abstract).

Stepnowsky et al. (2013) had three experts analyze and score the sleep states of 44 (adults, 22-69 years) PSG recordings. The first group of twenty-one PSG

recording were scored using R&K criteria. This group contained 6 control subjects and 15 patients with varying severity of obstructive sleep apnea (OSA). The second group contained PSG recordings of twenty-three patients with mild to severe OSA. The PSGs in the second group were scored using AASM criteria. Stepnowsky et al. (2013, p. 3) found that in the first group 99.9% of epochs scored had at least two experts agree on the sleep state classification. In the second group 97.1%, and 98.4% overall. However, the percentage of epochs where all three sleep experts agreed was lower in both groups, as seen in Figure 2.A. The first group had just over 80% and the second group under 60%. The disagreement between experts, seen in Table 2.2, is also reflected in the κ statistics. All three raters agree quite strongly for the first data set, but the κ statistics between rater 3 and the other two raters drop significantly for the second patient data set.

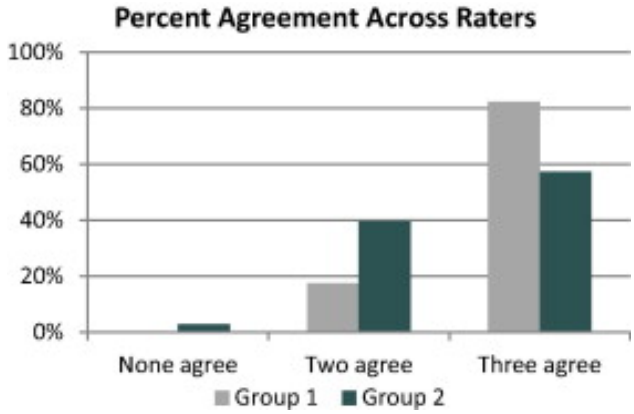


Figure 2.A: Inter-rater agreement from Stepnowsky et al. (2013). "Percentage of epochs with no agreement, majority agreement (two agree), and consensus agreement (all three agree)" (Stepnowsky et al., 2013)

	Raters 1 vs 2	Raters 1 vs 3	Raters 2 vs 3
Group 1	0.85	0.89	0.77
Group 2	0.80	0.46	0.49
Overall	0.83	0.68	0.64

Table 2.2: κ agreement between experts. Table taken from Stepnowsky et al. (2013).

The overall agreement across all subjects was only 72.6%, $\kappa = 0.72$ and the "[differences] in agreement were observed based on raters, obstructive sleep apnea (OSA) severity, medications, and signal quality" (Stepnowsky et al., 2013, Abstract). This indicates that sleep related disorders, specifically OSA, can negatively impact the scoring of sleep states from one expert to the next. The study by Boostani et al. (2017) also emphasizes the importance of the sleep state classification performance of patients versus healthy subjects. Their analysis compares the performance of five different statistical feature extraction methods from Acharya, Chua, Chua, Min, and Tamura (2010); Fraiwan et al. (2010, 2012); Güneş, Polat, and Şebnem Yosunkaya (2010); Weiss, Clemens, Bódizs, Vágó, and Halász (2009) across four different classification algorithms: random forest, Gaussian mixture models, K-nearest neighbours, and linear discriminant analysis. Each sleep state classification method in the analysis is performed on a healthy subject data set and a patient data set using a leave-one-patient-out-cross-validation (LOPOCV). They used the publicly available *Sleep-EDF 2002* and *CAP* (Cyclical Alternating Pattern) data from *PhysioNet* (Goldberger et al., 2000). The *Sleep-EDF 2002* data set consists of 20 healthy adults subjects (25-34 years) who did not require medication to sleep. The *CAP* data set contains PSG recordings for 20 patients with REM behaviour disorder. Boostani et al. (2017) concluded that the overall sleep state classification accuracy for each method is lower for the patient group across all feature extraction and classifier combinations. The best classification performance used the statistical features of Fraiwan et al. (2012) with the random forest classifier, which provided a mean accuracy of 87.06% for healthy subjects and only 69.05% accuracy for patients.

2.2 Automated Classification Methods

2.2.1 Multivariate Classification Methods

The typical starting point for sleep state classification begins with an applied multivariate analysis. The goal is to extract useful statistical features from each 30

second epoch of a PSG recording using the first or second approach described in Section 2.1.1, then combine all epochs into a single data set where the 30 second epochs are treated as independent observations and the extracted features are treated as the covariates. A multivariate classification algorithm is then implemented using the expert's sleep state scores as the true labels. Šušmáková and Krakovská (2008) examined the classification accuracy of 73 different statistical features extracted from EEG, EOG, EMG, and ECG channels, for 20 healthy subjects. Fisher discriminant analysis was used to select the best individual features for classification of epochs into the five distinct states. Šušmáková and Krakovská (2008) reported the best single-discrimination feature obtained 57.4% accuracy across all AASM sleep states. The mean classification accuracy of all features was 53.28% across all AASM sleep states, which suggests that in order to successfully classify sleep state within a reasonable amount of error, more than one feature must be used.

Acharya et al. (2010) considered two cohort data sets in their research. The first was a group of 25 adults with mild to severe OSA and the second was 14 healthy subjects. According to Acharya et al. (2010), they extracted higher order spectra features from bi-spectrum and bi-coherence plots for the various sleep stages from all epochs. Analysis of variance (ANOVA) was used to select optimal features for use in a Gaussian mixture model. Overall, Acharya et al. (2010) achieved 88.7% accuracy for the six-state R&K classification. Güneş et al. (2010, p. 7923) introduced a "data pre-processing technique, [a] K-Means clustering based feature weighting to increase the classification ability of sleep stages using [K-Nearest Neighbours] classifier and decision tree classifier" on 5 adult male patients, which obtained 82.5% classification accuracy.

Fraiwan et al. (2010) used the continuous wavelet transformation of a single EEG signal and compares three different mother wavelets: Daubechies order 20, reverse bio-orthogonal, and Gaussian of order 1. Entropy of the wavelet coefficients in seven frequency sub-bands was used as statistical features. In total, 21 features were calculated for each epoch. There were 32 subjects in the study for a total of 41,778 epochs. Linear discriminant analysis was the chosen classifier reaching

a mean classification accuracy of 84% and κ = 0.78 using 10-fold cross validation. The successive study by Fraiwan et al. (2012) compared three different time-frequency methods for EEG signal analysis: continuous wavelet transformation, Hilbert-Huang Transformation, and Choi-Williams distribution. Renyi entropy of the same sub-bands as those in Fraiwan et al. (2010) were then calculated as the statistical features. The data consists of 20,269 epochs from 16 PSG recordings from the *Sleep-EDF 2002* data set. A random forest, or rather a bootstrap aggregation of 10 decision trees, with each tree over fit on a bootstrap sample of training data, was implemented with classification accuracy of 83% and $\kappa = 0.76$ for the continuous wavelet transformation features. This methodology will be discussed further in chapter 3.

The study by da Silveira et al. (2017) used an approach adapted from Fraiwan et al. (2012). They decomposed the EEG signal into 6 frequency bands using the discrete wavelet transform with a Daubechies wavelet of order 2. The variance, skewness, and kurtosis of the wavelet coefficients in each frequency band were extracted as features. The study used the *Sleep-EDF Expanded* data set, which contains PSG recordings of 20 healthy adults (10 male, 10 female) for a two night sleep study with a total of 106,376 epochs scored with R&K standards. "Two PSGs of about 20 hours each were recorded during two subsequent day-night periods at the subjects homes" (Goldberger et al., 2000). da Silveira et al. (2017) employed the same setup for random forest as Fraiwan et al. (2012), but increase the number of trees from 10 to 64. A 10-fold cross validation over 100 trials was performed, which achieved a mean accuracy of 90.5% and κ=0.8 agreement with R&K standards. da Silveira et al. (2017) also analyzed the data using a (1/3) testing, (2/3) training split of the data for comparison with Fraiwan et al. (2012) and reported 90.2% accuracy for R&K classification.

The study by Koley and Dey (2012) examined PSG recordings for patients that had current medical conditions, such as high blood pressure and diabetes. Out of 28 patients, 13 had a confirmed diagnosis of OSA while the remaining 15 did not. The data was then randomly split into testing (12 subjects: 5 OSA, 7 No OSA)

and training (16 subjects: 8 OSA, 8 No OSA) sets. They creatively used a support vector machine (SVM) recursively for selecting a subset of optimal features from a total 39 calculated. The classification scheme they employed used five parallel binary SVM (one SVM for each state). A one state against all other states strategy was used in each SVM to determine the class label for each epoch. A cross validation of the training data was performed to train model parameters. Model validation was performed using the test data, which obtained an accuracy of 89.91% and κ = 0.868 for non-OSA subjects and 88.86% accuracy and κ = 0.846 for OSA subjects. This analysis provides evidence that classification performance is not always significantly lower for epochs of patients with a sleep related breathing disorder, especially when the patient epochs are compared to healthy subject epochs in the same sleep study, which means all PSG recordings are done using the same equipment settings and have the same rater scoring the epochs.

Hassan and Subasi (2017) decompose 8 healthy adult EEG recordings from the *Sleep-EDF 2002*. They also used 20 (16 female, 4 male) healthy adult EEG recordings from the *DREAMS* (2016) data set. The recordings were segmented into epochs, which are then decomposed into distinct sub-bands using the tune able Q-wavelet transform. The first four statistical moments in each sub-band were extracted as features. They used a Kruskal-Wallis one-way analysis of variance to determine which individual features were useful for discriminating between sleep states. Each data set was then split evenly into mutually exclusive testing and training sets. Classification was performed using bootstrap aggregating of decision trees, like that of Fraiwan et al. (2012). Results for the *Sleep-EDF 2002* reached 93.69% accuracy and κ=0.8543 for AASM classification. The results for the *DREAMS* (2016) data was found to be 78.5% accuracy and κ = 0.82 for AASM classification.

Another study by Seifpour, Niknazar, Mikaeili, and Nasrabadi (2018) decomposed a single EEG channel from the same data as Hassan and Subasi (2017) into six sub-bands using a band-pass filter. The statistical behaviour of the local extrema in each epoch was extracted and used as the statistical features. A multi-class SVM was the chosen classifier, which achieved 91.8% accuracy and κ = 0.87 agreement

with AASM state classification of the *Sleep-EDF 2002* recordings. Seifpour et al. (2018) found the classification performance on the *DREAMS* (2016) data set was 83.3% and κ= 0.77 for AASM state classification.

2.2.2 Neural Network Methods

There are many different variations of neural networks that have been successfully applied to sleep state classification. The study by Özşen (2012) examined PSG recordings of 5 healthy subjects using five artificial neural networks (ANN), one for each sleep state, to classify the epochs. A validation data set was created using untouched epochs from 3 of the 5 subjects and reported 90.93% accuracy for AASM state classification. A recurrent neural network (RNN) approach was explored by Hsu, Yang, Wang, and Hsu (2013) using 8 PSG recordings from the *Sleep-EDF 2002* data. Performance of their proposed method was compared to a feed-forward neural network (FNN) and a probabilistic neural network (PNN). The RNN outperformed both of its competitors. The RNN achieved 87.2% accuracy compared to 81.1% for FNN and 81.8% for PNN.

Complex-valued neural networks (CVANN) have been successfully implemented by Peker (2016a, 2016b). Peker (2016a) examined 8 PSG recordings from the *Sleep-EDF 2002* data set. Using the raw EEG signal, nine non-linear features were extracted and then "converted into a complex-valued number using a phase encoding method" (Peker, 2016a, Abstract). A LOPOCV was used to evaluate classification performance. Peker (2016a) obtained 93.84% accuracy and κ = 0.919 agreement with AASM standards and 91.57% accuracy and κ = 0.892 agreement with R&K standards. Peker (2016b) used a dual-tree complex wavelet decomposition on the EEG signals of 25 adults with mild to severe sleep apnea. Five basic statistical measures were extracted from the wavelet coefficients in each epoch (min, max, mean, standard deviation, median) and used as statistical features in the CVANN classifier. A 10-fold cross validation was employed and Peker (2016b) reported a mean classification accuracy of 95.42% for AASM standards and 93.84% for R&K

standards.

More recently, Supratak, Dong, Wu, and Guo (2017, Abstract) presented a model that used a combination of "convolutional neural network (CNN) to extract time invariant features and bidirectional-long Short-term memory to learn transition rules" from the raw EEG data. The study examined 20 recordings of the *Sleep-EDF 2013* data set with a 20-fold cross validation. Supratak et al. (2017) reported a mean accuracy of 82% and $\kappa = 0.76$ for AASM state classification.

The study by Mousavi, Afghah, and Acharya (2019, Abstract) used "deep CNNs to extract time-invariant features, frequency information, and a sequence to sequence model to capture the complex and long short-term context dependencies between sleep epochs and scores". This study used two single EEG signals (Fpz-Cz and PzOz) from the *Sleep-EDF 2013* and *Sleep-EDF 2018* data sets. The analysis employed a 20-fold cross validation for the *Sleep-EDF 2013* and 10-fold cross validation for the *Sleep-EDF 2018* data set. For the *Sleep-EDF 2013* data, Mousavi et al. (2019) reported 84.26% accuracy and $\kappa = 0.79$ for the Fpz-Cz channel and 82.83% accuracy and $\kappa = 0.77$ for the Pz-Oz channel. For classification performance on *Sleep-EDF 2018* data Mousavi et al. (2019) reported 80.03% accuracy and $\kappa = 0.73$ for the Fpz-Cz channel and 77.56% accuracy and $\kappa = 0.69$ for the Pz-Oz channel.

2.3 Hidden Markov Models

2.3.1 A Brief History

The hidden Markov model (HMM) was first developed during the years 1966-1969 by Leonard Baum and his colleague, Ted Petrie, at the Institute for Defense Analysis. At the end of the 1960s, Baum and his colleague, Lloyd Welch, developed the Baum-Welch algorithm (Baum, Petrie, Soules, & Weiss, 1970) for estimating the unknown parameters of HMM. "A computationally efficient iterative procedure for maximum likelihood estimation of parameters for a HMM using the forward-

backward algorithm recursions within the Expectation-Maximization algorithm" (Ephraim & Merhave, 2002, pp. 1519-1520). Specifically, the Baum-Welch algorithm is the variation of the Expectation-Maximization algorithm specific to the HMM. However, it is worth noting that the work by Baum et al. (1970) was an important precursor to the work by Dempster, Laird, and Rubin (1977), who were the first to fully define the Expectation-Maximization algorithm in a general setting and provide proofs for its properties.

In the early 1980s, "HMMs were used for automatic speech recognition in studies conducted at the Institute for Defense Analysis and by another group at AT&T Bell Laboratories" (Ephraim & Merhave, 2002, p. 1520). Rabiner (1989) provided a wonderful tutorial of HMMs and their application to speech recognition, which provided a solid foundation for the automatic continuous speech recognition seen today. Once the studies that used HMMs for automatic speech recognition were published in academic journals read by engineers, not just pure mathematicians, researchers from many disciplines started to use HMMs to model real world phenomena. Since the 1990s, HMMs were further developed and today HMMs are employed in many applications that include, but are not limited to, computational finance, cryptanalysis, EEG analysis, speech synthesis, gene prediction, time series analysis, transportation forecasting, and ecological survival.

2.3.2 Hidden Markov Models and Sleep

The earliest work using HMM in sleep state classification and understanding of human sleep was done by Zung, Naylor, Gianturco, and Wilson (1966). The use of HMMs describing human sleep was further explored by Yang and Hursch (1973) and again by Kemp and A. C. Kamphuisen (1986). Around a decade later, Penny and Roberts (1999) developed the framework of Gaussian observation HMM for EEG analysis. They used synthetic EEG data to show "HMMs can detect changes in [Direct Current] levels, correlation, frequency and coherence that are typical of the non-stationarities in an EEG signal" (Penny & Roberts, 1999, Abstract).

Flexer, Dorffner, Sykacekand, and Rezek (2002) further investigated the Gaussian observation HMM in the sleep state classification setting. This was done using a single EEG channel in 9 adult PSG recordings from five different European sleep laboratories, "[the goal was] not to replicate R&K scoring but to find a new description of human sleep which is based on the comparably unambiguous 'extreme' cornerstones of traditional sleep staging. Since R&K sleep staging is based on a predefined set of rules which leave much room for subjective interpretation there can be considerable disagreement between human scorers analyzing the same sleep recording" (Flexer et al., 2002, pp. 3-4). Flexer, Gruber, and Dorffner (2005) considered the same Gaussian observation HMM as Flexer et al. (2002), but used data sets from two different sleep laboratories. The first data set consisted of 40 healthy adult PSG recordings and the second consisted of 28 healthy adult PSG recordings. The data sets were both split evenly into testing and training data sets, as well as being balanced for age and gender. The overall classification accuracy with respect to R&K sleep scores was 54% for the first data set and 42.33% for the second data set. The most recent study found that uses the GOHMM, was Längkvist, Karlsson, and Loutfi (2012). This study examines the same Physionet data set as (Acharya et al., 2010). "Subjects were randomly selected over a 6-month period (September 02 to February 03) from patients referred to the Sleep Disorders Clinic at St Vincent's University Hospital, Dublin, for possible diagnosis of obstructive sleep apnea, central sleep apnea or primary snoring. Subjects had to be above 18 years of age, with no known cardiac disease, autonomic dysfunction, and not on medication known to interfere with heart rate. Twenty-five subjects (21M, 4F) were selected (age: 50 ± 10 years, range $28 - 68$ years; BMI: 31.6 ± 4.0 $\frac{kg}{m^2}$, range $25.1 - 42.5$ $\frac{kg}{m^2}$; AHI: 24.1 ± 20.3, range $1.7 - 90.9$)" (Goldberger et al., 2000). Längkvist et al. (2012, p. 4) considered epochs of 1s instead of 30s and "[each] epoch before and after a sleep stage switch is removed from the training set to avoid possible subsections of mislabeled data within one epoch. This resulted in 20.7% of total training samples to be removed". 28 hand crafted features were extracted from the epochs, and used in three models. This first model (feat-GOHMM)performs

a feature selection, then transforms the data with a principal component analysis, followed by a GMM and lastly a HMM for classification of sleep states. The second (feat-DBN) and third (raw-DBN) models use Deep Belief Networks trained using all features for the second model and the raw data for the third model. They performed a LOPOCV approach for each of the three models where approximately 250,000 epochs were chosen randomly from 24 patients PSGs to train the models, then 50,000 epochs from the last patient were used as the testing data. Längkvist et al. (2012) report accuracies of $63.9\% \pm 10.8$, $72.2\% \pm 9.7$, and $67.4\% \pm 12.9$ for their models.

The discretization of observations methods for HMM have been implemented by G Doroshenkov, Konyshev, and Selishchev (2007, p. 27), who used a clustering algorithm to "isolate a group of uniform elements equal to the number of states. The centers of the clusters are used as observation centers". The performance of the HMM had a mean state accuracy of 61.08% with respect to R&K standards. Pan, Kuo, Zeng, and Liang (2012) used a discrete-HMM and enforced certain constraints on the transition probabilities from one sleep state to another. Pan et al. (2012) used PSG recordings from 20 healthy adults scored with AASM standards. Subjects were evenly sorted into testing and training data sets. The enforced constraints reduced the number of estimated model parameters and performed relatively well. They reported subject-wise accuracies between 77.09% and 92.62% and κ values between 0.64 and 0.78. Another discretization of observations approach was explored by Chen, Zhu, and Chen (2015) that used vector quantization to discretize all numerical features, thus partitioning the original feature space into discrete codewords. Chen et al. (2015) considered various sizes of libraries, of codewords for the analysis with their HMM. Classification was done in a LOPOCV setting with respect to four states: wake, deep sleep, light sleep and REM. The study by Chen et al. (2015) used the ECG recordings of 15 healthy adults subjects (7 male, 8 female) from the *CAP* data set (Goldberger et al., 2000) and reported a mean classification accuracy of $79.85\% \pm 6.31$.

Yaghouby, Modur, and Sunderam (2014) explored the comparison of the HMM,

with the Gaussian mixture model, the K-means classifier, and linkage trees for sleep state classification and used the same 15 ECG recordings from the *CAP* data set as Chen et al. (2015). κ statistics were used to compare the classifier performances. Yaghouby et al. (2014) reported K-means, linkage trees, and Gaussian mixture model all had median patient κ's between 0.5 and 0.6 agreement with AASM standards. The HMM outperformed the other three classifiers with a median $\kappa = 0.70$. The studies by Yaghouby et al. (2014) and Chen et al. (2015) provide evidence for accurate sleep state classification without the use of EEG features.

2.4 Non-Homogeneous Hidden Markov Models

2.4.1 A Brief History of Applications

One of the first natural extensions of HMM is allowing the transition probabilities to vary with time or be dependent on extraneous covariates. In this setting, the transition probabilities are non-homogeneous. See chapter 10 of Zucchini, MacDonald, and Langrock (2016), for information about the incorporation of time varying and extraneous covariates into an HMM.

The Non-Homogeneous Hidden Markov Model (NHMM) was first explored by Hughes, Guttorp, and Charles (1999) to study hydrology. Hughes et al. (1999, p. 17) "[modelled] a 15-year record (1978-1992) of daily winter rainfall occurrences (2760 days, total) at 30 stations in south-western Australia". The model was able to accurately reproduce the last five years of statistical rainfall information with little bias. Ocañ-Riola (2005) used a NHMM to model breast cancer data of 24 women. A three state model was proposed: symptoms, no symptoms, and death (an absorbing state). The primary focus of the research was estimation of state transition probabilities. They found that "the probability that a patient who was with symptoms [half way through treatment] shall be without symptoms one year later is 0.80. Only 10% of women who are without symptoms one year after the diagnosis will be dead five years later" (Ocañ-Riola, 2005, p. 373). Lagona, Maruotti, and Picone (2011)

examined multi-pollutant exceedance, a multivariate time series measuring hourly levels of air pollutants at six weather stations in Rome, Italy, in 2009. The study estimated NHMMs with varying number of states. The estimated 3-state NHMM was the best at clustering days "according to their maximum posterior probabilities of class membership" (Lagona et al., 2011, p. 215). Another environmental study using NHMMs, by Ailliot, Bessac, Monbet, and Pène (2015), examined a bi-variate wind time series for the month of January off the French Atlantic coast from 1958 to 2001. Ailliot et al. (2015, Abstract) found that their proposed NHMM could "reproduce complex features of wind time series such as non-linear dynamics and multimodal marginal distributions". These studies provide evidence that NHMMs have success modelling real world phenomena in different research areas, opening the door for their use in modelling sleep data.

2.4.2 Non-Homogeneous Hidden Markov Models and Sleep

A closer examination of Figure 1.B shows an important, yet subtle characteristic of sleep. As a person's sleep progresses, the proportions of NREM and REM in each sleep cycle change. In the beginning of the night NREM sleep dominates the sleep cycle, but as sleep progresses, the proportion of NREM decreases in favor of more REM sleep towards the end of the night. This change in the proportions REM and NREM sleep states might affect the sleep state transition probabilities as the night progresses. A person is less likely to transition to a NREM state at the end of the night as compared to the beginning. Thus, the sleep state transition probabilities may be non-homogeneous with respect to time.

According to AASM standards for scoring NREM 2, experts are recommended to "continue to score epochs with low-amplitude, mixed-frequency EEG activity without K complexes or sleep spindles as stage [NREM 2] if they are preceded by epochs containing [either] of the following: [1]. K complexes unassociated with arousals, [2]. Sleep spindles" (Iber et al., 2007, p. 23). Sleep experts are trained to use information from the previous epoch to score the current one. Therefore, it

makes sense to incorporate statistical information from a previous epoch or epochs to model the time-varying sleep state transition probabilities.

A NHMM that used statistical information from the previous epoch has been applied in a previous sleep EEG study of five Zebra finch birds by Xu (2005). The model assumed multivariate Gaussian distributions for the state dependent distributions. The transition probabilities were modelled with a multinomial logistic link function that used the current observation to model the next state transition. Four frequency domain features were extracted from each epoch and used as observations. Xu (2005) used 1200 consecutive epochs from the fourth bird as training data and 1000 consecutive epochs from the fifth bird as test data. All epochs in these data sets were scored as REM or NREM by a sleep expert. Using classification accuracy of the test data, Xu (2005) compared the performance of a Gaussian mixture model, HMM, and NHMM. Xu (2005) found the HMM achieved the highest accuracy of 81.0%, the Gaussian mixture the lowest at 73.7%, and the NHMM achieved 78.60% accuracy, which suggests that classifiers that model a dependence structure for the sleep states may also perform better for human sleep state classification.

The study by Trevenen, Turlach, Eastwood, Straker, and Murray (2019) used first and second order Gaussian observation HMM and NHMM on raw accelerometer data to classify sleep states for 242 healthy adult subjects. Epochs were scored into sleep states by an expert using separate PSG information for classification purposes. The raw 3-dimension accelerometer data for each subject was segmented into epochs. A total of nine measures were collected from each accelerometer epoch as features. A baseline comparison of continuous Gaussian observation HMM and NHMM was done against the generalized linear mixed model, which, unlike the Gaussian mixture, can model a dependence structure between observations by adding random effects. However, the generalized linear mixed model does not model the dependence structure between the sleep states.

The NHMMs of Trevenen et al. (2019) were much simpler than that of Xu (2005). The collection of epochs for each patient were divided into two and three segments with each segment pertaining to one section of the night. A HMM (of or-

der 2) is estimated separately for each segment of the night. In this setting, transition probabilities were not modelled using previous observations via a link function.

A 10-fold (subject wise) cross validation was used to evaluate the classification performance. In the 5-state classification Trevenen et al. (2019) reported the second order NHMM performed the best overall. The median accuracy (Inter-Quartile Range) of each state for the second order NHMM was reported: Wake 0.61(0.281), NREM 1 0.100(0.087), NREM 2 0.522(.460), NREM 3 0.638(.658) and REM 0.064(0.367) Trevenen et al. (2019).

Chapter 3

Methodology

3.1 Random Forest

The random forests methodology of Ho (1995, 1998) is a popular ensemble learning method for classification and regression. A multitude of decision trees is constructed during training and the output is the class label that is the mode, or the lowest mean square error in regression. Decision trees are constructed using a random number of features or covariates and bootstrap samples of the training data. The randomness of features selected helps to ensure the collection of trees constructed does not overfit the training data, even if individual trees have the habit of doing so. The bootstrap aggregation (bagging) method was introduced by Breiman (2001), where the remaining training data not selected to construct each tree is called the out-of-bag portion. The classification error of the out-of-bag portion, or mean square error for regression, is used to validate the performance of the constructed tree. Using random bootstrap samples to construct each tree means that even if a single tree overfits the sample, the collection of trees will not overfit the entire training data.

Random forest methodology was used in this work for two purposes: to compare the classification classification performance with the literature Fraiwan et al.

(2012), Boostani et al. (2017), and da Silveira et al. (2017). As well as for selecting the features to create a final data set to be used in the HMM and NHMM analysis.

3.1.1 Comparison with Literature

The random forests implemented in Fraiwan et al. (2012) and da Silveira et al. (2017) were recreated in this work using the `R` library `randomForest` (Liaw & Wiener, 2002) and then used to compare classification performance of the CF00N data with the respective literature. This random forest design took different bootstrap samples of the training data to construct each single decision tree. The process was repeated, constructing forests of 10, 64, and 128 trees. The graphical structure for the construction used is seen in Figure 3.A.

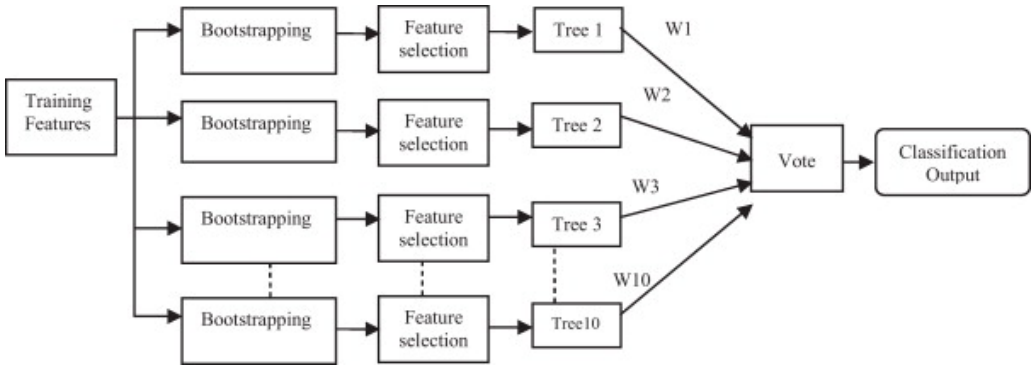


Figure 3.A: Random forest structure of Fraiwan et al. (2012). da Silveira et al. (2017) uses the same structure, but increases the tree number from 10 to 64. Image taken from Fraiwan et al. (2012)

Fraiwan et al. (2012) examined a single EEG channel of 16 PSG recordings from the *Sleep EDF 2002* data set and then extracted the Renyi entropy of continuous wavelet coefficients in seven sequential frequency bands. They constructed a random forest of 10 trees using (2/3) training and (1/3) testing split on the data. da Silveira et al. (2017) examined a single EEG channel from the *Sleep EDF Expanded* data set and extracted statistical moments (variance, skewness, kurtosis) of the discrete wavelet coefficients in six sequential frequency bands. They constructed a forest with 64 trees and used 10-fold cross validation to examine classification performance. Furthermore, they also used the same training and testing split

as Fraiwan et al. (2012) for comparative purposes. See Chapter 4 for further details on these feature sets.

This thesis extended the cross validation of the random forest analysis one step further by examining a classification performance using a LOPOCV approach. This was done to ensure the testing and training data did not contain epochs from the same patients, as this can inflate the classification accuracy. The purpose of this extension was to examine the performance of random forest sleep state classification in a clinical setting.

3.1.2 Feature Selection Tool

The `randomForest` (Liaw & Wiener, 2002) library allows for nested calculations of cross validation error across the number of randomly selected variables used to build trees, which essentially allows researchers to find the optimal number of random features, r, used to construct a random forest that produces the smallest classification error. Then, using this r, another random forest of the data was constructed to extract the importance measures of each variable in order to find which features are to be selected. Since the classification performance will be extended to a LOPOCV, the feature selection strategy stated above was used on a combined version of each patient's epoch features to ensure that the variables selected were appropriate for classifying the epochs of individual patients.

Statistical features from all feature sets were combined into a single data frame for each patient. A 10-fold cross validation scheme was employed to analyze the number of ideal features needed for the individual patients. Random forests of 10,000 trees were grown for every possible value of r and the cross validation error of each forest calculated. This process was repeated for every patient to assess the cross validation error across all values of r for all patients. The main criteria for determining r was the minimum cross validation error for all patients. Once r was found, a single forest of 10,000 trees was grown for each patient's data that used r randomly selected variables to build each tree. The variable importance measures

from each patient's random forest were extracted and then used to find at least r features which best discriminated between the sleep states for all patients.

3.2 Hidden Markov Model

The HMM structure, notation, and properties used for the analysis are closely adopted from Zucchini et al. (2016). The proofs of theorems and properties presented can be found in Zucchini et al. (2016) with the heavier details included in Appendix B. As well, Sucar (2015) and the tutorial by Rabiner (1989) are recommended for supplementary source material. The programming of the HMM algorithms for evaluation, decoding, and parameter estimation are implemented in the `R` library `depmixS4` (Visser & Speekenbrink, 2010). This library is well equipped to evaluate, decode, and estimate HMM parameters, while simultaneously allowing for extensions to the basic HMM that are applied in this work. However, the methods described here-in will be those of Zucchini et al. (2016) in order to build a solid foundation of HMM notation and concepts. The parameter estimation employed in `depmixS4` will be described in the NHMM section of this chapter.

3.2.1 HMM Model, Parameters and Properties

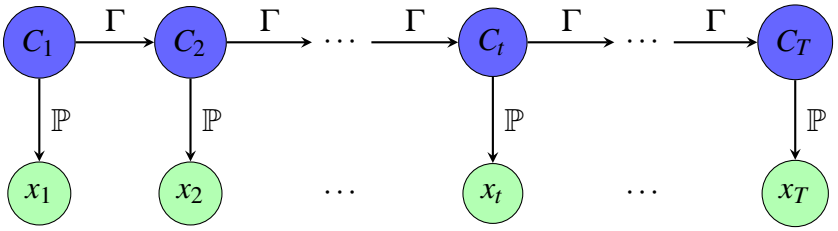


Figure 3.B: The directed graph of the hidden Markov model. At each time t the chain emits an observation $\mathbf{x}_t$ from the current unobserved state C_t. This forms two time series $C^{(T)}$ and $\mathbf{x}^{(T)}$ whose relationship is governed through Γ and $\mathbb{P}$. A Markov chain is used to model Γ and Gaussian probability density functions to model $\mathbb{P}$.

The Gaussian Observation HMM Sleep Model

Let T = the number of observations and the state space S={Wake, NREM 1, NREM 2, NREM 3, REM}. For the purpose of summation notation and without loss of generality, the number of states in S is defined as m. The hidden sleep state sequence of the epochs is denoted $C^{(T)} = C_1, \ldots, C_T \in$ S. The extracted statistical features of each epoch are the emitted observations $\mathbf{x}^{(T)} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$, where $\mathbf{x}_t$ is the t^{th} observation. The multivariate domain of the emitted observations is $\mathbb{R}^d$, where d is the number of features for each epoch.

Parameters

γ_{ij} = $\Pr(C_{t+1} = j | C_t = i)$ is the probability of the system (patient) transitioning from state i at time t to state j at time $t+1$, for all $i, j \in S$. The $m \times m$ transition matrix Γ = $[\gamma_{ij}]$ is the collection of all possible combinations of i, j transitions.

p_{tj} = $\Pr(\mathbf{X}_t = \mathbf{x}_t | C_t = j)$, $t \in \{1, ..., T\}$, $j \in S$ are defined as the emission probabilities. The probability of observing $\mathbf{x}_t$ at time t when $C_t = j$. The multivariate Gaussian distribution was used to model the observations emitted from the sleep states. The observations emitted from state j are dependent on the state parameters (μ_j, Σ_j), and $\Pr(\mathbf{X}_t = \mathbf{x}_t | C_t = j) = \phi(\mathbf{x}_t | \mu_j, \Sigma_j)$, where

$$\phi(\mathbf{x}_t | \mu_j, \Sigma_j) = ((2\pi)^d |\Sigma_j|)^{-\frac{1}{2}} exp\{-\frac{1}{2}(\mathbf{x}_t - \mu_j)' \Sigma_j^{-1} (\mathbf{x}_t - \mu_j)\}$$

$\mathbb{P} = [p_{tj}]$ is defined as the $T \times m$ matrix of emmission probablities. The initial distribution $\mathbf{u}(1) = [\Pr(C_1 = 1), \Pr(C_1 = 2), \ldots, \Pr(C_1 = m)]$ is the row vector of unconditional probabilities for the chain starting in each state. In a compact form the HMM is represented as H=($\mathbf{u}(1)$, Γ, $\mathbb{P}$).

Properties

Markov Property: The transition to the next sleep state in the sequence is dependent only on the current sleep state.

$$\Pr(C_t = j | C_{t-1} = i, C_{t-2} = k...) = \Pr(C_t = j | C_{t-1} = i) \text{ for } i,\ j,\ k \in S$$

In general, HMMs have three additional properties that can be relaxed or modified to create variations of HMMs that model complex phenomena. Zucchini et al. (2016) (Chapter 10) give a nice catalogue of the basic extensions, along with other variations used for real world applications.
Time-homogeneous:

$$\Pr(C_t = j | C_{t-1} = i) = \Pr(C_{t+l} = j | C_{t+l-1} = i)$$

Stationarity: For a HMM to be considered stationary, the unconditional probabilities are the same for $\forall\, t$, $\mathbf{u}(1) = \ldots = \mathbf{u}(t) = \ldots \mathbf{u}(T) = \delta$, which means the probability of being in any state is the same for all t. This property is not assumed for sleep states, due to changing proportions of REM and NREM in different sleep cycles.
Independence of the observations:

$$\Pr(\mathbf{X}_t = \mathbf{x}_t | C_t = j, C^{(t-1)} = c^{(t-1)}, \mathbf{X}^{(T-1)} = \mathbf{x}^{(t-1)}) = \Pr(\mathbf{X}_t = \mathbf{x}_t | C_t = j)$$

The observations only depend on the current state, not the previous states or observations.

3.2.2 Joint Probability of States and Observations

The joint probability of states and observations of the graphical model for the HMM is found by using the chain rule for Bayesian networks. The joint distribution of a set of random variables $X_1, \ldots, X_N$ represented in a Bayesian network graph is

$\Pr(X_1,\ldots,X_N) = \prod_{i=1}^{N} \Pr(X_i|pa(X_i))$ where $pa(X_i)$ is the set of all parents of X_i.

In the HMM directed graphical model, Figure 3.B, C_1 has no parent, C_t has parent C_{t-1} for t = 2,..., T, and $\mathbf{X}_t$ has parent C_t. Hence, the joint probability for the HMM is given by

$$\Pr(C^{(T)},\mathbf{X}^{(T)}) = \Pr(C_1)\Pr(\mathbf{X}_1|Pa(\mathbf{X}_1))\ldots\Pr(C_T|Pa(C_t))\Pr(\mathbf{X}_T|Pa(\mathbf{X}_T))$$

$$= \Pr(C_1)\Pr(\mathbf{X}_1|C_1)\ldots\Pr(C_T|C_{T-1})\Pr(\mathbf{X}_T|C_T)$$

$$= \Pr(C_1)\Pr(\mathbf{X}_1|C_1)\prod_{t=2}^{T}\Pr(C_t|C_{t-1})\Pr(\mathbf{X}_t|C_t)$$

The form of the individual distributions of $\mathbf{u}(1)$, Γ, and $\mathbb{P}$ are not as important as the fact that the joint probability factors in a way that is represented by Figure 3.B. Marginalizing the joint probability of states and observations is done to find the joint probability of the observations. This factorization of the joint probability function is what makes HMMs a natural fit for modelling sequences of sleep states.

3.2.3 HMM Likelihood

Given a sequence of observations $\mathbf{x}^{(T)}$ emitted from H=($\mathbf{u}(1)$, Γ, $\mathbb{P}$), estimating the probability of observing the sequence, $\mathbf{L}_T = \Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)})$, is done by calculating the value of the likelihood. To calculate $\mathbf{L}_T$ one needs to marginalize $\Pr(C^{(T)} = c^{(T)}, \mathbf{X}^{(T)} = \mathbf{x}^{(T)})$ by summing over all possible state sequences. Representation of this calculation by Zucchini et al. (2016) requires $\mathbf{L}_T$ to be a product of matrices. First, define $\mathbf{P}(\mathbf{x}_t)$ as the diagonal matrix whose entries are the t^{th} row of $\mathbb{P}$, evaluated for $\mathbf{x}_t$.

$$\mathbf{P}(\mathbf{x}_t) = \begin{pmatrix} \Pr(\mathbf{X}_t = \mathbf{x}_t|C_t = 1) & & & 0 \\ & \Pr(\mathbf{X}_t = \mathbf{x}_t|C_t = 2) & & \\ & & \cdots & \\ 0 & & & \Pr(\mathbf{X}_t = \mathbf{x}_t|C_t = m) \end{pmatrix}$$

Then,

$$\mathbf{L}_T = \Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}) = \mathbf{u}(1)\mathbf{P}(\mathbf{x}_1)\Gamma\mathbf{P}(\mathbf{x}_2)\Gamma\mathbf{P}(\mathbf{x}_3)\cdots\Gamma\mathbf{P}(\mathbf{x}_T)\mathbf{1}'$$

The brute force calculation of L_T across all possible state sequence combinations is of the order m^T, which, even with a computer, is not feasible for large values of T. The Forward algorithm or what is sometimes called the Forward-Backward algorithm, is the recursive solution used to calculate the likelihood.

3.2.4 Forward and Backward Probabilities

Forward Probabilities

To understand how the Forward algorithm works, one must first define the $1 \times m$ row vectors of forward probabilities and the $m \times 1$ column vectors of backward probabilities. Zucchini et al. (2016) define the row vectors α_t, $t = 1, \ldots, T$, where $\alpha_t(j) = \Pr(\mathbf{X}^{(t)} = \mathbf{x}^{(t)}, C_t = j)$ is the joint probability of a partial sequence of observations $\mathbf{X}^{(t)} = \mathbf{x}^{(t)}$ being in state j at time t.
Begin with

$$\begin{aligned}\alpha_t &= \mathbf{u}(1)\mathbf{P}(\mathbf{x}_1)\Gamma\mathbf{P}(\mathbf{x}_2)\Gamma\mathbf{P}(\mathbf{x}_3)\cdots\Gamma\mathbf{P}(\mathbf{x}_t) \\ &= \mathbf{u}(1)\mathbf{P}(\mathbf{x}_1)\prod_{k=2}^{t}\Gamma\mathbf{P}(\mathbf{x}_k), \text{ where} \\ \alpha_1 &= \mathbf{u}(1)\mathbf{P}(\mathbf{x}_1) \text{ with } \alpha_t = \alpha_{t-1}\Gamma\mathbf{P}(\mathbf{x}_t), \text{ for } t = 2, \ldots, T\end{aligned}$$

This results in the row vector

$$\alpha_t = \Pr(\mathbf{X}^{(t)} = \mathbf{x}^{(t)}, C_t), \text{ and } \alpha_t(j) = \Pr(\mathbf{X}^{(t)} = \mathbf{x}^{(t)}, C_t = j)$$

Backward Probabilities

In an analogous way that defined the forward probabilities, Zucchini et al. (2016) define the $m \times 1$ column vectors of backward probabilities. As the name implies,

calculating these probabilities starts at $t = T$ and works backwards to $t = 1$. Formally,

$$\beta_t^{'} = \Gamma\mathbf{P}(\mathbf{x}_{t+1})\Gamma\mathbf{P}(\mathbf{x}_{t+2})\cdots\Gamma\mathbf{P}(\mathbf{x}_T)\mathbf{1}'$$

$$= \left(\prod_{k=t+1}^{T}\Gamma\mathbf{P}(\mathbf{x}_k)\right)\mathbf{1}' = \Gamma\mathbf{P}(\mathbf{x}_{t+1})\beta_{t+1}^{'} \text{ for } t = 1,\ldots,T-1$$

This results in the row vector

$$\beta_t^{'} = \Pr(\mathbf{X}_{t+1}^T = \mathbf{x}_{t+1}^T|C_t) \text{ and } \beta_t^{'}(j) = \Pr(\mathbf{X}_{t+1}^T = \mathbf{x}_{t+1}^T|C_t = j)$$

where $\mathbf{x}_{t+1}^T = (\mathbf{x}_{t+1}, \mathbf{x}_{t+2}, \ldots, \mathbf{x}_T)$.

The Forward algorithm inductively uses the α_t's moving forward from $t = 1$ to $t = T$ to recursively calculate the likelihood. The Forward algorithm by Sucar (2015) using this dynamic is given on the following page. This recursion reduces the order of calculations from m^T to Tm^2. The backward probabilities can be used in a similar fashion to calculate the likelihood of observations in a single backward pass through the data. However, there are actually T paths that can be used to make this calculation, which is achieved by using the **Property 1** of the forward and backward probabilities, given in the next section.

The Forward Algorithm of Sucar (2015, p. 71)

Given a HMM, $H = (\mathbf{u}(1), \Gamma, \mathbb{P})$, and $\mathbf{X}^{(T)} = \mathbf{x}^{(T)}$

Initialization, for $j = 1$ to m compute

$$\alpha_1(j) = \Pr(\mathbf{X}_1 = \mathbf{x}_1, C_1 = j) = \mathbf{u}_j(1) p_{1j}$$

for $t = 2$ to T

for $j = 1$ to m compute

$$\alpha_t(j) = \left[\sum_{i=1}^{m} \alpha_{t-1}(i) \gamma_{ij} \right] p_{tj}$$

compute $\mathbf{L}_T = \sum_{j=1}^{m} \alpha_T(j)$

return($\mathbf{L}_T$)

3.2.5 Properties of Forward and Backward Probabilities

Property 1: Proposition 4 of Zucchini et al. (2016) For $t = 1, \ldots, T$ and $i = 1, \ldots, m$

$$\alpha_t(i)\beta_t(i) = \Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}, C_t = i)$$

$$\text{and } \alpha_t \beta_t^{'} = \Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}) = \mathbf{L}_T$$

Recall that α_t is $1 \times m$ and β_t is $m \times 1$, which means that their dot product $\mathbf{L}_t$ is 1×1

Property 2: Part 1 of Proposition 5 of Zucchini et al. (2016) For $t = 1, \ldots, T$

$$\frac{\alpha_t(j)\beta_t(j)}{\mathbf{L}_T} = \Pr(C_t = j | \mathbf{X}^{(T)} = \mathbf{x}^{(T)})$$

Property 3: Part 2 of Proposition 5 of Zucchini et al. (2016) For $t = 2, \ldots, T$

$$\frac{\alpha_{t-1}(i)\gamma_{ij}p_{tj}\beta_t(j)}{\mathbf{L}_T} = \Pr(C_{t-1} = i, C_t = j | \mathbf{X}^{(T)} = \mathbf{x}^{(T)})$$

The last two properties allow for calculation of the marginal and joint conditional probabilities of sleep states, given the observations. These probabilities are then used for decoding the most probable sequence of sleep states. Put more formally, a clustering each of the patient's epochs into distinct states that maximizes the log-likelihood of observations.

3.2.6 Decoding the Hidden States

In local decoding, the goal is to find the most probable sleep state at time t, given the observations. This is done using **Property 2** by calculating the conditional probabilities of each C_t, given the observations. C_t is then assigned to the sleep state with the largest probability.

$$c_t = \underset{j \in S}{\operatorname{argmax}} \Pr(C_t = j | \mathbf{X}^{(T)} = \mathbf{x}^{(T)})$$

This can be done for all C_t individually and the solutions, $c_1, \ldots, c_T$, combined into a sequence of sleep states. However, this solution does not account for transition probabilities between sleep states, as the maximum probability is considered for each t independently.

Global decoding, on the other hand, does account for the transition probabilities. The goal is to find the maximum *a-posteriori* (MAP) of all states, in order to obtain the most probable sequence of sleep states, given the observations. Zucchini et al. (2016) defined $c^{*(T)}$ to accomplish this task, where

$$c^{*(T)} = \underset{c^{(T)}}{\operatorname{argmax}} \Pr(C^{(T)} = c^{(T)} | \mathbf{X}^{(T)} = \mathbf{x}^{(T)})$$

Instead of maximizing the conditional probability of sleep states given observations, $\Pr(C^{(T)} = c^{(T)} | \mathbf{X}^{(T)} = \mathbf{x}^{(T)})$, the joint probability, $\Pr(C^{(T)} = c^{(T)}, \mathbf{X}^{(T)} = \mathbf{x}^{(T)})$, is maximized. Maximization of the joint probability yields the solution $c^{*(T)}$, which also maximizes the conditional probability. However, "maximizing [the joint probability] over all possible state sequences $c_1, c_2, \ldots, c_t$ by brute force requires the evaluation m^T probabilities" (Zucchini et al., 2016, p. 89). The Viterbi algorithm (Viterbi, 1967; Forney, 1973) is used to reduce the computational order.

The Viterbi Algorithm

To begin, Zucchini et al. (2016) define for $t = 1$

$$\xi_{1j} = \Pr(C_1 = j, \mathbf{X}_1 = \mathbf{x}_1) = \mathbf{u}_j(1) p_{1j}$$

and, for $t = 2, 3, \ldots, T$,

$$\xi_{tj} = \max_{c_1, \ldots, c_{t-1}} \Pr(C^{(t-1)} = c^{(t-1)}, C_t = j, \mathbf{X}^{(t)} = \mathbf{x}^{(t)})$$

Where ξ_t is the $1 \times m$ vector of maximum probabilities for a sub-sequence of sleep states and observations, up to time t. Fortunately, like the forward and backward probabilities, the ξ_{tj} has a recursive solution. For $t = 2, 3, \ldots, T$ and $j = 1, \ldots, m \in S$

$$\xi_{tj} = \left(\max_i (\xi_{t-1,i} \cdot \gamma_{ij}) \right) p_{tj}$$

Decoding the global solution involves inductively working backwards and starts with c_T^*, the state which maximizes

$$\Pr(C^{(T-1)} = c^{(T-1)}, C_T = c_T^*, \mathbf{X}^{(T)} = \mathbf{x}^{(T)})$$

Using c_T^*, to find the state c_{T-1}^* that maximizes the joint probability of the most probable sequence of states and observations up to time $T - 1$, while accounting for

the transition to state c_T^* from state c_{T-1}^* at time T, is found via the recursion.

$$\Pr(C^{(T-2)} = c^{(T-2)}, C_{T-1} = c_{T-1}^*, \mathbf{X}^{(T-1)} = \mathbf{x}^{(T-1)}) \times \Pr(C_T = c_T^* | C_{T-1} = c_{T-1}^*)$$

Continue in this backwards fashion for $t = T-2, \ldots, 1$ and the end result is the most probable sequence of sleep states $c^{*(T)}$. When executing this procedure in a programming language, (Sucar, 2015, p. 73) recommends "[introducing] an additional variable $[\psi_{t,j}]$, that stores for each state j at each time step t the previous state that gave the maximum probability ξ_{tj}." The Viterbi algorithm of Sucar (2015) is given below.

The Viterbi Algorithm: Algorithm 5.2, (Sucar, 2015, p. 71)

MAP(*Maximum a-posteriori*) = empty vector of length T

Initialization, for $i = 1$ to m

$$\text{compute } \xi_{1i} = \mathbf{u}_i(1) \cdot p_{1i} \text{ and set } \psi_{1i} = 0$$

for $t = 2$ to T for $j = 1$ to m

$$\xi_{tj} = \left(\max_i (\xi_{t-1,i} \cdot \gamma_{ij}) \right) p_{tj} \text{ and } \psi_{tj} = \operatorname*{argmax}_i (\xi_{t-1,i} \cdot \gamma_{ij})$$

$$\text{MAP}[T] = \operatorname*{argmax}_i (\xi_{T-1,i} \cdot \gamma_{ij})$$

for $t = T$ to 2

$$c_{t-1}^* = \text{MAP}[t-1] = \psi_{tc_t^*}$$

return(MAP), where MAP $= [c_1^*, c_2^*, \ldots, c_T^*]$ $= c^{*(T)}$

3.2.7 Estimation of HMM Parameters

The most common approach to parameter estimation for HMMs is the Expectation-Maximization (EM) algorithm. The Baum-Welch algorithm (Baum et al., 1970) is the EM algorithm for HMMs, and the Baum-Welch algorithm described in this work is that of Zucchini et al. (2016). The name "Expectation-Maximization" comes from the fact that every iteration in the algorithm has an expectation step (E-step) followed by a maximization step (M-step). The algorithm begins with an initial estimate of the model parameters for the complete data (observations and missing data) log-likelihood function. Then it computes the (E-step) conditional expectation of the log-likelihood, given the observations, and then maximizes (M-step) the conditional log-likelihood with respect to model parameters. The parameter values obtained are used as the estimates in the next iteration, which are updated until some convergence criteria is met or the number of iterations is exhausted. The maximum number of iterations to fit a HMM used in this work was 500 and the convergence criteria for the EM algorithm was the "relative" log-likelihood defined by Visser and Speekenbrink (2010). The log-likelihood at iteration i is $log(L_{T_i})$ and the convergence criteria is ε. The EM algorithm stops when

$$\frac{log(L_{T_i}) - log(L_{T_{i-1}})}{log(L_{T_{i-1}})} < \varepsilon$$

For a given HMM, $\theta = \{\mathbf{u}(1), \Gamma, \mu, \Sigma\}$ is the set of model parameters to be estimated. $\mathbf{u}(1)$ and Γ are the $1 \times m$ and $m \times m$ initial state and transition probabilities. The multivariate Gaussian parameters of the states are μ the array of $d \times 1$ mean vectors and Σ the collection of $d \times d$ covariance matrices. Specifically, for each $j = 1, \ldots, m$ the parameters (μ_j, Σ_j) of the multivariate Gaussian density need to be estimated, since

$$p_{tj} = \Pr(\mathbf{X}_t = \mathbf{x}_t | C_t = j) = \phi(\mathbf{x}_t | \mu_j, \Sigma_j) \sim \mathcal{N}_d(\mu_j, \Sigma_j)$$

In total, there are $m + m^2 + md + m\frac{d(d+1)}{2}$ parameters to estimate for the HMM

with m states and d-dimensional Gaussian state dependent distributions. A draw back of the EM algorithm is if the likelihood function has more than one local maximum, then the EM algorithm may not find the globally maximum solution for the estimated parameters and, consequently, the MAP sequence of sleep states. "In practice, it is common to start EM from multiple random initial guesses, and choose the one with the largest likelihood as the final guess for model parameters" (Gupta & Chen, 2011, p. 228). This approach was used to find the best global decoding of each patient's sleep states.

Baum-Welch Algorithm of Zucchini et al. (2016)

To begin the parameter estimation, Zucchini et al. (2016) define the complete data log-likelihood by representing the sequence of states, $c^{(T)}$, followed by the Markov chain using indicator variables at each time step.

$$\delta_j(t) = 1 \text{ if and only if } c_t = j \text{ for } t = 1,2,\ldots,T$$

$$\omega_{jk}(t) = 1 \text{ if and only if } c_{t-1} = j \text{ and } c_t = k \text{ for } t = 2,3,\ldots,T$$

making the complete data log-likelihood

$$\sum_{j=1}^{m} \delta_j(1) log(\mathbf{u}_j(1)) + \sum_{t=2}^{T}\sum_{j=1}^{m}\sum_{k=1}^{m} \omega_{jk}(t) log(\gamma_{jk}) + \sum_{j=1}^{m}\sum_{t=1}^{T} \delta_j(t) log(p_{tj})$$

The conditional expectation of this expression, given the observations and current parameter estimates, is commonly known as the Q-function, $Q(\theta|\hat{\theta})$, where $\hat{\theta}$ is the current parameter estimates. Thus, $Q(\theta|\hat{\theta}) =$

$$E_{C^{(T)}|\mathbf{x}^{(T)},\hat{\theta}} \left[\sum_{j=1}^{m} \delta_j(1) log(\mathbf{u}_j(1)) + \sum_{t=2}^{T}\sum_{j=1}^{m}\sum_{k=1}^{m} \omega_{jk}(t) log(\gamma_{jk}) + \sum_{j=1}^{m}\sum_{t=1}^{T} \delta_j(t) log(p_{tj}) \right]$$

The linearity of the expectation operator and the complete data log-likelihood simplifies the computation of $Q(\theta|\hat{\theta})$ by calculating the conditional expectation of each

term individually. The first term corresponds only with the initial distribution, the second with the transition probabilities, and the third with the parameters of the multivariate Gaussian state distributions. A closed form expression for the expectation of each term is found using properties of the forward and backward probabilities.

$$E_{C^{(T)}|\mathbf{x}^{(T)},\hat{\theta}}[\delta_j(t)] = \Pr(C_t = j|\mathbf{x}^{(T)}) = \frac{\alpha_t(j)\beta_t(j)}{\mathbf{L}_T}$$

$$E_{C^{(T)}|\mathbf{x}^{(T)},\hat{\theta}}[\omega_{jk}(t)] = \Pr(C_{t-1} = j, C_t = k|\mathbf{x}^{(T)}) = \frac{\alpha_{t-1}(j)\gamma_{jk}p_{tk}\beta_t(k)}{\mathbf{L}_T}$$

Now replace $\delta_j(t)$ and $\omega_{jk}(t)$ in $Q(\theta,\hat{\theta})$ with the conditional expectations, denoted $\hat{\delta}_j(t)$ and $\hat{\omega}_{jk}(t)$. This becomes

$$Q(\theta|\hat{\theta}) = \sum_{j=1}^{m} \hat{\delta}_j(1)log(\mathbf{u}_j(1)) + \sum_{t=2}^{T}\sum_{j=1}^{m}\sum_{k=1}^{m} \hat{\omega}_{jk}(t)log(\gamma_{jk}) + \sum_{j=1}^{m}\sum_{t=1}^{T} \hat{\delta}_j(t)log(p_{tj})$$

Again, the form of the complete log-likelihood allows for maximization of the terms separately with respect to their model parameters of interest. Maximization of the first two terms can be done using Lagrange multipliers. For the first term maximize

$$\sum_{j=1}^{m} \hat{\delta}_j(1)log(u_j(1)) \text{ with respect to } \mathbf{u}(1), \text{ subject to the constraint } \sum_{j=1}^{m} \mathbf{u}_j(1) = 1$$

For the second term maximize

$$\sum_{t=2}^{T}\sum_{j=1}^{m}\sum_{k=1}^{m} \hat{\omega}_{jk}(t)log(\gamma_{jk}) \text{ with respect to } \Gamma, \text{ subject to the constraint } \sum_{k=1}^{m} \gamma_{jk} = 1$$

yielding the solutions

$$\mathbf{u}_j(1) = \hat{\delta}_j(1) = \frac{\alpha_1(j)\beta_1(j)}{\mathbf{L}_T} \text{ and } \hat{\gamma}_{jk} = \frac{\sum_{t=2}^{T} \hat{\omega}_{jk}(t)}{\sum_{k=1}^{m}\sum_{t=2}^{T} \hat{\omega}_{jk}(t)}$$

Maximization of the third term is essentially finding the maximum likelihood estimates of the multivariate Gaussian distributions. Maximize

$$\sum_{j=1}^{m}\sum_{t=1}^{T}\hat{\delta}_j(t)log(p_{tj}) \text{ with respect to } \Sigma_j \text{ and } \mu_j \text{ for } j=1,\ldots,m$$

Fortunately, there is a closed form solution in this setting, yielding the Maximum likelihood estimates based on state membership.

$$\hat{\Sigma}_j = \frac{\sum_{t=1}^{T}\hat{\delta}_j(t)(\mathbf{x}_t-\hat{\mu}_j)(\mathbf{x}_t-\hat{\mu}_j)'}{\sum_{t=1}^{T}\hat{\delta}_j(t)} \qquad \hat{\mu}_j = \frac{\sum_{t=1}^{T}\hat{\delta}_j(t)\mathbf{x}_t}{\sum_{t=1}^{T}\hat{\delta}_j(t)}$$

3.2.8 Model Selection

Each patient's epochs were modelled with an HMM using 500 random starting points for the parameter estimation and the model with the largest log-likelihood was chosen. Typically, Akaike information criterion (AIC) or Bayesian information criterion (BIC) would have been used. However, for each individual patient the number of parameters and epochs does not change, meaning the model selection for each patient using AIC or BIC is equivalent to selecting the model with the largest log-likelihood.

3.3 Non-homogeneous Hidden Markov Model

The graphical representation of the NHMM model initially planned this thesis is seen in Figure 3.C. This NHMM is similar to the NHMM used by Xu (2005) for modelling REM and NREM sleep in zebra finches. Multivariate Gaussian distributions are used for the state dependent distributions and the transition probabilities are modelled with a multinomial logistic link function. In order to correctly model Figure 3.C there needs to be $\Pr(\mathbf{Y}_t = \mathbf{y}_t|\mathbf{X}_t = \mathbf{x}_t)$ incorporated into the joint proba-

bility of states and observations. The `R` library `depmixS4` (Visser & Speekenbrink, 2010) allows for the inclusion of covariates on the transition probabilities, however, in the work presented here the covariates were used direct inputs for the state transitions. The graphical model fitted in this thesis using `depmixS4` is Figure 3.D. That is, the covariates were not modelled as the result of the observations. Furthermore, a transition covariate is required for each state in the sequence. Thus, a covariate, $\mathbf{y}_0$, must be supplied for C_1 and there are two choices: $\mathbf{y}_0 = 0$, or some estimation of $\mathbf{y}_0$. In this work, $\mathbf{y}_0 = \mathbf{y}_1$ because at the beginning of the PSG recording the patient was in the wake state and the state before C_0 was also likely the wake state, which made setting $\mathbf{y}_0 = \mathbf{y}_1$ the most sensible solution. The observation data was first scaled to variance 1 and then transformed (or rotated orthogonally) using principle component analysis (PCA). The current values of the first k components, which capture 95% or more of the total variation, were used as covariates in the multinomial logistic regression model (MLR) of each state, not the previous observation. Note: the observation data is only scaled to variance 1 for the purpose of performing the PCA. The observations used in the model are the original statistical features extracted from the epochs.

$$\mathbf{x}^{(T)} \xrightarrow{PCA} \mathbf{y}^{(T)}$$

The PCA ensured that the components used in the MLR model had low collinearity; a key assumption of the MLR model. The NHMM model in Figure 3.D differs from what what was done by Xu (2005), not just in the covariates used to model transition probabilities, but the work of this thesis used five sleep states instead of two and models human sleep.

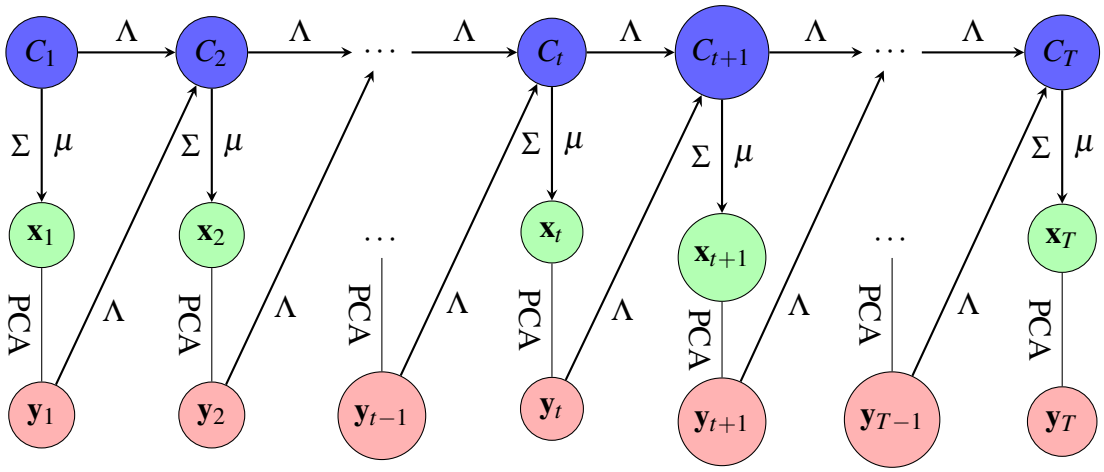


Figure 3.C: The directed graphical model of the NHMM, where transitions at time, t, depends on the current state and principle component vector $\mathbf{y}_t$, via, MLR coefficients Λ.

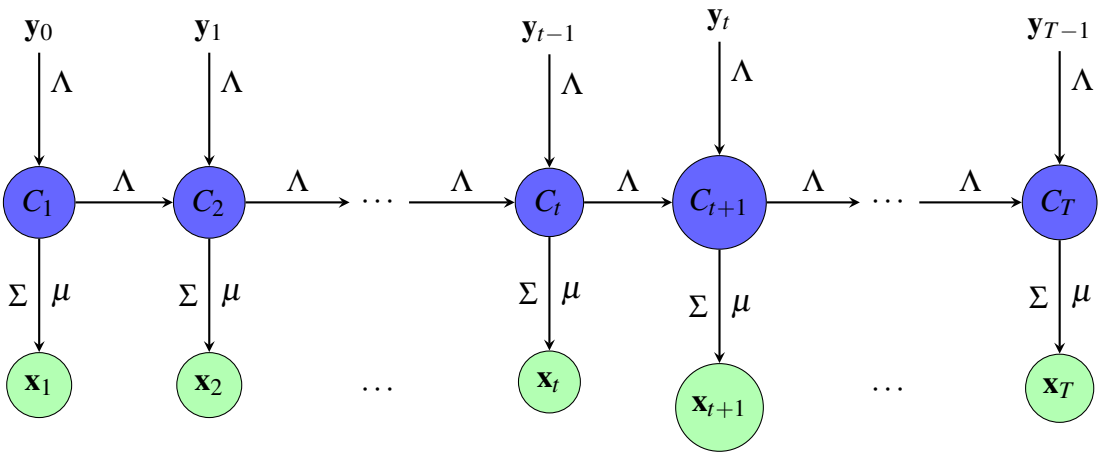


Figure 3.D: The directed graphical model of the NHMM, where the transition at time, t, depends on the current state and covariate vector $\mathbf{y}_t$, via, MLR coefficients Λ.

3.3.1 Definition of Model Parameters

In a compact form the NHMM is $NH = (\mathbf{u}(\mathbf{y}_0), \Gamma = [{}_1\Gamma, {}_2\Gamma, \ldots, {}_{t-1}\Gamma], \mathbb{P})$. Where $\mathbb{P}$ is still the $T \times m$ emission matrix defined in the HMM section of this chapter. $\Pr(C_1|\mathbf{y}_0) = \mathbf{u}(1)$ is the row vector of initial state probabilities and ${}_t\Gamma$ is the transition matrix at time t. Again, the multivariate Gaussian distribution is used to model the observations in each state, which means $\Pr(\mathbf{X}_t = \mathbf{x}_t|C_t = j) = \phi(\mathbf{x}_t|\mu_j, \Sigma_j)$ with

$\mu = (\mu_1, \ldots, \mu_m)$ and $\Sigma = (\Sigma_1, \ldots, \Sigma_m)$ being the collections of $1 \times d$ mean vectors and the $d \times d$ covariance matrices.

To model the transitions from one sleep state to the others an MLR is fitted for each state. The collections of MLR coefficients are denoted $\Lambda = [\Lambda_1, \Lambda_2, \ldots, \Lambda_m]$, where

$$\Lambda_i = \begin{bmatrix} \lambda_{i0_1} & \lambda_{i0_2} & \cdots & \lambda_{i0_m} \\ \lambda_{i1_1} & \lambda_{i2_1} & \cdots & \lambda_{im_1} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{i1_k} & \lambda_{i2_k} & \cdots & \lambda_{im_k} \end{bmatrix}$$

is the matrix of coefficients for transitioning from state i to any state. Where λ_{i0_j} is the intercept of the MLR pertaining to state i transitioning to state j. λ_{ij} is the $k \times 1$ column vector, $[\lambda_{ij_1}, \ldots, \lambda_{ij_k}]'$, of coefficients of the MLR for transitioning from state i to state j. For a given NHMM, $\theta = \{\mathbf{u}(\mathbf{y}_0), \Lambda = [\Lambda_1, \Lambda_2, \ldots, \Lambda_m], \mu, \Sigma\}$ is the set of parameters. The graphical representation of the transition from state i that depends on the MLR coefficients of state i is seen in Figure 3.E. The transition probability from state i to state j, evaluated for $\mathbf{y}_t$, is denoted

$${}_t\gamma_{ij} = \Pr(C_{t+1} = j | C_t = i, \mathbf{y}_t) = \frac{e^{\lambda_{i0_j} + \lambda_{ij}' \mathbf{y}_t}}{\sum_{s=1}^{m} e^{\lambda_{i0_s} + \lambda_{is}' \mathbf{y}_t}}$$

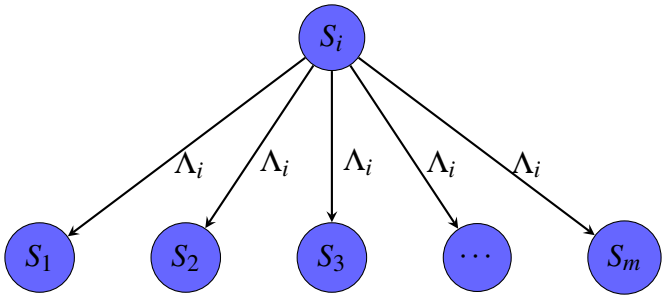


Figure 3.E: Graphical model showing the transition from state i to any state depends on the MLR coefficients for state i.

Statistically speaking, the MLR for each state is identifiable if different values of parameters generate different probability distributions of the observations. In or-

der to ensure identifiability of the transition probability parameters a base category must be chosen for the MLR of each state to ensure that the parameters learned for transitions are distinct. Setting λ_{i0_1} and λ_{i1} for $i \in S$, equal to zero, makes state 1 the base category in each MLR model.

$$\Lambda_i = \begin{bmatrix} 0 & \lambda_{i0_2} & \cdots & \lambda_{i0_m} \\ 0 & \lambda_{i2_1} & \cdots & \lambda_{im_1} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \lambda_{i2_k} & \cdots & \lambda_{im_k} \end{bmatrix}$$

Evaluation of Λ_i using $\mathbf{y}_t$ gives the i^{th} row of the transition matrix, at time t.

$${}_t\Gamma_i = \begin{bmatrix} \frac{1}{\sum_{s=1}^{m} e^{\lambda_{i0_s} + \lambda_{is}^{'}\mathbf{y}_t}}, & \frac{e^{\lambda_{i0_2} + \lambda_{i2}^{'}\mathbf{y}_t}}{\sum_{s=1}^{m} e^{\lambda_{i0_s} + \lambda_{is}^{'}\mathbf{y}_t}}, & \cdots, & \frac{e^{\lambda_{i0_m} + \lambda_{im}^{'}\mathbf{y}_t}}{\sum_{s=1}^{m} e^{\lambda_{i0_s} + \lambda_{is}^{'}\mathbf{y}_t}} \end{bmatrix}$$

with

$${}_t\Gamma = \begin{bmatrix} {}_t\Gamma_1 \\ {}_t\Gamma_2 \\ \vdots \\ {}_t\Gamma_m \end{bmatrix} = \begin{bmatrix} {}_t\gamma_{11} & {}_t\gamma_{12} & \cdots & {}_t\gamma_{1m} \\ {}_t\gamma_{21} & {}_t\gamma_{22} & \cdots & {}_t\gamma_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ {}_t\gamma_{m1} & {}_t\gamma_{m2} & \cdots & {}_t\gamma_{mm} \end{bmatrix}$$

Furthermore, when $\lambda_{i,j} = 0, \forall i, j \in S$, the transition probabilities become

$${}_t\gamma_{ij} = \frac{e^{\lambda_{i0_j} + \mathbf{0}\mathbf{y}_t}}{\sum_{s=1}^{m} e^{\lambda_{i0_s} + \mathbf{0}\mathbf{y}_t}}$$

Hence, the multinomial regression no longer depends on $\mathbf{y}_t$, meaning the transition to the next state is no longer dependent on $\mathbf{y}_t$ and the NHMM is reduced to the homogeneous model.

$${}_t\gamma_{ij} = \Pr(C_{t+1} = j | C_t = i, \mathbf{y}_t) = \Pr(C_{t+1} = j | C_t = i) = \gamma_{ij}$$

3.3.2 The Likelihood

The likelihood function for the NHMM is very similar to the basic HMM, but now the time dependence structure of the transition probabilities must be incorporated into the joint distribution of hidden states and observations. That is, $\mathbf{y}_0^{T-1}$ must be accounted for in the the transition probabilities. The covariates are incorporated into the notation with $\mathbf{y}^{(T-1)} = \mathbf{y}_0^{T-1}$. In the basic HMM setting, the joint distribution of states and observations is

$$\Pr(C^{(T)}, \mathbf{X}^{(T)}) = \Pr(C_1)\Pr(\mathbf{X}_1|C_1)\prod_{t=2}^{T}\Pr(C_t|C_{t-1})\Pr(\mathbf{X}_t|C_t)$$

and the likelihood is

$$\mathbf{L}_T = \mathbf{u}(1)\mathbf{P}(\mathbf{x}_1)\Gamma\mathbf{P}(\mathbf{x}_2)\Gamma\mathbf{P}(\mathbf{x}_3)\cdots\Gamma\mathbf{P}(\mathbf{x}_T)\mathbf{1}'$$

In the NHMM setting these become

$$\Pr(C^{(T)}, \mathbf{X}^{(T)}|\mathbf{y}^{(T-1)}) =$$

$$\Pr(C_1|\mathbf{y}_0)\Pr(\mathbf{X}_1|C_1)\prod_{t=2}^{T}\Pr(C_t|C_{t-1}, \mathbf{y}_{t-1})\Pr(\mathbf{X}_t|C_t)$$

and the likelihood becomes

$$\mathbf{L}_T = \Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}|\mathbf{y}^{(T-1)}) = \mathbf{u}(\mathbf{y}_0)\mathbf{P}(\mathbf{x}_1)_1\Gamma\mathbf{P}(\mathbf{x}_2)_2\Gamma\mathbf{P}(\mathbf{x}_3)\cdots_{T-1}\Gamma\mathbf{P}(\mathbf{x}_T)\mathbf{1}'$$

3.3.3 NHMM Forward and Backward Probabilities

Analogous to the homogeneous case, the definitions of the forward and backward probabilities and their recursions are essentially the same for NHMM. However, the covariates, $\mathbf{y}^{(T-1)}$, must be accounted for in the the forward and backward probabilities. The $1 \times m$ row vectors of forward probabilities, at time t, denoted α_t. Define

for each $t = 1, \ldots, T$,

$$\alpha_t = \mathbf{u}(\mathbf{y}_0)\mathbf{P}(\mathbf{x}_1)_1\Gamma\mathbf{P}(\mathbf{x}_2)_2\Gamma\mathbf{P}(\mathbf{x}_3)\cdots_{t-1}\Gamma\mathbf{P}(\mathbf{x}_t)$$

$$= \mathbf{u}(\mathbf{y}_0)\mathbf{P}(\mathbf{x}_1)\prod_{r=2}^{t}{}_{r-1}\Gamma\mathbf{P}(\mathbf{x}_r), \text{ where}$$

$$\alpha_1 = \mathbf{u}(\mathbf{y}_0)\mathbf{P}(\mathbf{x}_1) \text{ with } \alpha_t = \alpha_{t-1} \times {}_{t-1}\Gamma\mathbf{P}(\mathbf{x}_t), \text{ for } t = 2, \ldots, T$$

This results in

$$\alpha_t = \Pr(\mathbf{X}^{(t)} = \mathbf{x}^{(t)}, C_t|\mathbf{y}^{(t-1)}), \text{ and } \alpha_t(j) = \Pr(\mathbf{X}^{(t)} = \mathbf{x}^{(t)}, C_t = j|\mathbf{y}^{(t-1)})$$

Now, define the $m \times 1$ column vectors of backward probabilities for the NHMM as

$$\beta_t^{'} = {}_t\Gamma\mathbf{P}(\mathbf{x}_{t+1})_{t+1}\Gamma\mathbf{P}(\mathbf{x}_{t+2})\cdots_{T-1}\Gamma\mathbf{P}(\mathbf{x}_T)\mathbf{1}'$$

$$= \left(\prod_{k=t+1}^{T}{}_{k-1}\Gamma\mathbf{P}(\mathbf{x}_k)\right)\mathbf{1}' = {}_t\Gamma\mathbf{P}(\mathbf{x}_{t+1})\beta_{t+1}^{'} \text{ for } t = 1, \ldots, T-1$$

This results in

$$\beta_t^{'} = \Pr(\mathbf{X}_{t+1}^{T} = \mathbf{x}_{t+1}^{T}|C_t, \mathbf{y}_t^{T-1}) \text{ and } \beta_t^{'}(i) = \Pr(\mathbf{X}_{t+1}^{T} = \mathbf{x}_{t+1}^{T}|C_t = j, \mathbf{y}_t^{T-1})$$

3.3.4 Properties of Forward and Backward probabilities

Property 1 For $t = 1, \ldots, T$ and $i = 1, \ldots, m$

$$\alpha_t(i)\beta_t(i) = \Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}, C_t = i|\mathbf{y}^{(T-1)})$$

$$\text{and } \alpha_t\beta_t^{'} = \Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}|\mathbf{y}^{(T-1)}) = \mathbf{L}_T$$

Property 2 For $t = 1, \ldots, T$

$$\frac{\alpha_t(j)\beta_t(j)}{\mathbf{L}_T} = \Pr(C_t = j | \mathbf{X}^{(T)} = \mathbf{x}^{(T)}, \mathbf{y}^{(T-1)})$$

Property 3 For $t = 2, \ldots, T$

$$\frac{\alpha_{t-1}(i)_{t-1}\gamma_{ij}\phi(\mathbf{X}_t = \mathbf{x}_t | \mu_j, \Sigma_j)\beta_t(j)}{\mathbf{L}_T} = \Pr(C_{t-1} = i, C_t = j | \mathbf{X}^{(T)} = \mathbf{x}^{(T)}, \mathbf{y}^{(T-1)})$$

3.3.5 Estimation of NHMM Parameters

The estimation of parameters for an HMM in `depmixS4` is typically done "using the expectation-maximization (EM) algorithm or through the use of a general Newton-Raphson optimizer" (Visser & Speekenbrink, 2010, p. 4), with the Newton-Rhapson optimizer being used when there are linear constraints on the model parameters. This work does not enforce any constraints on parameters. The dynamic programming of the Forward algorithm and the Viterbi algorithm does not change from the homogeneous case, as the forward and backward probabilities and their properties are used in the same way.

Let $\theta = \{\mathbf{u}(\mathbf{y}_0), \Lambda = [\Lambda_1, \Lambda_2, \ldots, \Lambda_m], \mu, \Sigma\}$ denote the set of model parameters to be estimated, and $Q(\theta|\hat{\theta}) =$

$$E_{C^{(T)}|\mathbf{x}^{(T)},\mathbf{y}^{(T)},\hat{\theta}}\left[\sum_{j=1}^{m} {}_1\delta_j log(\mathbf{u}_j(\mathbf{y}_0)) + \sum_{t=2}^{T}\sum_{j=1}^{m}\sum_{k=1}^{m} {}_t\omega_{jk} log(\gamma_{jk}) + \sum_{j=1}^{m}\sum_{t=1}^{T} {}_t\delta_j log(p_{tj})\right]$$

The estimates for ${}_t\delta_j$ and ${}_t\omega_{jk}$ are found in the same way as the homogeneous case, but now incorporate the transition covariates.

$${}_t\hat{\delta}_j = E_{C^{(T)}|\mathbf{x}^{(T)},\mathbf{y}^{(T-1)},\hat{\theta}}[{}_t\delta_j] = \Pr(C_t = j | \mathbf{X}^{(T)} = \mathbf{x}^{(T)}, \mathbf{y}^{(T-1)}) = \frac{\alpha_t(j)\beta_t(j)}{\mathbf{L}_T}$$

$${}_t\hat{\omega}_{jk} = E_{C^{(T)}|\mathbf{x}^{(T)}\mathbf{y}^{(T-1)},\hat{\theta}}[{}_t\omega_{jk}] = \Pr(C_{t-1} = j, C_t = k | \mathbf{X}^{(T)} = \mathbf{x}^{(T)}, \mathbf{y}^{(T-1)})$$

$$= \frac{\alpha_{t-1}(j)_t \gamma_{jk} p_{tk} \beta_t(k)}{\mathbf{L}_T}$$

The maximization of the first two terms of $Q(\theta|\hat{\theta})$, however, is performed using a neural network approach in `depmixS4`. A neural network approach with m states in the input and output layers and no hidden layer, as seen in Figure 3.F, is used to calculate the parameters for $\mathbf{u}(\mathbf{y}_0), \Lambda = [\Lambda_1, \Lambda_2, \ldots, \Lambda_m]$. This is done using the `R` library `nnet` (Venables & Ripley, 2002). Conveniently, the softmax function (the MLR) is the transfer function between the input and output layers that calculates the transition probabilities. Thus, when the neural network estimates the parameters of the softmax transfer function for each node, it is estimating the MLR coefficients, $\Lambda = [\Lambda_1, \Lambda_2, \ldots, \Lambda_m]$, for modelling the transition probabilities. Furthermore, a neural network approach that employs the softmax function allows for estimation of "the probability distribution over class labels conditioned on the input" (Bridle, 1990, p. 229). Simply, it can estimate the posterior state probabilities given the covariates. Specifically, using the MLR in the output layer allows the neural network to estimate $\Pr(C_1|\mathbf{y}_0) = \mathbf{u}(\mathbf{y}_0)$.

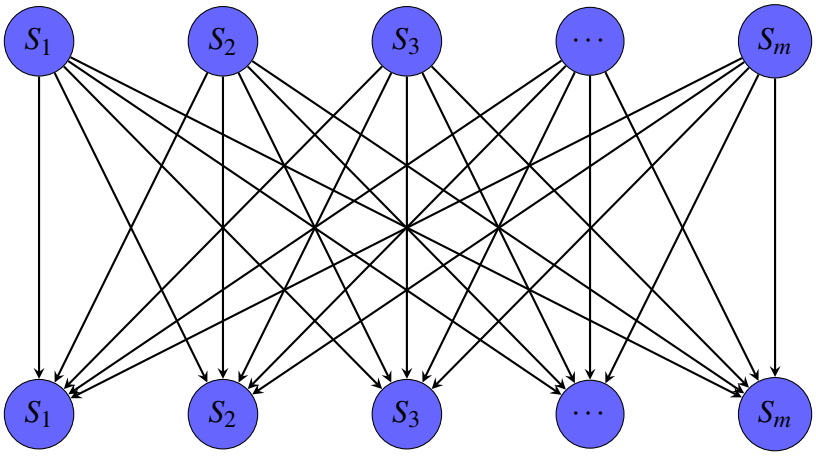


Figure 3.F: Neural network used to estimate MLR paramters. Only an Input (Top) layer and output layer (Bottom) are used, no hidden layer

The maximization of the third term of $Q(\theta|\hat{\theta})$ finds the maximum likelihood estimates of the state dependent parameters μ, Σ. This is done by the iteratively reweighted least squares algorithm, via the `glm` function. Visser and Speekenbrink

(2010, p. 5) state that "the expected values $[_t\delta_j]$ are used as prior weights of the observations". When the covariates are only used to model that transition probabilities, the estimates for μ and Σ are the same as the MLEs for the homogeneous HMM, since the emitted observation only depends on the current state. In total, there are $m + m(m-1)(k+1) + md + m(\frac{d^2+3d}{2})$ parameters to be estimated for each NHMM. The model selection performed for the HMM was repeated for the NHMM using 500 random starting points and selecting the model with the largest log-likelihood.

Chapter 4

The Data

4.1 The Data

The CF00N data presented in this work comes from a sleep study performed at the University of Alberta hospital from July 2015 to October 2017. The clinical study was approved by the Health Research Ethics Board of the University of Alberta (Pro0057638). There are 75 patient (58 male and 17 female) all night sleep-PSG recordings with a sampling rate of 512 Hz, along with a corresponding clinician event file that contains detailed apnea-hypopnea, arousals, and the sleep state labels of each epoch. Patients underwent an in laboratory PSG sleep study for diagnosis of obstructive sleep apnea (OSA) severity and treatment. All PSG recording were scored by the same sleep expert. Patient ages ranged from 2-18 years (41 patients $<$ 13 years, 34 patients $\geq$ 13 years) with a mean of 8.81$\pm$(4.52) years. For the purposes of pre-processing and cleaning the data 8 EEG electrodes (2 Frontal, 2 Central, 2 Occipital, 2 Temporal) , 2 EOG (Left, Right), 5 EMG (2 Chin, 2 Legs, 1 Abdomen), and 1 ECG channel were selected from each recording file. Tables 4.1 and 4.2 give the PSG channels information. The mean duration (standard deviation) of the PSG recordings is 9.02$\pm$(0.79) hours, which is approximately 1076 epochs for each patient, totalling 80,763 epochs for all patients. The proportions of sleep

states present in the CF00N data can be found in Table 4.3.

Name	Placement
F4M1/F3M2	Right/Left Frontal
C4M1/C3M2	Right/Left Central
T6M1/T5M2	Right/Left Temporal
O2M1/O1M2	Right/Left Occipital

Table 4.1: EEG electrodes selected from each patient.

Type	Name	Placement
EMG	EMG21/EMG31	Right/Left Chin
EMG	RLEG/LLEG	Right/Left Leg
EMG	ABD EMG	Abdomen
EOG	REOG/LEOG	Right/Left Eye
ECG	ECG	Abdomen

Table 4.2: EOG, EMG, and ECG electrodes selected from each patient.

States	Wake	NREM 1	NREM 2	NREM 3	REM
# Epochs	13,955	2,539	29,049	21,554	13,660
% Epochs	17.3	3.1	36	26.7	16.9

Table 4.3: Proportions of sleep states in the data for all 75 patients.

4.2 Pre-Processing the Data

Each PSG recording file contained the entire recording from when the equipment was turned on and calibrated until the end of the night when the equipment was turned off. Preparation of the data for pre-processing began with consulting the event files. The event files contained the start time for each recorded 30 second epoch, which allowed for extraction of the start and end times of the sleep state scored epochs. Epochs that contained bad electrode information from severe power line interference, bad electrode connection, or power outages were removed. Afterwards, a subset of each event file was made that selected only epochs with the expert scored sleep state. There was one patient CF069 that only had four distinct sleep states, presented in Table 4.4. After careful consideration, patient CF069

was included in this study on the basis that the information contained in their PSG recording was too valuable to not include the patient's epochs.

States	Wake	NREM 1	NREM 2	NREM 3	REM
# Epochs	639	35	119	261	0

Table 4.4: The distribution of epochs across sleep state for CF069.

The pre-processing stage began with a forward-backward filtering of the PSG signals through a digital Butterworth band-pass filter of order 3 in accordance with the AASM (Iber et al., 2007) electrode band-pass filtering frequencies: EEG (0.3 - 35 Hz), EOG (0.3 - 35 Hz), EMG(10 - 100 Hz), and ECG(0.3 - 70 Hz). The forward-backward filtering approach was used instead of just the forward approach to avoid a phase shift in the filtered signal. The filtered PSG data was then segmented into matrices of 30 second epochs. The sampling rate of 512 Hz meant the there were 15,360 observations recorded for each PSG channel in each epoch. The first eight columns of each epoch's PSG data was the four right EEG electrodes followed by the left four EEG electrodes. The last eight columns were the two EOG, five EMG, and one ECG electrodes. Once the filtering and segmentation was completed, the data was ready to be cleaned.

4.3 Cleaning of the Data

The PSG electrodes measure electrical activity of the brain from the surface of the skin, however, they can also measure electrical activity from other sources. When this happens the non-neural activity present in the EEG signal is called an artifact. Artifacts are split into two main types: physiologic and extra-physiologic. Physiological artifacts are caused by other systems in the subject's body that contaminate the EEG, such as ECG activity, eye blinks or movement, EMG interference caused by changing sleeping position, swallowing/clearing of throat, and sweat on surface of the skin at electrode site, to name a few. The extra-physiologic are caused by external sources, from the equipment or the environment, such as power line inter-

ference, bad electrode attachment, external stimuli that causes arousal, or medical equipment.

4.3.1 Independent Component Analysis for Artifact Removal

EEG Artifact removal itself is an area of active research in computing and diagnostic systems. There are many blind source separation approaches for separating neural activity and non-neural activity present in EEG signals. The simplest, most common, and highly effective one is the Independent Component Analysis (ICA). There are several variations of ICA (fastICA, InfoMax, and JADE) that have been implemented for artifact removal in the `eeglab` (Delorme & Makeig, 2004) MatLab plugin and `R` library `ica` (Helwig, 2018). The fastICA algorithm of Hyvarinen (1999); Hyvärinen and Oja (2000) was chosen for artifact removal for its simplicity and computational speed. Implementation of the fastICA algorithm in `R` was performed on each filtered epoch, $\mathbf{X}$, instead of the entire PSG recording, in order to reduce the computation time. Essentially, "[the] fastICA algorithm finds the orthogonal rotation matrix $\mathbf{R}$ that (approximately) maximizes the negentropy of the estimated source signals" (Helwig, 2018, p. 5). Where the source signals, $\mathbf{s}_1, \ldots, \mathbf{s}_C$, are the columns of the source matrix $\mathbf{S}$ and $\mathbf{S} = \mathbf{R}'\mathbf{X}$, with $rank(\mathbf{S}) \leq rank(\mathbf{X})$.

4.3.2 Source Signal Selection and Rejection

Despite fastICA's computational efficiency, one limitation of it is that the number of source signals is less than or equal to the number of observed signals. Thus, to maximize separation of neural and non-neural activity in the epochs of the CF00N data, all 16 selected PSG channels were included in order to maximize the number of source signals. Incorporating all 8 EEG electrodes helped to isolate the true neural activity present in all EEG electrodes into a few source signals while removing noise in others. Furthermore, the non-EEG channels that were already uncorrelated with the EEG were separated with ease and most of the variation in the EEG

caused by the non-EEG artifacts was captured by the source components pertaining to the non-EEG electrodes. The next step was to select source components that pertained to non-neaural activity and remove them. Automation of this source signal rejection process has been implemented previously in `eeglab` (Delorme & Makeig, 2004) Matlab software plug-ins, FASTER, Fully Automated Statistical Thresholding for EEG Artifact Rejection (Nolan, Whelan, & Reilly, 2010), ADJUST, Automatic EEG Artifact Detection based on the Joint Use of Spatial temporal Features (Mognon, Jovicich, Bruzzone, & Buiatti, 2010), and SASICA, Semi-Automatic Selection of Independent Components for Artifact correction in EEG (Chaumon, Bishop, & Busch, 2015). These algorithms use a wide variety of statistical features from the time, frequency, and time-frequency domains to select source components from the ICA decomposition as non-neural activity. In this work, all 16 source components were subjected to three rejection criteria. Once a source component, $\mathbf{s}_i$, was selected as non-neural activity, it was removed by replacing the i^{th} column of $\mathbf{S}$ with $\mathbf{0}$, a column of 0s. Of the total 80,763 epochs, only 2 epochs had all source signals selected for rejection. Hence, these epochs were rejected and not included in the analysis.

Criteria 1: Hurst Exponent

The Hurst exponent, named after Harold Edwin Hurst, who developed the method while studying hydrology of reservoirs (Hurst, 1951; Hurst, Black, & Simaika, 1965), is a measure related to the auto correlation function of a time series. It measures the long term dependency of a time series by examining the tendency of the times series to regress to the mean or trend in an extreme direction. According to Gneiting and Schlather (2004), the Hurst exponent takes values in the range 0 to 1 with values 0-0.5 indicating the time series behaves in a fashion that consecutive pairs alternate between high and low values (regress to the mean). A Hurst exponent of 0.5 - 1 indicates that consecutive pairs of observations are succeeded by values that are increasing in magnitude. Vorobyov and Cichocki (2002, p. 296) reported

"the Hurst exponent takes values between 0.70-0.76 for most human phenomena". Bian, Wang, Cao, and Zhang (2006) used a range of 0.7 - 0.9 to select source signals that represented neural information. The FASTER and SASICA algorithms employ the Hurst exponent for detection of source components containing artifacts. In this work, 10,654 epochs from 10 randomly selected patients of the CF00N data were analyzed to find appropriate Hurst exponent values for EEG and non-EEG activity. These ranges of Hurst exponent values were used to select source components that pertained to neural or non-neural activity. Figure 4.A and Table 4.5 present these findings.

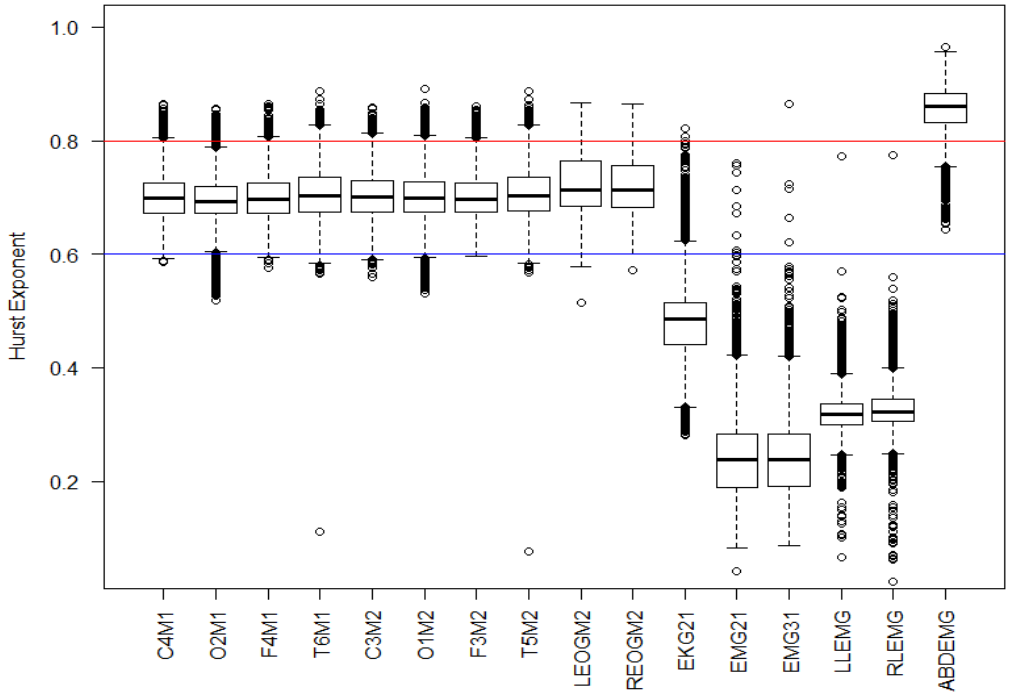


Figure 4.A: Hurst Exponents for Epochs of 10 patients across PSG channels. A total of 10,654 epochs used to asses Hurst exponent values for PSG signals.

In terms of selecting source components as neural activity, the range of the box plots of Hurst exponent values for the EEG electrodes were used, as this range captures the majority of epochs in the sample. The range was meant to be flexible and conservative, since a narrower range might remove too many source components

detected as non-neural activity, which can result in losing substantial amounts of neural information. On the other hand, too wide of a range will capture more non-EEG activity, which results in retaining more source components that contain artifacts. Specifically, for the CF00N data, source components with a Hurst exponent less than 0.6 or greater than 0.8 were selected as non-neural activity and removed, source components with a Hurst exponent below 0.6 were considered indicative of spontaneous non-neural activity or artifacts originating from the ECG, facial and leg EMGs, or spontaneous environmental activity during sleep, and source components with a Hurst exponent greater than 0.80 were considered persistent non-EEG activity coming from the abdominal EMG, that was caused by shifting in body position during sleep. However, source components that pertained to EOG or abdominal EMG activity that had a Hurst exponent in the acceptable range were not removed. Thus, the above facts leave a need for a second selection criteria to identify the source components that pertained to EOG and abdominal EMG activity.

Electrode Type	Range
EEG	0.6-0.80
ECG	0.35 - 0.65
EOG	0.60-0.85
Face EMG	0.1 - 0.45
Leg EMG	0.25 -0.35
Abdomen EMG	0.75-0.95

Table 4.5: Box plot range of Hurst exponent values of PSG channels.

Criteria 2: Correlation with Non-EEG signals

The second criteria was the source component's correlation, ρ, with all non-EEG channels. Correlation with EOG electrodes is used in ADJUST to identify components that contain ocular artifacts, which was extended to all non-EEG electrodes for the CF00N data. Source components that had a $|\rho| \geq 0.6$ with non-EEG electrodes were selected for rejection. This simple yet effective criteria helped identify source components with a Hurst exponent in the range 0.6 - 0.8 that pertained to non-neural activity from the EOG or abdominal EMG electrodes.

Criteria 3: Temporal Kurtosis

Lastly, the kurtosis of every component was calculated using

$$Kurt(\mathbf{X}) = \frac{1}{N}\frac{\sum_{i=1}^{N}(\mathbf{X}_i - \bar{\mathbf{X}})^4}{(\sum_{i=1}^{N}(\mathbf{X}_i - \bar{\mathbf{X}})^2)^2} - 3$$

where N is size of source component, $\mathbf{X}$, and $\bar{\mathbf{X}}$ is the mean of $\mathbf{X}$. Statistically speaking, this kurtosis is used to measure the tailedness of a probability distribution. The larger the kurtosis, the slower the exponential decay of the tails, which means that the distribution is more likely to produce outliers or extreme values. Therefore, source components that had a relatively larger kurtosis were more likely to have extreme values that pertained to non-neural activity containing artifacts. Bian et al. (2006, p. 722) used a "rejection threshold [for kurtosis] in terms of the number of standard deviation from the mean, *e.g.* 20%." The kurtosis criteria in this research used a much stricter rejection threshold to enforce a more conservative selection process. A source component with a kurtosis value greater than 3 standard deviations ($< 2\%$) from the mean kurtosis value of *all* source components was selected for rejection.

4.3.3 Reconstructing the Cleaned EEG Data

Once source components were selected for rejection, the columns of $\mathbf{S}$ that pertained to those source components were replaced with columns of zeros to form the new source signal matrix $\tilde{\mathbf{S}}$. Recall that the fastICA algorithm finds the matrix of source signals $\mathbf{S}$ and the rotation matrix $\mathbf{R}$ such that the independence between source signals is maximized and $\mathbf{S} = \mathbf{R}'\mathbf{X}$. In order to reconstruct the clean PSG signals, denoted $\tilde{\mathbf{X}}$, replace $\mathbf{S}$ by $\tilde{\mathbf{S}}$ and multiple on the left side by $\mathbf{R}^{-1}$ to get $\tilde{\mathbf{X}} = \mathbf{R}^{-1}\tilde{\mathbf{S}}$. The non-EEG channels are reconstructed, but are essentially non-existent (a flat line, y=0), as these source components were replaced by a column of 0s. The first 8 columns of $\tilde{\mathbf{X}}$ are the artifact free EEG signals and the last 8 being the non-existent EOG, EMG, and ECG signals. These non-existent signals were

simply removed and replaced with the original filtered non-EEG signals. See Figure 4.B for an example of the C4M1 EEG electrode before and after ICA cleaning presented in this work. Notice the reduction in amplitude of the artifact in the green boxes and the reduction of noise over the entire epoch.

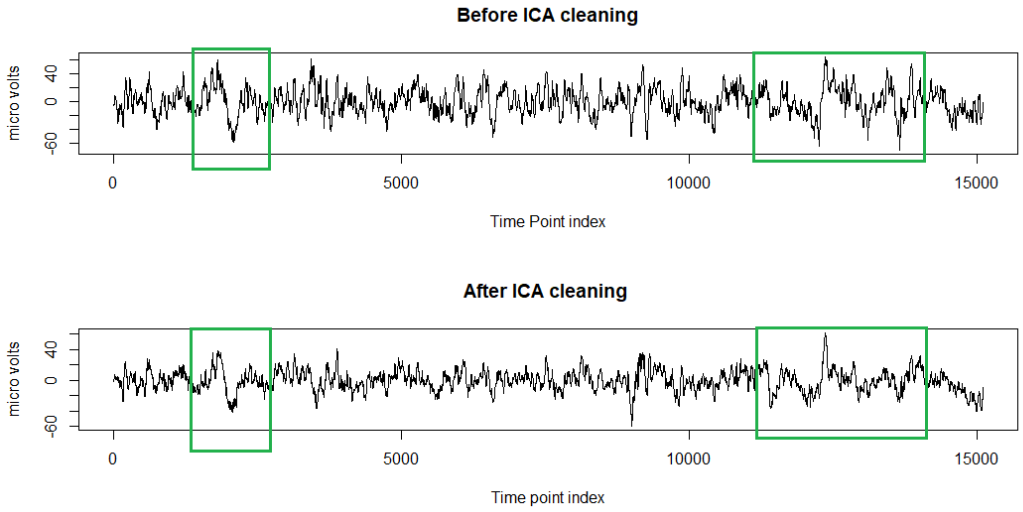


Figure 4.B: Comparison of C4M1 electrode before and after ICA cleaning.

4.4 Feature Extraction

In this work three sets of PSG features were calculated. The first two sets of EEG features were that of Fraiwan et al. (2012) and da Silveira et al. (2017), as these feature sets were found to be quite successful for classification of sleep states. These two methods decomposed a single EEG channel via wavelet transformations into the time-frequency domain and extracted statistical measures from the EEG frequency bands listed in Table 4.6. The third feature set was calculated using the non-EEG channels, as non-EEG information is used by sleep experts to score sleep states. In total, 40 statistical features were extracted from every epoch.

4.4.1 The Wavelet Transformation

EEG is considered a non-stationary process, as its statistical moments tend to change as the patient progresses through sleep cycles, so the time and frequency domain statistics retrieved can be somewhat limited, but still useful in conjunction with time-frequency domain statistics. A popular method of time-frequency analysis used for extraction of statistical information in EEG signals is the wavelet transformation. There are two main types: continuous (CWT) and discrete (DWT).

The CWT

The CWT is a mathematical tool that provides a complete representation of the time series $x(t)$ in the time-frequency domain. This is done by convolution of $x(t)$ with a mother wavelet $\Psi(t)$ to produce W, the matrix of wavelet coefficients. The $\Psi(t)$ has a scale parameter, a ($a \neq 0$), that dilates or compresses $\Psi(t)$ to find frequency domain information of $x(t)$. The scaling parameters were chosen to cover the entire EEG frequency range of 0.3 - 35Hz. The a value pertains to a specific EEG frequency, f, found using the formula $a = \frac{f_c f_s}{f}$. Where f_c is the center frequency of $\Psi(t)$ and f_s ($f_s = 512Hz$ for CF00N data) is the sampling frequency of $x(t)$. The translation parameter, b, is used to shift $\Psi(t)$ across the time domain. Thus, when a and b vary on $\mathbb{R}^+$, the result is the complete representation of $x(t)$ in the time-frequency domain. For all pairs of a and b, the result of the convolution is a matrix of wavelet coefficients, denoted $W_{a,b}$. The wavelet coefficients are essentially the amplitude of the convolution and interpreted as how similar the EEG signal is with the wavelet for specified values of a and b.

$$W_{a,b} = \int x(t)\Psi_{a,b}(t)dt \text{ with}$$

$$\Psi_{a,b} = \frac{1}{\sqrt{|a|}}\Psi\left(\frac{t-b}{a}\right) \text{ is the dilated and shifted wavelet.}$$

The DWT

The DWT consists of repeatedly passing the signal $x(t)$ through a series low-pass, $g(t)$, and high-pass, $h(t)$, filters, which can be seen in Figure 4.C.

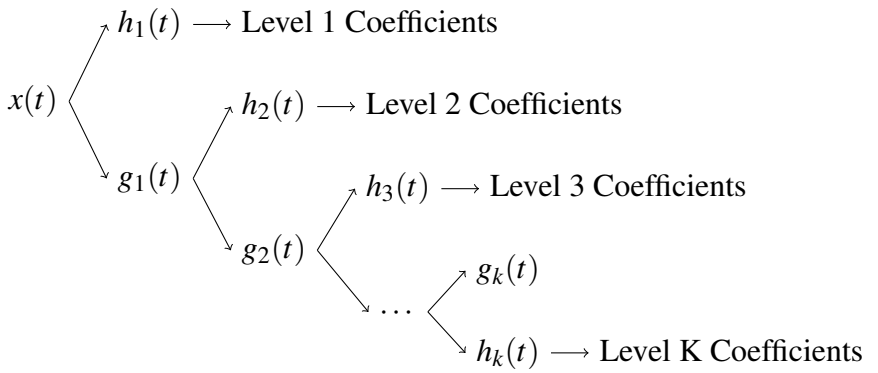


Figure 4.C: DWT decomposition diagram. At the i^th level of the decomposition, the approximation coefficients of $g_{i-1}(t)$ are down sampled by a factor of 2.

The convolution of $x(t)$ with $g(t)$ filters out frequencies higher than what is specified by $g(t)$. The result gives the approximation coefficients that are the lower frequency representation of $x(t)$. The convolution with $h(t)$ gives the detail, or the wavelet coefficients. This process removes the upper half of the frequency range for $x(t)$ at each level of decomposition in accordance with the Nyquist-Shannon (Shannon, 1948) Sampling Theorem. That is, for a signal $x(t)$ with a largest frequency, f_s, the sampling rate of $x(t)$ must be at least $2f_s$, which means that, at the first level of decomposition for the EEG signals, in the CF00N data, the largest observable frequency is 256Hz, since the sampling rate is 512Hz. Furthermore, at each level of decomposition half of the approximation coefficients can be discarded before passing them through the next set of filters. For a signal $x(t)$ of length L where $L = P2^k$ ($k,\ P \in \mathbb{N}$), and a sampling frequency f_s, it is possible to calculate at most k levels of wavelet coefficients. Analogous to the CWT, the j^{th} level detail coefficients pertaining to the frequency range $f_s/2^j$ to $f_s/2^{j-1}$ are the result of the convolution of $x(t)$ with $\Psi_{a_j,b}$, where $a_1 = f_s/2,\ a_2 = f_s/4, \ldots,\ a_j = f_s/2^j$ and

$$\Psi_{a_j,b} = \frac{1}{\sqrt{2^j}} \Psi \left(\frac{t - b2^j}{2^j} \right) \text{ with } j = 1 \ldots, k$$

4.4.2 Feature Set 1

The first feature set calculated was that of Fraiwan et al. (2012). The EEG signal, C4M1, was decomposed into seven frequency bands, as seen in Table 4.6, via the CWT using the Daubechies wavelet of order 20 as the $\Psi(t)$. There are seven EEG frequency bands used by Fraiwan et al. (2012) to cover the range of 0.3Hz to 35Hz.

Waveform	Frequency Range (Hz)	Sleep State
δ Delta	0.5-4	NREM 3
θ Theta	4-8	NREM 1, REM
α Alpha	8-13	Wake, REM
β_1 Beta 1	13-22	Wake, REM
β_2 Beta 2	22-35	Wake, REM
Sleep Spindles	12-14	NREM 2
K-Complexes	0.5-1.5	NREM 2

Table 4.6: Fraiwan et al. (2012) uses these seven primary EEG frequency bands and sleep state(s) where each characteristic waveform is dominant

Once the CWT of an epoch was performed, the rows of the $\mathbf{W}_{a,b}$ pertaining to the seven specified frequency bands are used to calculate the Renyi entropy of the wavelet coefficients. This results in 7 statistical features for each epoch.

Renyi Entropy

In information theory, the measure of entropy is the expected rate at which information is emitted by a stochastic process. The information associated with each event that occurs can be viewed as a random variable and the information entropy, or Shannon entropy, as the expected value. Mathematically, entropy is defined as

$$En = -\sum_{k=1}^{n} p_i log_2(p_i)$$

"where p_i is the histogram distribution of the time-frequency coefficients with n bins" (Fraiwan et al., 2012, p. 13). The entropy is used to measure the randomness of the wavelet coefficients in a specified frequency range. The Renyi entropy is

defined as

$$\text{R Ent} = \frac{1}{1-\alpha} log_2 \left(\sum_{i=1}^{n} p_i^{\alpha} \right)$$

where p_i and n are the same as before, but now α is called the order of Renyi entropy. In this work and in the study by Fraiwan et al. (2012) $\alpha = 2$.

For each epoch in the CF00N data, a histogram with n=5000 bins was used to estimate the probability density of the wavelet coefficients in each frequency band. This was done by first calculating $W_{a,b}$ of all epochs for each patient. Then for each patient, the maximum and minimum wavelet coefficient values of all epochs are collected in each of the seven frequency bands listed in Table 4.6. This was done to find the lower and upper boundaries of each frequency band, for each patients epochs, so that the Renyi entropy of each epoch was calculated using histograms with the same boundaries. For each patient the lower boundaries in each frequency band were set to the 0.025 quantile of all the minimum values for that frequency band. Similarly for the upper boundaries, the .975 quantiles of all the maximum values in each frequency band were used. This ensured the histogram distribution for every epoch in each frequency band had large (and small) enough boundaries, but was censored to outlier coefficient values, and the Renyi entropy values calculated for all epochs were done on the same domain for that patient.

4.4.3 Feature set 2

The features from da Silveira et al. (2017) used the DWT with a Daubechies mother wavelet of order 2 on a single EEG channel and these same features were calculated in this work using the C4M1 EEG channel. The sampling rate of the CF00N data is 512 Hz with $15,360 = 2^{10} \times 15$ observations in each 30s epoch, which meant that the C4M1 signal could be decomposed, using the DWT, into at most 10 levels. However, only 8 levels were required to cover the desired EEG frequency range for sleep state classification. Table 4.7 provides the range of frequencies covered by each level of the DWT in this work.

Level	Range in Hz	Number of coefficients
D_1	128-256	7,680
D_2	64-128	3,640
D_3	32-64	1,820
D_4	16-32	910
D_5	8-16	455
D_6	4-8	228
D_7	2-4	114
D_8	1-2	57
C_8	0-1	29

Table 4.7: Frequency ranges and number of observations for the DWT, by levels.

The first two levels of coefficients are ignored, as the pre-processing of the data filtered out frequencies above 35 Hz. Then, for the remaining levels, $D_3, \ldots, D_8$, and C_8, the 2nd, 3rd, and 4th statistical moments of the coefficients were calculated. C_8 is the approximation, or the scale coefficients from the 8^{th} level of DWT decomposition, the representation of the D_7 coefficients in the 0-1Hz frequency range. Thus, 21 statistical features were extracted from every epoch. The statistical moments were calculated using the following:

$$Var(\mathbf{X}) = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{X}_i - \bar{\mathbf{X}})^2 \qquad Skew(\mathbf{X}) = \frac{\frac{1}{N}\sum_{i=1}^{N}(\mathbf{X}_i - \bar{\mathbf{X}})^3}{Var(\mathbf{X})^{3/2}}$$

$$Kurt(\mathbf{X}) = \frac{1}{N}\frac{\sum_{i=1}^{N}(\mathbf{X}_i - \bar{\mathbf{X}})^4}{Var(\mathbf{X})^2} - 3$$

4.4.4 Feature set 3

The original filtered non-EEG signals were used to construct the last set of features. The presence of EOG or EMG information helps sleep experts discriminate between states of active sleep (REM), non-active sleep (NREM 3), and wake states. To incorporate non-EEG information, the 2nd, 3rd, and 4th statistical moments of LEOG, REOG, EMG21, and ECG PSG channels were calculated. The moments were calculated using the same equations for the second set of features, resulting in 12 statistical features for each epoch.

Chapter 5

Random Forest Feature Selection

5.1 Random Forest for Feature Selection

5.1.1 Assessing Multivariate Normality of Features

To begin the feature selection, the statistical features extracted from all epochs were combined into a larger data set and then each individual statistical feature was assessed for univariate normality within each sleep state. Several features were extremely right skewed and first required a transformation via the natural logarithm to approximate univariate normality within sleep states. Features that were still extremely skewed, or presented excessive multimodality within sleep states, were not included in the feature selection. Density and Quantile-Quantile (QQ) plots for the individual statistical features are in the Appendix. For ease of reference, the Appendix A.1 is for the Renyi entropy features, A.2 is for the discrete wavelet coefficient (DWC) features, and A.3 is for the non-EEG features. The next part of the feature selection was to examine the cross validation error of each patient's epochs using the different number of randomly selected features to build the trees in each random forest, which determined the minimum number of features required to achieve minimal cross validation error for all patients. The third part was determining which features produced the lowest cross validation error for all

patients. Finally, the selected features were then examined for multivariate normality within sleep states using Chi-square QQ plots of the Mahalanobis distance between epochs. The goal of the feature selection was to find a set of features, for all patients, which are then used in the HMM and NHMM analysis.

5.2 Univariate Normality Within Sleep States

5.2.1 Renyi Entropy Features

The Renyi entropy features were the most promising. Although, some of the univariate distributions within specific sleep states appeared to have slightly heavy tails, which indicates that there may be outliers present, but examination of the QQ plots showed that the majority of data quantiles matched those of a univariate normal distribution. The density and QQ plots of the k-complex feature are presented in Figures 5.A and 5.B, while the plots for the remaining Renyi entropy features can be found in A.1. The possible outliers could be from a single patient as a result of a poor recording, or possibly many patients caused by missed artifacts in the cleaning process. Despite the slightly heavier tails and potential outliers, the data did not provide strong evidence against the assumption of the Renyi entropy features coming from Gaussian distributions, within each sleep state. Therefore all the Renyi entropy features were included in the feature selection.

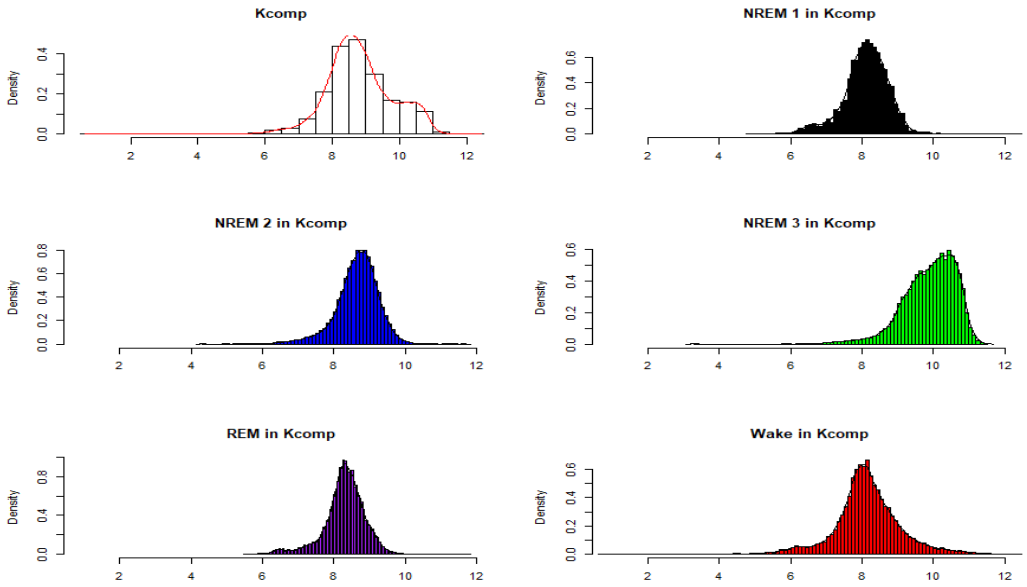


Figure 5.A: Density plots for Renyi entropy in the K complex band, overall and within sleep states. In each sleep state there appear to be influential values making the left tails heavier.

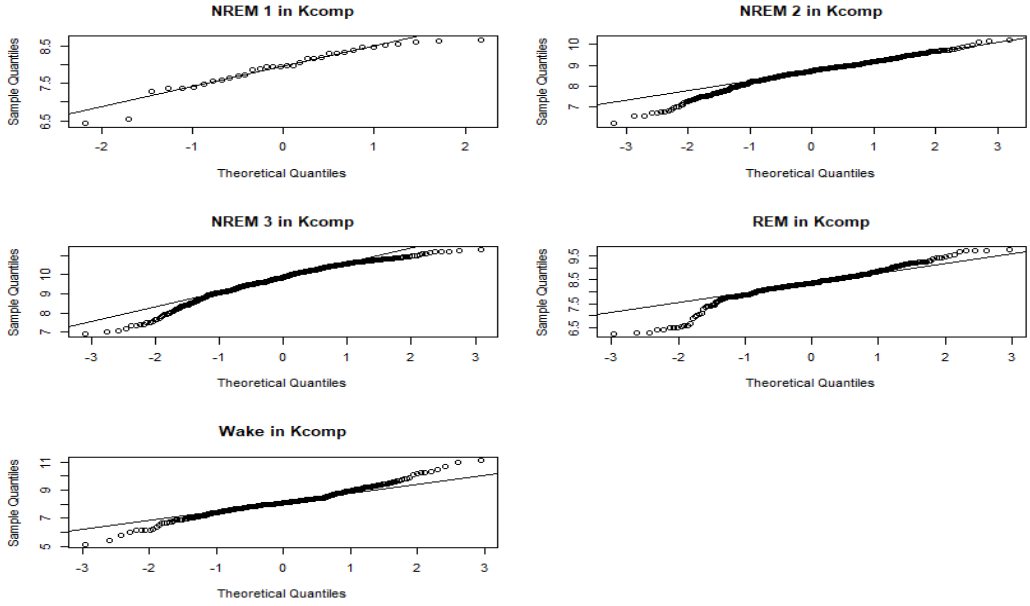


Figure 5.B: Quantile-Quantile plots of Renyi Entropy in the K complex frequency band. In all plots the lower quantiles deviate below the theoretical quantiles of the univariate Gaussian, indicating a heavier left tail. Overall, the majority of actual quantiles does agree with the theoretical quantiles.

5.2.2 The Discrete Wavelet Coefficient Features

The variance and kurtosis features all required a transformation with the natural logarithm, however, the kurtosis features remained extremely skewed and were not included in the feature selection. On the other hand, all of the variance features appeared symmetric and their QQ plots showed that within each sleep state the majority of data quantiles matched the theoretical quantiles of a Gaussian distribution. However, in the wake state, the variance features had a slightly heavier right tail, but not heavy enough for the variance features to be excluded from the feature selection. The skewness features did not require the logarithm transformation and had QQ plots that showed that within each sleep state the majority of data quantiles matched the theoretical quantiles of a Gaussian distribution. Therefore, the data did not provide strong evidence against these features coming from a Gaussian distribution. However, a closer inspection of the skewness features QQ plots showed these features had very small variance, since the line of agreement between quantiles was almost horizontal. These variables were included, but with such small variances the author of this thesis had reservations about their ability to discriminate effectively between sleep states.

5.2.3 The Non-EEG Features

As in the DWC features, the variance and kurtosis features of the non-EEG data required the logarithm transformation. There were seven non-EEG features that made it into the feature selection: variance of EMG, variance of REOG, variance of LEOG, skewness of REOG, skewness of LEOG, Kurtosis of REOG, and Kurtosis of LEOG. The variances of EMG, REOG, and LEOG had slightly heavier right tails, but were considered overall suitable. The skewness of REOG and LEOG showed the same properties as the skewness features of the DWC. The Kurtosis of REOG and LEOG are not quite symmetric but in each sleep state the data quantiles closely matched those of a univariate Gaussian distribution. Again, the lack of symmetry suggested there may be possible outliers that would affect state dependent parame-

ters in the HMM and NHMM.

Overall, 28 (7 Renyi, 14 DWC, 7 NonEEG) of the original 40 features were used in the random forest feature selection process. Although the presence of outliers and heavier tails for most of the features was a bit concerning, as in each sleep state the mean and covariances would be affected, and consequently the classification performance for the HMMs and NHMMs. For each patient these 28 features were collected into a separate data set for that patient, which were then used to find the optimum number of features for all patients.

5.2.4 Finding the Optimal Number of Features

The `R` library `randomForest` (Liaw & Wiener, 2002) allowed for calculation of k-fold cross validation prediction error, for each patient, for each number of randomly selected input features used to grow trees in the random forest. In this work, a 10-fold cross validation of each patient's 28 included features data was used to calculate the mean cross validation error of the 10 folds across each possible number of random selected features for that particular patient. For each fold, a random forest was of 10,000 trees was constructed, using the predetermined number of randomly selected features to build each tree. That is, each feature had equal probability of being used to construct each tree. This allowed the author of this thesis to find the minimum number of features, r, such that cross validation error was minimal for all patients. The box plots of the cross validation errors for all patients across each possible r is presented in Figure 5.C. The first r value that captures the mean cross validation error for all patients within the range of the box plot is $r = 9$. Furthermore, when $r = 9$ it ensures at least 50% of patients had at most a 20% mean cross validation error. This is also true for $r = 6, 7, 8$, but there were patients with higher mean cross validation errors outside the range of the box plot for those r values, which indicates a less consistent classification performance for all patients. The significance of performing the above analysis was that in order to find which features should be selected, the author of this work forced the random

forest algorithm in the next part to use $r = 9$ randomly selected features to build the trees, since by default the algorithm will choose $r = 5 = \lfloor \sqrt{28} \rfloor$.

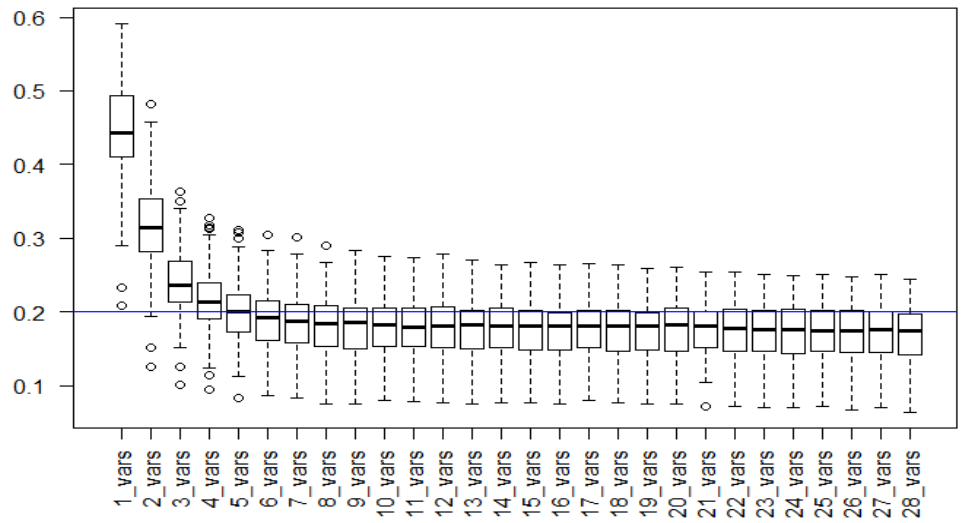


Figure 5.C: Box plots of patient cross validation prediction error VS the number of features used to build each tree in the random forest.

5.2.5 Finding the Optimal Features

Fortunately, the `randomForest` (Liaw & Wiener, 2002) library also supports measuring of variable importance when constructing a random forest for each patient's 28 included features data. The variable importance calculated "is the increase in percent of times a case is [Out-of-bag] and misclassified when the variable is permuted" (Liaw & Wiener, 2002, p. 20). Specifically, for each variable there is a variable importance measure for each class (sleep state), where for each feature the "class-specific measures [are] computed as mean decrease in accuracy" (Liaw & Wiener, 2002, p. 19). Extraction of variable importance measures for each patient in each sleep state was done by constructing a large random forest of 10,000 trees with $r = 9$ randomly selected Gaussian features to build each tree. Each feature had equal probability of being chosen for construction of a single tree so that each

feature would be used in tree construction equally often. Even though there are on average 1076 epochs for each patient, the number of trees was chosen to be 10,000 to ensure that each epoch was predicted at least a few times, since the proportions of sleep states were drastically imbalanced.

Once the variable importance measures for each patient were collected, an analysis of variable importance in each state was performed. This was done by recording the 14 most important features of for sleep state for each patient. Then in each sleep state a tally of the top 14 features for all patients was recorded. Essentially, this tally counted the number of patients a feature was in the top 14 features of each sleep state. This was done to determine the top 14 features in each state that was found important to at least 50% of patients. For example in the NREM 3 state, it can be seen in Figure 5.D that 13 features were found to be important to at least 50% of patients. The top important features for the NREM 3 state were all of the Renyi entropy features, and the variances of DWCs in levels 3 to 7. The importance plots for the other sleep states can be found in A.4.

The features that were important to at least 50% of patients in each state were then further tallied to determine the number of states where each variable was found to be important. This tally can be seen in Figure 5.E. The final selected features were the ones that were found to be important to 50% or more patients in every sleep state. In total there are nine final selected features the Renyi entropy features: K-complex, Delta, Theta, Alpha, and Sleep spindle frequency bands. The last four features selected were the variances of the DWCs in: Level 4, Level 7, Level 8, and Scale Coefficient frequency bands. On a side note, none of the skewness features of the DWC and Non-EEG made it into the final features, as was expected.

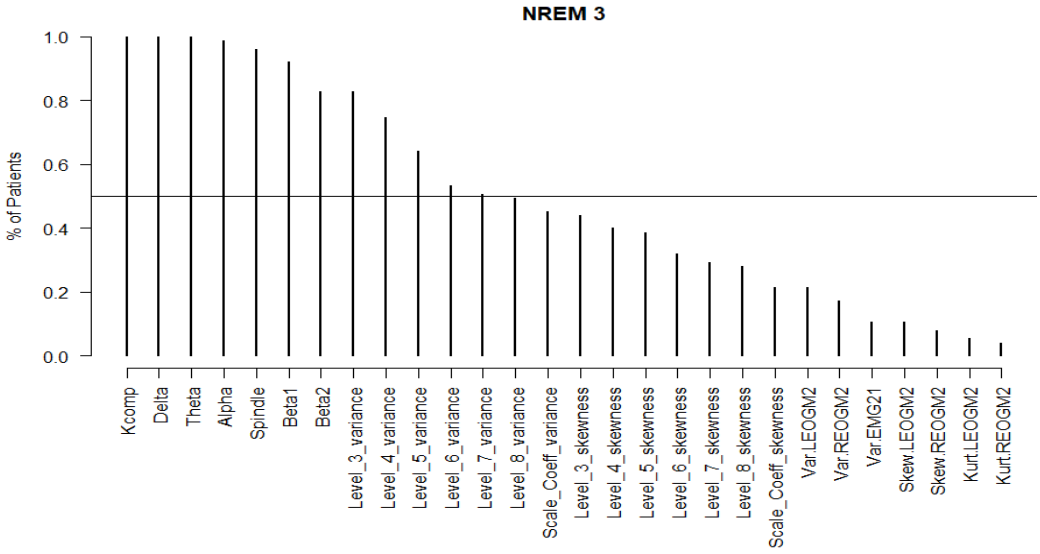


Figure 5.D: The percentage of patients where each feature was found to be in the top 14 features for NREM 3.

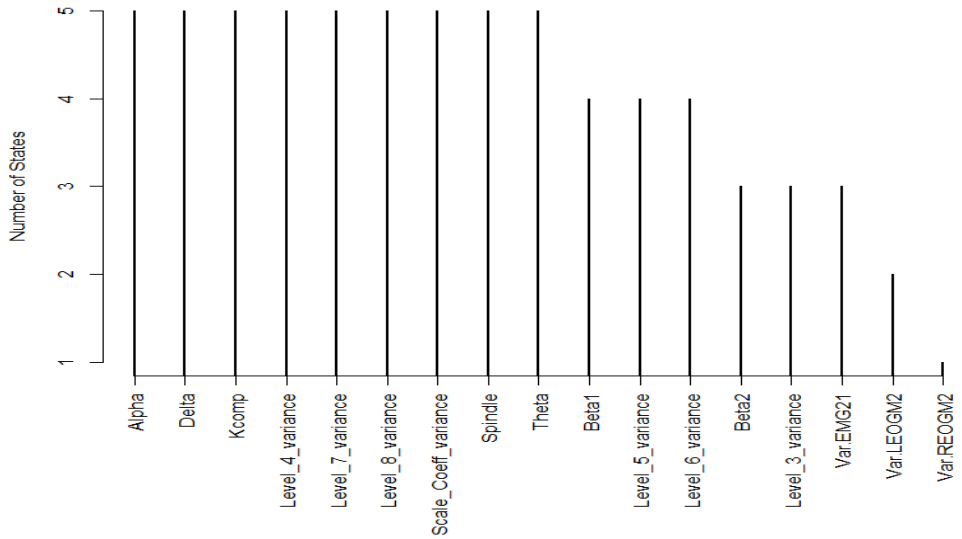


Figure 5.E: Number of states in which the feature was found to be in the top 14 for majority of patients. There are 9 of 17 potential final features that are important in all sleep states.

5.2.6 Data for HMM and NHMM

The data did not provide strong evidence that the marginal distributions of the Renyi entropy features and final selected features were not from univariate normal distributions within the sleep states, however, the combined feature sets needed to be

assessed for multivariate normality. This was based on the Mahalanobis distance between observations, since "[when] the parent population is multivariate normal and both n and $n-p$ are greater than 25 or 30, each of the square differences $d^2_{(1)}, d^2_{(2)}, \ldots, d^2_{(n)}$ should behave like a chi-square random variable [with p degrees of freedom]" (Johnson & Wichern, 2007, p. 184). Graphing the ordered distances with the ordered theoretical quantiles of the chi-square distribution forms the Chi-square QQ plot and if the data points should form a straight line through the origin with a slope of 1, then the data is multivariate normal.

$$\text{Mahalanobis Distance for the } i^{th} \text{ epoch, } d_i = (\mathbf{x}_i - \bar{\mathbf{x}})' \mathbf{S}^{-1} (\mathbf{x}_i - \bar{\mathbf{x}})$$

An assessment of each patient's data sets individually, across states, would require far too many plots to be included, even in an appendix. Therefore, the Renyi entropy and final features data sets from all patients are combined into their respective larger data sets, however, calculating distances of 80,763 epochs is simple enough, but plotting them is not feasible. In this work, 10,000 epochs were randomly selected with the distances calculated and then plotted in Figures 5.F and 5.G. Note that the same random 10,000 epochs used to assess the Renyi entropy data were also used for the final features data. For both data sets, the distances and quantiles in the NREM 1 state line up slightly underneath the line of slope 1, but do not form any strongly deviating non linear pattern. The distances and quantiles in the REM state line up well on the line with slope 1, but there appears to be small amount of evidence of a nonlinear pattern, which could mean that the features in the REM state are slightly skewed. On the other hand, for all other states the distances and quantiles match nicely with the exception of a few possible outliers. Overall, the data does not provide strong evidence that the renyi and final selected features do not come from a multivariate normal distribution.

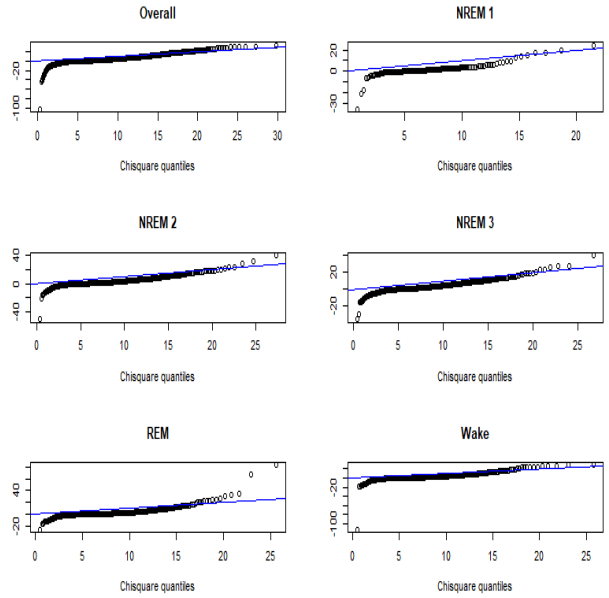


Figure 5.F: Chi-square Quantile-Quantile plots to assess multivariate normality of Renyi entropy features.

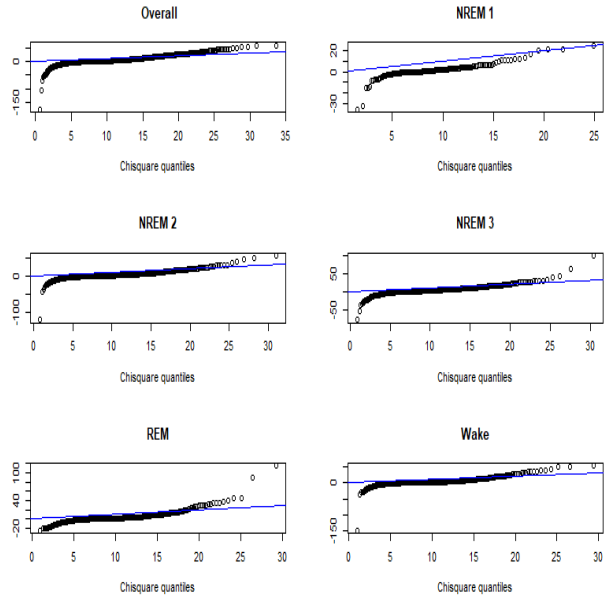


Figure 5.G: Chi-square Quantile-Quantile plots to assess multivariate normality of selected final features.

Chapter 6

Random Forest Analysis

6.1 Analysis & Comparison

Analysis was performed using the random forests of Fraiwan et al. (2012) and da Silveira et al. (2017) with 64-trees and an extension to 128-trees across all four feature sets. For comparison with the literature, the 10-tree random forest of Fraiwan et al. (2012) on the Renyi entropy features was also explored. There will be 3 settings for comparison: a random split of (1/3) testing and (2/3) training, 10-fold cross validation, and leave-one-patient-out-cross-validation (LOPOCV). The first two settings are for comparison with the respective literature, Fraiwan et al. (2012), and da Silveira et al. (2017). The importance of the LOPOCV analysis is that for an algorithm to be successful in clinical practice it must only use information from previous patients to predict the sleep states of a new patient. The algorithm cannot use information from the new patient in the model learning process. A comparison with Boostani et al. (2017) who used a LOPOCV in their study, but only used 10 tree random forests, is discussed at the end of the chapter. In each setting, the random forests with 64 and 128 trees were grown multiple times using the training data and then predicted the test data. Due to computational time restrictions, classification of sleep states in each setting was repeated for 100 trials, though the ideal

number of trials would have been at least 1000. Classification performance of the random forests presented here is assessed using classification accuracy and Cohen's κ (Cohen, 1960).

6.1.1 Setting 1: Random (1/3) Testing (2/3) Training

For each of the four feature sets analyzed the epochs from all patients are combined into a complete feature set. Then, a random 1/3 of the data is selected for testing purposes. In each trial, the random split uses the same epochs in the testing and training sets for each feature set analyzed. This was done to ensure a fair comparison across feature sets in each trial.

Study	Data Set	Feature Set	Accuracy (%)	κ
Fraiwan et al. (2012)	*Sleep-EDF 2002*	Renyi	82.57	0.76
This Work	CF00N	Renyi	70.93 (.3)	0.60(0.003)

Table 6.1: Classification performance comparison of 10-tree random forest using Renyi entropy of CWT coefficients features.

Table 6.1 shows a comparison of classification performance between the 10-tree random forests constructed using Renyi Entropy of CWT coefficients as features. The random forest performance on the CFOON data performs markedly lower in mean classification accuracy, $\approx 12\%$, and κ values, 0.16, than in Fraiwan et al. (2012). Overall, the performance of the 10-tree random forest for CF00N provides a great baseline to build from as more than 70% of epochs in the testing data are classified correctly. Figures 6.A and 6.B show that the mean classification accuracy using the Renyi Entropy features improved by $\approx 2\%$, when the number of trees in the random forest increased from 10 to 64.

Data Set	Features	# features	Accuracy (%)	κ
Sleep EDF Expanded	DWC Moments	18	90.8	-
CF00N	Renyi	7	73.32(0.2)	0.63(0.003)
CF00N	DWC Moments	21	75.02(0.2)	0.66(0.003)
CF00N	Non EGG	12	58.41(0.3)	0.41(0.004)
CF00N	Final	9	76.49(0.2)	0.68(0.003)

Table 6.2: 64 tree random forest performance using random (1/3) testing (2/3) training data. Values reported are the mean accuracy of the 100 trials with standard deviation in brackets. DWC: Discrete Wavelet Coefficients.

Data Set	Features	Accuracy (%)	κ
CF00N	Renyi Entropy	73.57(0.2)	0.64(0.003)
CF00N	DWC Moments	75.33(0.2)	0.66(0.003)
CF00N	Non EGG Moments	59.26(0.3)	0.42(0.005)
CF00N	Final Selected	76.68(0.2)	0.68(0.003)

Table 6.3: 128 tree random forest performance using random (1/3) testing (2/3) training data. Values reported are the mean accuracy of the 100 trials with standard deviation in brackets. DWC: Discrete Wavelet Coefficients.

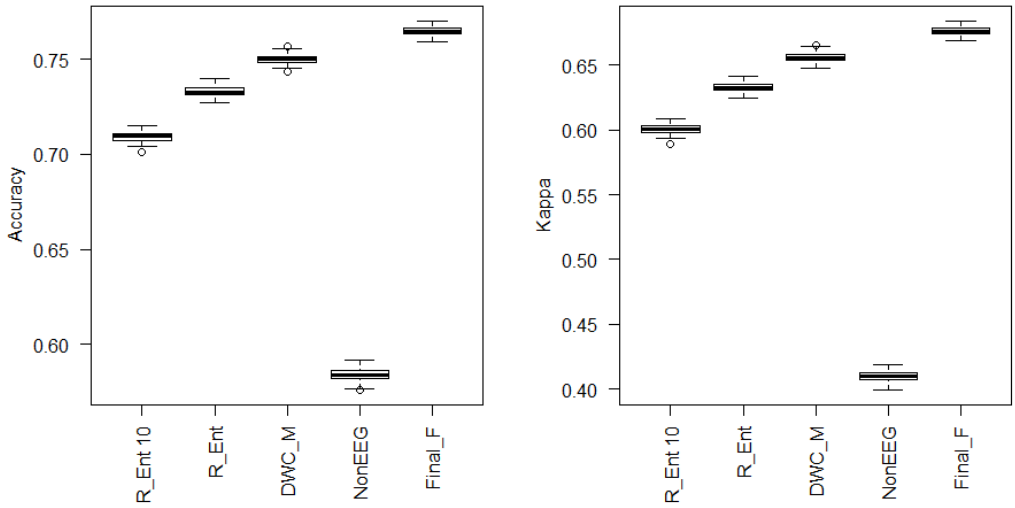


Figure 6.A: Box plots of 10- and 64- tree random forest classification accuracy and Cohen's κ for 100 trials, using random (1/3) testing, (2/3) training. R Ent is Feature Set 1, DWC M is Feature Set 2, NonEEG is Feature Set 3, Final F is Final Selected Features Set

In Table 6.2, classification performance with da Silveira et al. (2017) on the *Sleep EDF Expanded* data shows the 64-tree random forest constructed for the CF00N data in this study also underperformed. The same can be seen in Table 6.3 for the 128-tree random forests. However, in both the 64 and 128-tree random forests that used moments of the discrete wavelet coefficients (DWC) features performed better than ones that used the Renyi entropy features. The random forests that used the non-EGG features performed the worst. The accuracy of these random forests decreased between 14 - 17% and κ decreased by at least 0.22.

The performance results on the CF00N data are lower than those reported by Fraiwan et al. (2012) and da Silveira et al. (2017) for the *Sleep EDF 2002* and *Sleep*

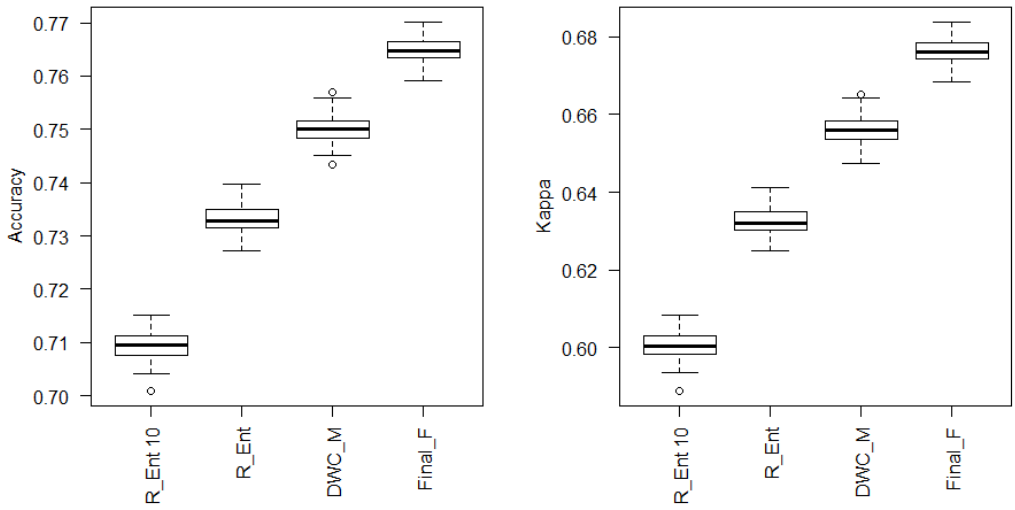


Figure 6.B: Boxplots for Figure 6.A with Non EEG feature set performance removed. To provide a closer look at the competitive feature sets.

EDF Expanded data sets. A closer inspection of the box plots from Figure 6.A shows that when the non-EEG performance was removed, Figure 6.B, the random forest constructed with the final features yielded the best classification results for CF00N, which indicates the feature selection process was successful. Figure 6.C shows the same trend across feature sets for the 128-tree random forests and in Table 6.2 the performance values compared to the values in Table 6.3 indicate the increase from 64 trees to 128 trees had almost no improvement. Hence, the performance measures of **Setting 1** converged for the CF00N data using the 64-tree random forests.

6.1.2 Setting 2: 10-Fold Cross Validation

Once again, for each feature set the epochs from all patients were combined into a complete data set in order to be analyzed across 100 trials. In each trial, ten equal sized random folds of the 80,763 epochs were created. Nine of the ten folds were used to build the random forest and the last fold was used as the testing data. This process was repeated until all ten individual folds had been used as the testing data.

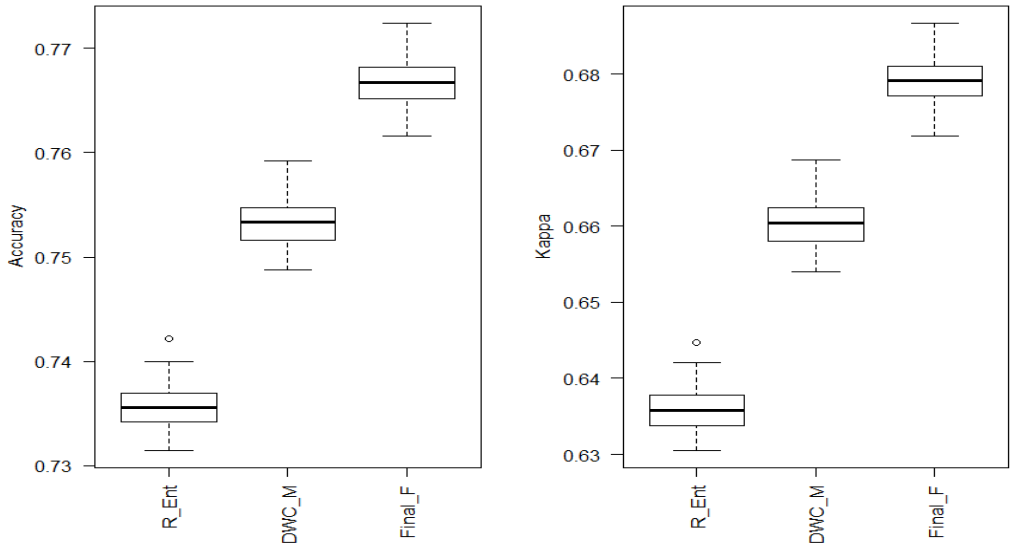


Figure 6.C: boxplots for 128-tree random forest classification accuracy and Cohen's κ for 100 trials, using random (1/3) testing, (2/3) training.

Then for each feature set the mean classification accuracy and κ of the ten folds were recorded in each trial. As in **Setting 1**, in each trial the epochs in each of the ten folds are the same across feature sets, which ensures a fair comparison across feature sets.

Data Set	Features	Accuracy (%)	κ
Sleep EDF Expanded	DWC M	91.5	0.83
CF00N	Renyi Entropy	73.74(0.07)	0.64(0.0007)
CF00N	DWC M	75.26(0.07)	0.66(0.0007)
CF00N	Non EEG	60.64(0.1)	0.44(0.0011)
CF00N	Final Selected	76.85(0.06)	0.68(0.0006)

Table 6.4: 64-tree random forest performance using 10-fold cross validation. Values reported are the mean accuracy of the 100 trials, of 10-fold cross validation.

The reported accuracy and κ value by da Silveira et al. (2017) that used 64-tree random forests is much higher for the *Sleep EDF Expanded* data set when compared to CF00N, across all feature sets for both 64 and 128- tree random forests, as seen in Tables 6.4 and 6.5. The discrete wavelet coefficient (DWC) features of da Silveira et al. (2017) again performed better than those of Fraiwan et al. (2012),

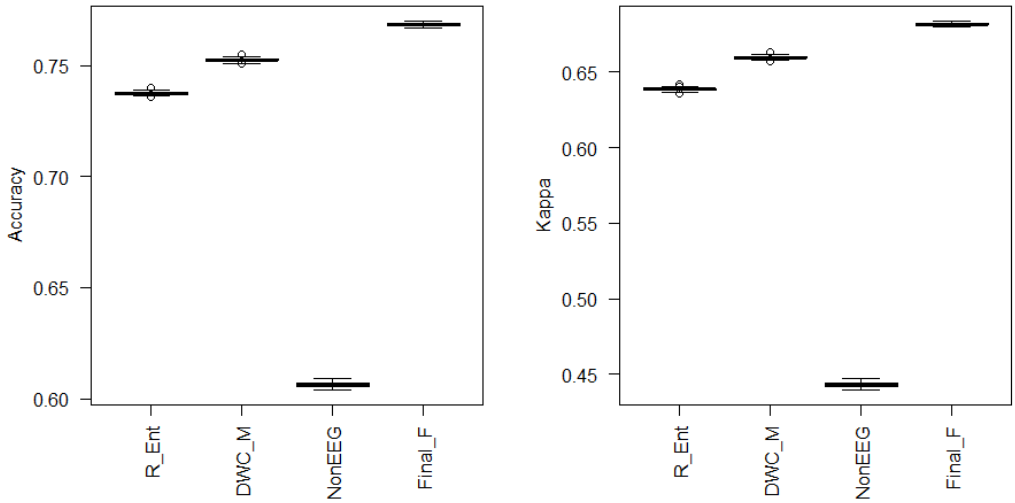


Figure 6.D: 64 tree random forest classification performance in the 10-fold cross validation setting.

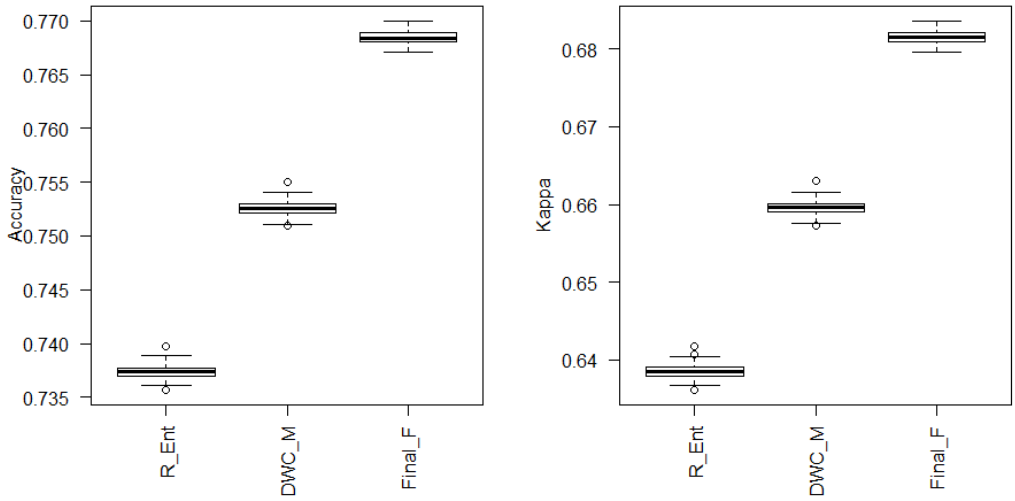


Figure 6.E: 64-tree random forest classification performance in the 10-fold cross validation setting. Non EEG performance removed.

Data Set	Features	Accuracy (%)	κ
CF00N	Renyi Entropy	73.96(0.05)	0.64(0.0007)
CF00N	Moments of DWC	75.5(0.05)	0.66(0.0006)
CF00N	Non EEG Moments	61.46(0.07)	0.45(.0011)
CF00N	Final Selected	77.04(0.05)	0.68(0.0007)

Table 6.5: 128-tree random forest performance using 10-fold cross validation. Values reported are the mean accuracy of the 100 trials, of 10-fold cross validation.

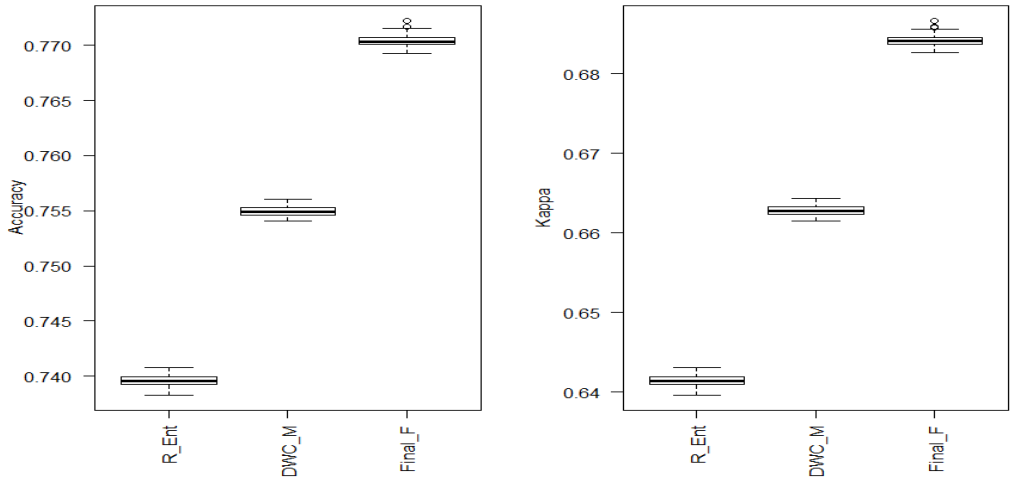


Figure 6.F: 128-tree random forest classification performance in the 10-fold cross validation setting. Non EEG performance removed.

but, overall, the random forests in **Setting 2** barely perform better than those of **Setting 1**. However, the standard deviations of the performance measures decreased, meaning the classification measures are more consistent in **Setting 2**. This is likely due to the increased size of the training data used to build each tree in the random forests. Again, the increase from 64 to 128 trees yielded little improvement in **Setting 2**, but once more the final selected features had the best performance measures for the CF00N data.

6.1.3 Setting 3: LOPOCV

In the LOPOCV, each patient's epochs are considered the testing data for 100 trials. In each of the 100 trials, a random forest is constructed using epochs from all other

patients as the training data and the sleep states of the testing data predicted. The mean classification accuracy and κ of the 100 trials is recorded for each patient. The mean accuracy and κ values of performance for all patients across feature sets are in Table 6.6.

Features	64 Trees		128 Trees	
	Accuracy (%)	κ	Accuracy (%)	κ
Renyi Entropy	64.8(11.7)	0.51(0.149)	65(11.1)	0.51(0.15)
DWC M	73.79(6.3)	0.63(0.085)	74(6.3)	0.63(0.09)
Non EEG	33.21(6.3)	0.02(0.047)	33.7(6.7)	0.02(0.05)
Final Features	70.83(8)	0.58(0.114)	67.1(12.1)	0.54(0.13)

Table 6.6: 64and 128-tree random forest performance on CF00N data. Reported are the mean classification measures with standard deviation in brackets.

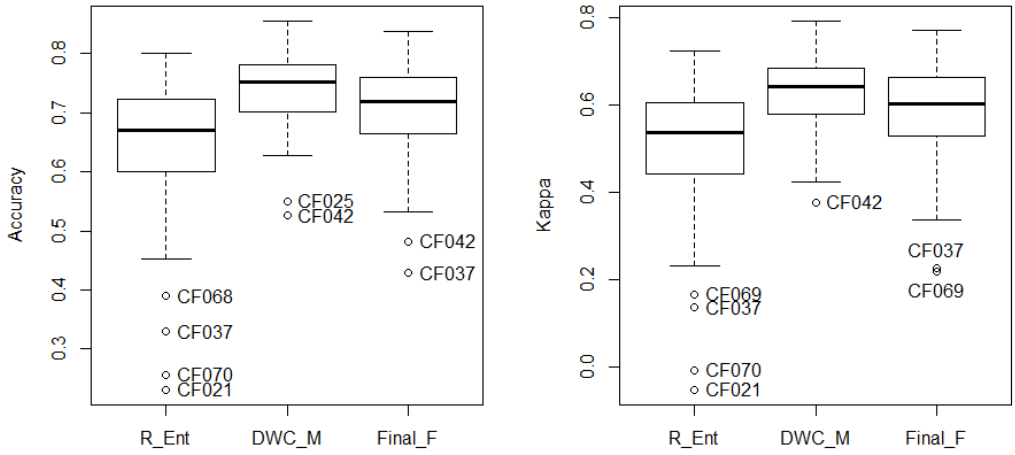


Figure 6.G: Boxplot of 64-tree random forest mean performance measures for all patients across Renyi Entropy, Discrete Wavelet Coefficient Moments and Final Features, in LOPOCV.

Features	64 Trees		128 Trees	
	Accuracy (%)	κ	Accuracy (%)	κ
Renyi Entropy	67.1 (7)	0.53(0.11)	67.0(7.5)	0.53(0.11)
DWC M	74.5 (5.3)	0.63(0.07)	74.4(5.9)	0.63(0.08)
Non EEG	33.6(6.3)	0.02(0.05)	34.0(6.7)	0.02(0.05)
Final Selected	72.2(6.4)	0.60 (0.1)	69.1(8.3)	0.56(0.11)

Table 6.7: Mean classification performance measures with patients CF021, CF025, CF037, CF042, CF068, and CF070 removed.

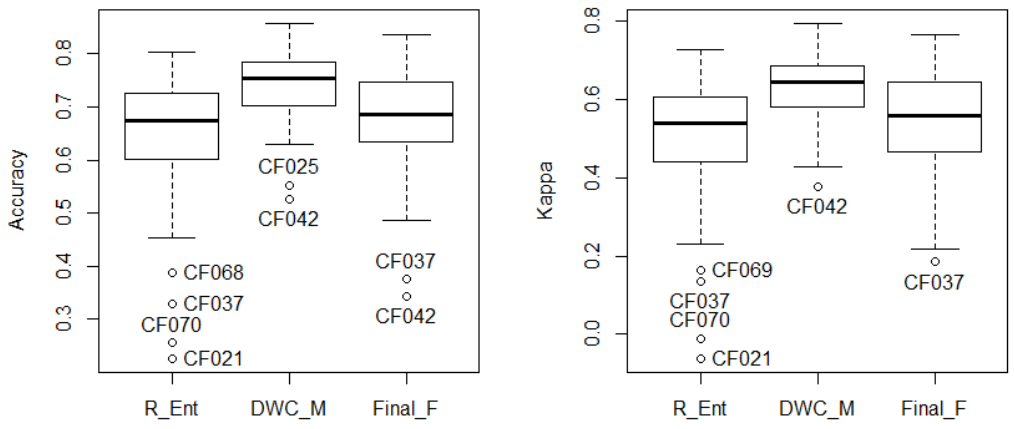


Figure 6.H: Boxplot of 128-tree random forest mean performance measures for all patients across Renyi Entropy, Discrete Wavelet Coefficient Moments and Final Features, in LOPOCV.

The classification performance for LOPOCV shows that in a clinical setting the mean performance measures have decreased despite increasing the size of the training data. The decrease in performance measures for **Setting 3** is likely due to the testing and training not sharing epochs from the same patient. The performance of random forests that used the final and DWC moments features only decreased slightly from **Setting 1** and **Setting 2**, but classification accuracy of the random forest that used Renyi entropy features has decreased by roughly 10%. As seen in Figures 6.G and 6.H, the classification performance for the epochs of patients CF021, CF025, CF037, CF042, CF068, and CF070 is quite poor, which means that there might be limitations to the random forest approach for sleep state classification in a clinical setting. Table 6.7 reports the mean accuracy of all patients with these six aforementioned patients removed. On the one hand, the mean performance values do improve, but they still remain lower than **Setting 1** and **Setting 2**. On the other hand, the mean accuracy for all patients is only slightly lower, $\approx 2\%$, than for the *CAP* data set used by Boostani et al. (2017).

Since the random forest analysis was performed for each patient's epochs, which used each feature set independently, an ANOVA with a block design was modelled,

with CF021, CF025, CF037, CF042, CF068, and CF070 removed, on each performance measure to determine which one was the best feature set for the LOPOCV classification. The individual patients were the blocks in the ANOVA in order to remove the variation of the performance measures between the individual patients from the total variation of all measurements. The performance measures for the non-EEG features are substantially lower than the other data sets. For this reason, the analysis was carried out using performance measures of the other three feature sets.

6.1.4 ANOVA with Patient Blocks

A MANOVA was considered for the purpose of determining the best feature set for random forest classification, but when using accuracy and κ as the multivariate response there is the problem of very strong multicollinearity ($\rho > 0.9$). Moreover, an initial investigation found the homoscedasticity assumption of the feature set groups was strongly violated. This was performed using Box's M-test (Box, 1949), finding a p-value of 0.001. Hence, the proposed ANOVA was performed for both accuracy and κ separately. Verification of the ANOVA assumptions must first be done before proceeding with results. This is presented simultaneously for accuracy and κ measures. The independence of observations is intact, as the performance measures of each patient in each feature set did not affect performance measures of another patient or in another feature set for the same patient. Thus, the observations were considered independent between patients and across feature sets. The homoscedasticity of the feature groups was assessed using Levene's-test (Levene, 1960) and the results are shown in Table 6.8. The p-value for accuracy in Table 6.8 is a bit concerning because, at a significance level of 5%, it just barely rejected the null hypothesis that covariances of feature groups were equal. However, the standard rule of thumb for assessment of equal variances considers the magnitude of largest variance to the smallest. If the largest variance divided by the smallest variance is less than 2, $\left(\frac{\sigma_{max}}{\sigma_{min}} < 2\right)$, then the assumption is considered to be met,

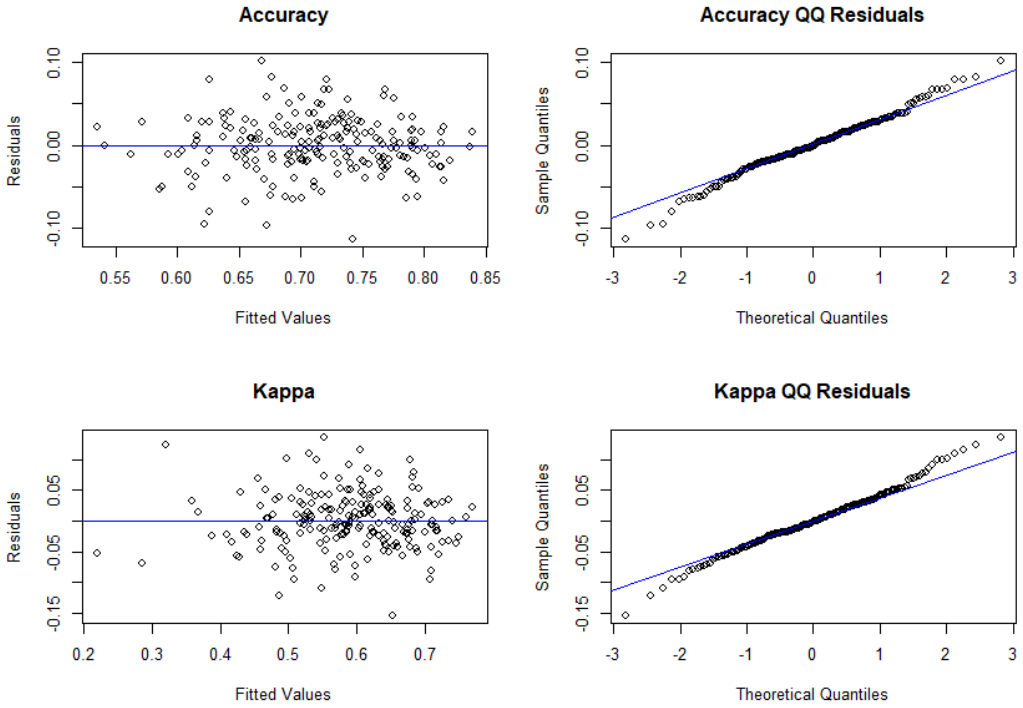


Figure 6.I: Residuals vs Fitted Values plot, and QQ plots of residuals for assessment of ANOVA assumptions.

which was satisfied for both performance measures. The normality assumption for the residuals was assessed using the QQ plots in Figure 6.I. The residuals for both accuracy and κ ANOVA models indicated the residuals were normally distributed. Furthermore, the residual vs fitted value plots in Figure 6.I indicated that there was no non-linear trend, which meant the linear ANOVA model was appropriate for both performance measures.

Performance Measure	F-statistic	p-value	$\frac{\sigma_{max}}{\sigma_{min}}$
Accuracy	3.179	0.044	1.779
κ	2.497	0.085	1.768

Table 6.8: Table of output for Levene's test of homogeneity of variances for feature groups and ratio of the largest variance to smallest.

The ANOVA tables for performance measures are Tables 6.9 and 6.10. At a 5% significance level, the data provides strong evidence that a significant difference in the mean of accuracies between feature sets exists, which is similarly seen for the mean κ measures. The significant difference found between feature sets was fol-

lowed by a Tukey's Honestly Symmetric Differences (HSD) test (Tukey, 1949) for both performance measures using an overall 5% significance level. The resulting confidence intervals and p-values of the mean differences between groups is presented in Table 6.11. With 95% confidence, the accuracy for random forests that used the DWC features was on average between 5.74% and 9.09% higher than the random forests that used the Renyi entropy features. Similarly, the accuracy was on average between 0.65% and 3.99% higher when compared to random forests that used the final selected features. Overall, the mean classification accuracy was the highest when using the DWC moments features. The same can be concluded for the κ measures in Table 6.11. The next step was to investigate the performance measures of random forest for these three feature sets across the various demographic groups: age, gender, and OSA status. This was done by incorporating the demographic information into a MANOVA model that used the performance measures of each feature set as the multivariate responses, $(n = 69, p = 3)$. This was done separately for both the classification accuracy and the κ measures.

Treatment	Df	Mean SS	F-statistic	p-value
Features	2	992.5	57.761	$< 2 \times 10^{-16}$
Patients	68	84.3	4.906	1.72×10^{-15}
Residuals	136	17.2	-	-

Table 6.9: ANOVA table for accuracy measures across feature sets.

Treatment	Df	Mean SS	F-statistic	p-value
Features	2	0.18412	59.441	$< 2 \times 10^{-16}$
Patients	68	0.02033	6.564	$< 2 \times 10^{-15}$
Residuals	136	0.00310	-	-

Table 6.10: ANOVA table for κ measures across feature sets.

Measure	Features	Lower Bound	Upper Bound	p-value
Accuracy (%)	DWC M - R Ent	5.74	9.09	$< 2 \times 10^{-16}$
	DWC M - Final F	0.65	3.99	0.003
	Final F - R Ent	3.42	6.77	$< 2 \times 10^{-10}$
κ	DWC M - R Ent	0.079	0.124	$< 2 \times 10^{-6}$
	DWC M - Final F	0.057	0.123	0.001
	Final F - R Ent	0.044	0.089	$< 2 \times 10^{-16}$

Table 6.11: Results of Tukey's HSD test, for multiple comparison following the ANOVA. Reported are the lower and upper bounds for the mean difference between feature sets.

6.1.5 MANOVA

In an ideal MANOVA setup, the demographic groups would have balanced numbers of subjects in each group combination. Table 6.12 show the number of patients in the OSA groups. As the OSA groups are very unbalanced, they were re-coded into a binary variable. No OSA and Mild was recoded to Low OSA and Moderate and Severe OSA re-coded into High OSA. Table 6.13 provides the number of patients in the re-coded OSA groups across all gender and age combinations. The number of males to females is very disproportionate and the relatively small numbers of females in each OSA age combination is problematic for MANOVA, especially for cells with less than 3 patients as there are three responses for each performance measure. Furthermore, there is only a single female under 13 years of age in the low OSA group, which meant the estimated covariance of this group was exactly 0, and the homoscedasticty for a 3-way MANOVA with interaction could not be assumed. Therefore, the gender factor was not included in the MANOVA models, leaving only the age and re-coded OSA factors to construct a 2-way MANOVA with interaction. All of the age OSA groups had more than 5 observations, but were still slightly imbalanced. The ideal MANOVA setup would be balanced with at least 20 patients in each cell.

As stated in the ANOVA analysis, before proceeding with the results, verification of the MANOVA assumptions comes first. The assumption that patients in the age and OSA group combinations were independent was clearly met. The

No OSA	Mild	Moderate	Severe
4	37	16	12
Low OSA		High OSA	
41		28	

Table 6.12: Number of patients in each OSA group

Gender	Low OSA		High OSA		Totals
	Under 13	Over 13	Under 13	Over 13	
Male	18	13	14	8	53
Female	1	9	4	2	16
Totals	19	22	18	10	69

Table 6.13: Number of Patients in all demographic group combinations, with CF021, CF025, CF037, CF042, CF068, and CF070 removed

homoscedasticity assumption of equal covariance for all age OSA group combinations was again verified using the Box M-test (Box, 1949). The results presented in Table 6.14 indicated that the homoscedasticity assumption was intact for both performance measures. The marginal and multivariate normality of residuals was assessed using the QQ plots in Figures 6.J and 6.K. The marginal normality of the residuals assumption appeared to be met for both classification accuracies and the κ measures. The Chi-square QQ plots in these figures indicated that the multivariate normality of the residuals was also present. Figure 6.L contains the residuals versus fitted values plots for each response variable. All demographic groups showed a similar spread around the line (y=0), however, there may be more potential outliers in accuracy (CF027) and κ measures (CF069), but the homoscedasticity assumption was not violated, so these patients were kept in the analysis. All assumptions of the 2-way MANOVAs with interaction were considered met and the linear model was considered appropriate.

Measure	Df	χ^2 statistic	p-value
Accuracy (%)	18	6.59	0.993
κ	18	11.43	0.875

Table 6.14: Results of Box's M-test (Box, 1949) for testing equal covariance of OSA and age group combinations in the MANOVA.

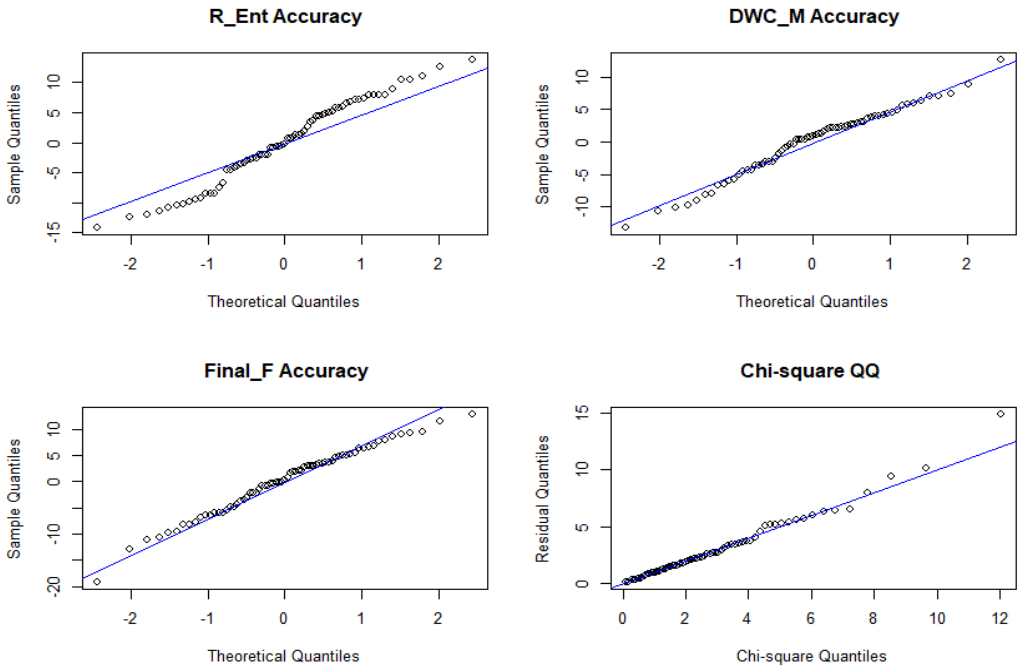


Figure 6.J: Residual QQ plots for Accuracy performance

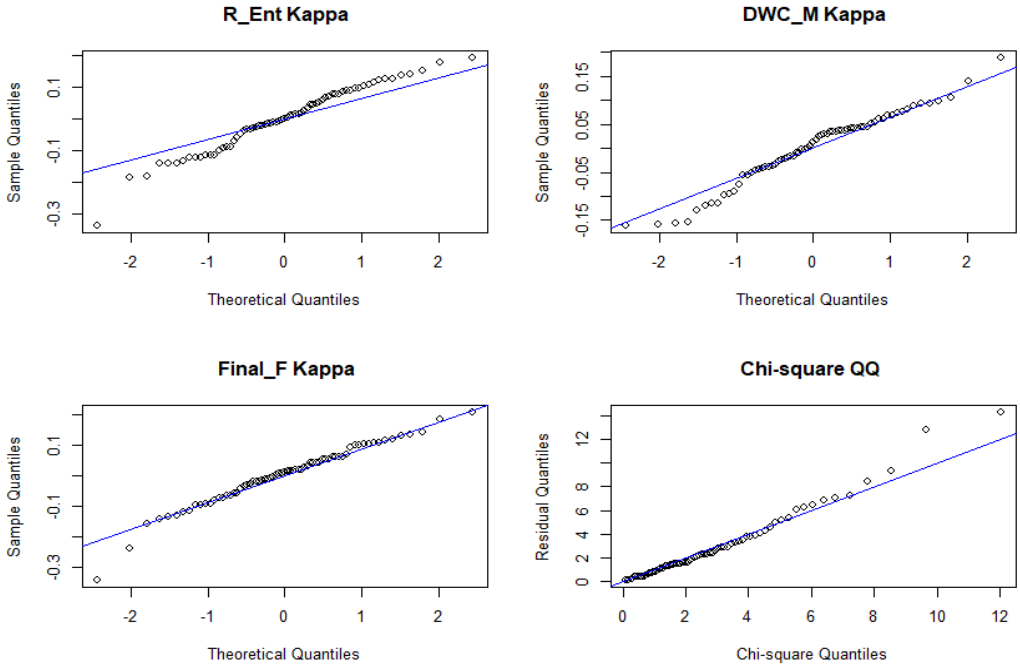


Figure 6.K: Residual QQ plots for κ measures

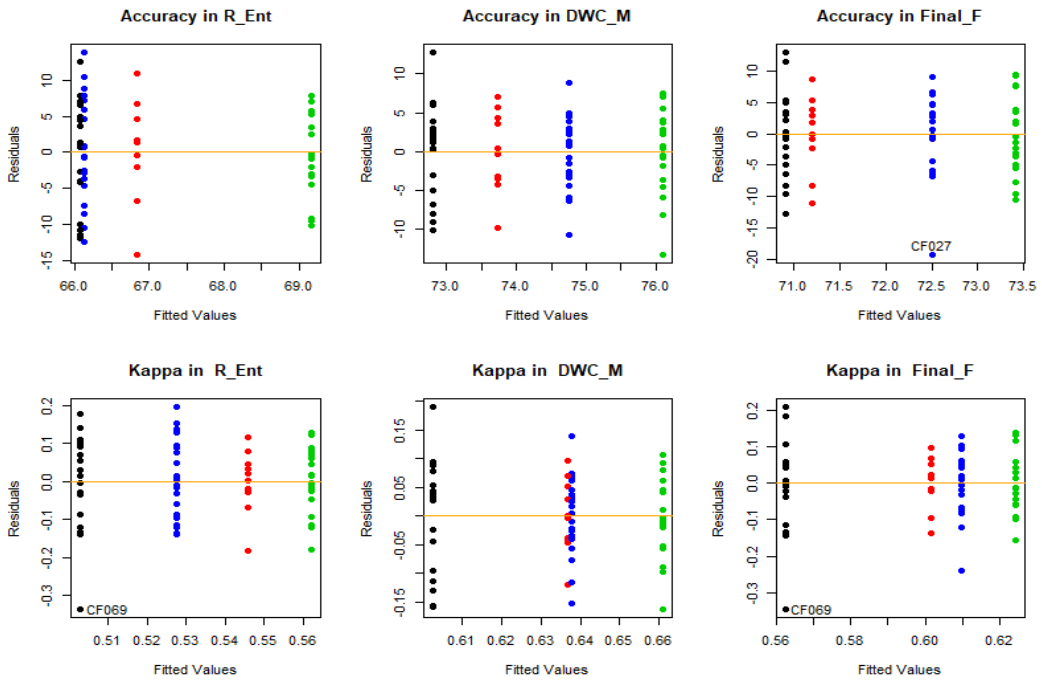


Figure 6.L: Residual versus fitted values plots. Group colors: Green (Under 13:Low OSA), Black (Under 13:High OSA), Blue (Over 13:Low OSA) , Red (Over 13:High OSA)

Factor	≈ F-statistic	Dfn	Dfd	p-value
Age	0.268	3	63	0.8482
OSA	1.03	3	63	0.3853
Age:OSA	0.626	3	63	0.6007

Table 6.15: 2-way MANOVA table for classification accuracy.

Factor	≈ F-statistic	Dfn	Dfd	p-value
Age	0.21	3	63	0.889
OSA	1.11	3	63	0.3516
Age:OSA	1.06	3	63	0.369

Table 6.16: 2-way MANOVA table for κ measures.

The results of the 2-way MANOVA are presented in Tables 6.15 and 6.16. The first hypothesis tested in each of the 2-way MANOVAs was for the interaction effect and with p-values of 0.6007 and 0.369 compared to a significance level of 5%, the data did not provide sufficient evidence of an interaction effect between age and OSA status in either MANOVA. Next, was the hypothesis test for the main effects of age and OSA, which, at a 5% significance level, the data did not provide sufficient

evidence that patient age or OSA status had an effect on the mean accuracy or the mean κ measures across feature sets.

Study	Data	Features	Accuracy (%)	# of Trees
Boostani et al. (2017)	*Sleep-EDF 2002*	Renyi	87.06	10
Boostani et al. (2017)	*CAP*	Renyi	69.05	10

Table 6.17: Classification performance of 10 -tree random forest using LOPOCV of Boostani et al. (2017).

The findings of the MANOVA presented in this work are contradictory to what was found by Boostani et al. (2017) presented in Table 6.17. However, there is a major difference between what was done here and by Boostani et al. (2017). The difference being that Boostani et al. (2017) analyzed PSG recordings from two different data sets, *Sleep EDF 2002* and *CAP*. Moreover, the *CAP* data base from *Physionet* (Goldberger et al., 2000) also contains PSG recordings for 16 healthy subjects that were not included in the study by Boostani et al. (2017). Including these recordings in the analysis by Boostani et al. (2017) might have been a better approach, as the PSGs of healthy subjects and patients were recorded and scored by the same sleep laboratory. This means that the difference in classification accuracy between patients and healthy subjects found in Boostani et al. (2017) could be a result of the PSG recordings being from different sleep studies. The PSG recordings of the CF00N data presented in this work are from the same sleep study performed at the University of Alberta Hospital. Lastly, Boostani et al. (2017) used the LOPOCV approach on the *Sleep EDF 2002* data and saw an increase in classification performance from Fraiwan et al. (2012), who used the approach in **Setting 1**, which was quite surprising, but Boostani et al. (2017) used all 20 PSG recordings versus Fraiwan et al. (2012) who used 16 PSG recordings.

The study by Koley and Dey (2012) used a support vector machine (SVM) to classify sleep epochs from 28 adults with suspected OSA (AHI > 5 considered positive for OSA). 16 subjects (8 OSA, 8 No OSA) composed the training data and 12 subjects (5 OSa, 7 No OSA) for the testing. The epochs of all patients were combined in the testing and training data sets, which is to say that subjects were not considered individually. Koley and Dey (2012) reported a classification accuracy

of 89.91% and κ= 0.868 for the epochs of the No OSA subjects in the testing data. For the epochs of the OSA subjects in the testing data a classification accuracy of 88.86% and κ = 0.846 was reported. These findings suggest that when PSG recordings from the same sleep study are analyzed there may not be a difference in classification performance between patient status groups, especially in regard to OSA status. However, the LOPOCV should be adopted in order to statistically confirm that a difference exists between the means of performance measures for the patient groups.

Chapter 7

HMM and NHMM Analysis

7.1 HMM and NHMM Data

The patient data sets used in the analysis of HMMs and NHMMs are the Renyi entropy features and the final features. The final feature set consists of the 9 selected features from chapter 5. The first 5 features are the Renyi entropy of the wavelet coefficients in the following sub-bands: K-complex, delta, theta, alpha, and sleep spindle. The variances of the discrete wavelet coefficients in level 4, level 7, level 8, and scale coefficient frequency bands transformed to normality via the natural logarithm are the other 4 features. For each patient's data sets, 500 HMMs and 500 NHMMs with different random initialization parameters were fitted and the models the largest log-likelihood were chosen as the final HMMs and NHMMs for that patient. The classification performance measures were calculated after the final models for each patient were chosen.

7.2 Performance Measures

The assessment of classification performance for HMM and NHMM models will be done using classification accuracy and the adjusted Rand index (ARI) of Steinley

(2004b). The ARI measures the pair-wise agreement between two partitions and is the widely preferred metric for validating cluster performance. McNicholas (2017, p. 7) points out that "Steinley (2004a) gives detailed simulations showing that the ARI is preferable to the misclassification rate when the number of clusters equals the number of known classes". In this work, the first partition for each patient was the expert scored AASM sleep states and the global decoding solution of the HMM or NHMM was the second. Since the HMM and NHMM in this work are designed to model a patient's epochs into 5 clusters, one for each AASM sleep state, the comparison between these partitions allows the ARI to be used for evaluation of classification performance.

7.3 Preliminary Analysis

To begin the preliminary analysis, classification performance of the HMM and NHMM models between the feature sets is analyzed. Box plots of the performance measures of HMM and NHMM for each patient's feature sets are presented in Figure 7.A, where it can be seen that the NHMMs appear to perform slightly better, but they had larger variance than the HMMs. Although it can be seen in the ARI box plot that the NHMMs were able to slightly improve the ARI values from the HMM for some patients that used the same features, the median ARI is below 0.5 for all HMM and NHMM performances, which indicated that the global decoding of patient sleep states was sub-optimal for at least half of the patients. Moreover, the accuracy box plot shows that the classification performance of patient CF069 is quite poor for all models. This is not overly surprising, as patient CF069 only has 4 distinct sleep states in their epochs (missing REM) and the HMM and NHMM cluster the epochs into 5 states. For this reason, the performance measures for CF069 were removed before further analysis. The values in Table 7.1 show the mean accuracy and ARI values (standard deviation) with CF069 removed.

Features	HMM		NHMM	
	Accuracy	ARI	Accuracy	ARI
Renyi	67.1 (7.9)	0.46 (0.10)	68.9 (7.8)	0.49 (0.11)
Final	65.3 (8.7)	0.45 (0.10)	67.6 (9.1)	0.48 (0.11)

Table 7.1: Mean of performance measures (standard deviation) for each model with CF069 removed.

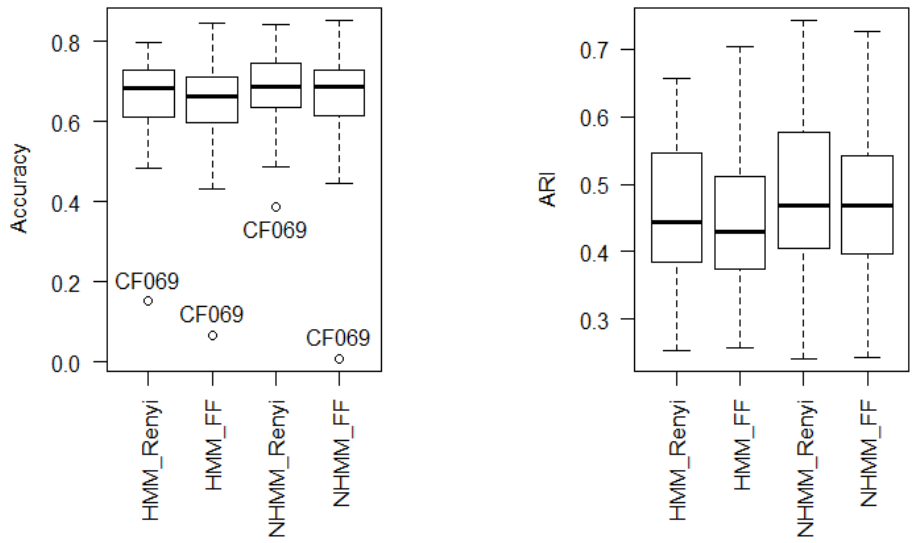


Figure 7.A: Box plot of performance measures across model (HMM or NHMM) and patient feature sets (Renyi or Final Features).

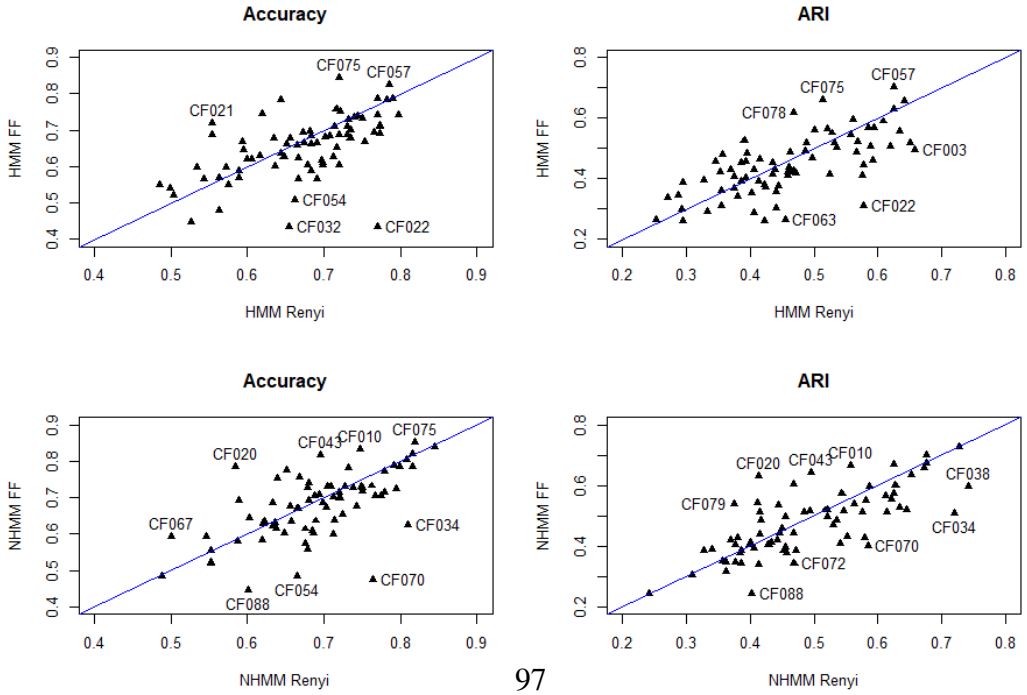


Figure 7.B: Comparison of HMM and NHMM performance across the Renyi entropy and final selected features

The mean classification performance measures in Table 7.1 indicated the Renyi features typically performed better on average, but only slightly. However, the scatter plots in Figure 7.B show the comparison of classification performance of HMMs and NHMMs between feature sets and gives insight on the model performances for individual patients. The line on the scatter plots indicates where the HMM or NHMM that used a patient's Renyi entropy features would be equal to the HMM or NHMM that used the final selected features of the same patient, ie. the line y=x on the domain [0,1]. Having performance measures above the line means that the model that used the final selected features performed better than the one that used the Renyi entropy features. The ARI plot for the HMMs shows that for patients with ARI values below 0.5, the HMM that used the final selected features typically performed better. However, for patients with ARI above 0.5, the HMM that used the Renyi entropy features performed better. The same trend can be seen in the ARI plot for NHMM. The plots for HMM and NHMM accuracy between feature sets show that the feature set used to construct the model can have a significant impact on classification accuracy for some patients. For example, the HMM accuracy for patient CF022 was rather poor, 43.3%, when the model used the final selected features but improved to 77.0% when the HMM used the Renyi features. The same argument can be made for the classification accuracies of the NHMM with patient CF070. Conversely, the HMM accuracy of patient CF021 improved when the HMM used the final features and the NHMM accuracy of patient CF020 also improved when using the final features. Overall, HMM and NHMM classification performances were found to be relatively the same between feature sets.

7.4 HMM vs NHMM

Now that the performance between features sets for HMM and NHMM overall have been considered, a further inspection between the best HMM performance of each feature set and their corresponding NHMM was considered before the overall comparison between HMM and NHMM in each feature set. Presented in Tables 7.2 and

7.3 are the confusion matrices for the HMM and NHMM that used Renyi entropy features of patient CF003. Similarly, for the final features data, Tables 7.4 and 7.5 are the confusion matrices for the HMM and NHMM of patient CF057. CF003 and CF057 were chosen because the ARI value for the HMMs of these patients were the highest in each feature set. The NHMM did improve the classification performance for CF003, specifically for the NREM 1, REM, and wake states, and there was also improvement though slight compared to CF003, for CF057 in the NREM 2, NREM 3, and wake states. In both feature sets the NHMM improved the classification of the best HMM performance.

CF003	HMM Classification, Accuracy = 77.23%, ARI = 0.658					
Expert Scores	States	NREM 1	NREM 2	NREM 3	REM	Wake
	NREM 1	4	1	0	13	1
	NREM 2	20	354	0	32	0
	NREM 3	0	19	201	0	0
	REM	8	2	2	113	0
	Wake	117	18	1	5	139

Table 7.2: Confusion matrix for HMM performance using Renyi entropy features of patient CF003.

CF003	NHMM Classification, Accuracy = 81.52%, ARI = 0.673					
Expert Scores	States	NREM 1	NREM 2	NREM 3	REM	Wake
	NREM 1	9	1	0	9	0
	NREM 2	26	353	0	27	0
	NREM 3	0	20	199	1	0
	REM	3	1	2	119	0
	Wake	76	17	1	10	176

Table 7.3: Confusion matrix for NHMM performance using Renyi entropy features of patient CF003.

CF057	HMM Classification, Accuracy =82.6% ARI=0.703					
Expert Scores	States	NREM 1	NREM 2	NREM 3	REM	Wake
	NREM 1	1	0	0	4	12
	NREM 2	26	325	4	18	5
	NREM 3	2	29	172	0	0
	REM	6	0	0	153	1
	Wake	57	1	0	0	130

Table 7.4: Confusion matrix for HMM performance using the final selected features of patient CF057.

CF057	NHMM Classification, Accuracy =84.03%, ARI = 0.726					
Expert Scores	States	NREM 1	NREM 2	NREM 3	REM	Wake
	NREM 1	1	1	0	3	12
	NREM 2	17	335	6	17	3
	NREM 3	2	27	174	0	0
	REM	7	0	0	153	0
	Wake	52	3	1	0	132

Table 7.5: Confusion matrix for NHMM performance using final selected features of patient CF057.

The scatter plots of accuracy and ARI for HMM versus the NHMM in each feature set can be seen in Figures 7.C and 7.F. Again, the line on each of the plots is where the HMM and NHMM performance measures would be equal when they used the same feature set. Having observations above the line mean the NHMM improved the classification performance. In Figure 7.C, roughly half of the patients saw an increase in classification performance when the NHMM was used over the HMM, but for CF022 there was a significant decrease in accuracy and ARI.

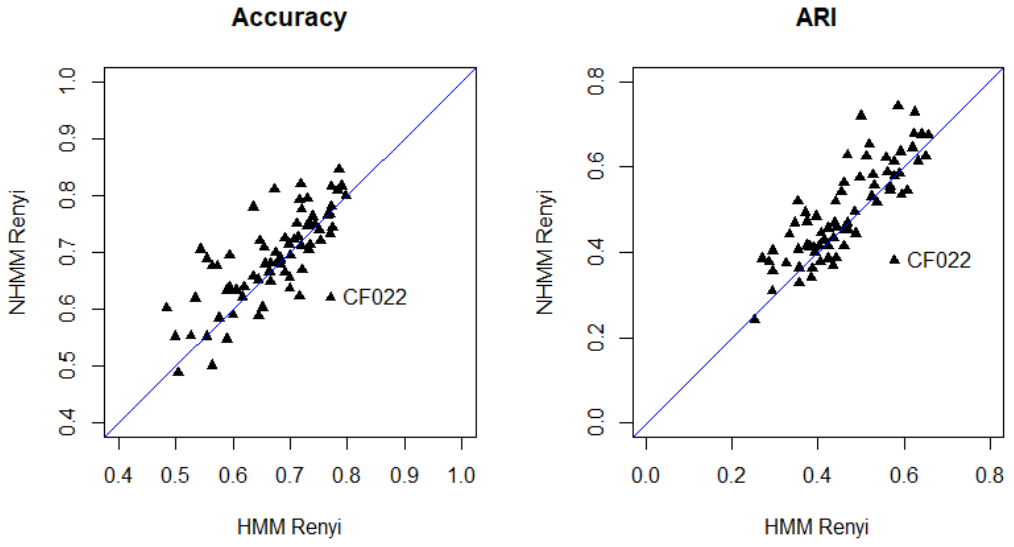


Figure 7.C: Performance of HMM vs NHMM using the Renyi features for patient data, with CF069 removed. Blue line is where HMM performance = HMM performance, ie. the line y=x on the domain [0,1]

Figures 7.D and 7.E are the hypnograms of CF022 and CF034, the patients who saw the worst and best improvement from the HMM to NHMM using the Renyi

entropy features. It can be seen in Figure 7.D that the HMM for CF022 was better able to detect the transitions to the NREM 2 and NREM 3 states from the active sleep states. However, the NHMM was better able to discriminate between the REM and wake states.

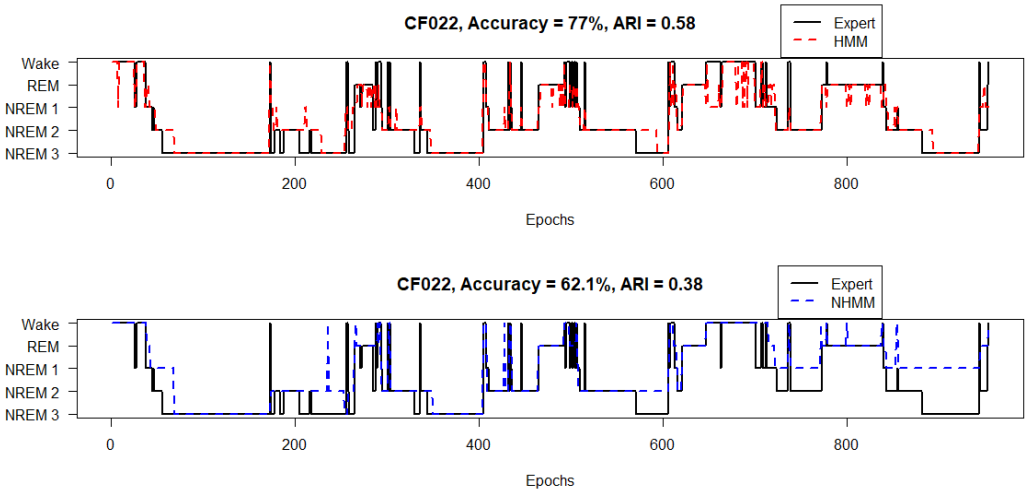


Figure 7.D: Hypnograms of patient CF022 comparing the HMM and the NHMM classification using the Renyi entropy features. Accuracy of each state for HMM: Wake 56%, REM 85.5%, NREM 1 61.11%, NREM 2 91.3%, NREM 3 72.4%. Accuracy of each state for NHMM: Wake 85.6%, REM 97.8%, NREM 1 9.3%, NREM 2 62.2%, NREM 3 47%

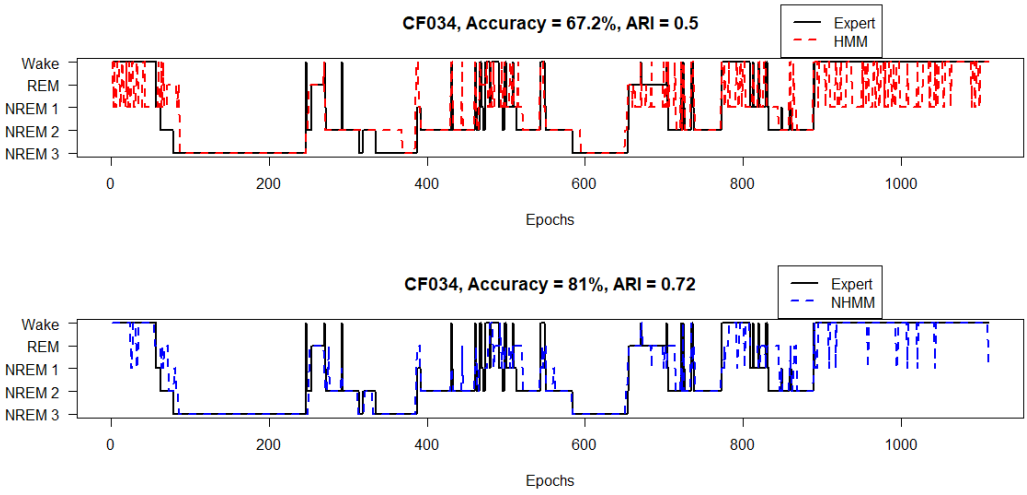


Figure 7.E: Hypnograms of patient CF034 comparing the HMM and the NHMM classification using the Renyi entropy features. Accuracy of each state for HMM: Wake 50.7%, REM 38.1%, NREM 1 67.7%, NREM 2 80.2%, NREM 3 78.6%. Accuracy of each state for NHMM: Wake 80.8%, REM 95.2%, NREM 1 13.8%, NREM 2 77.8%, NREM 3 96.6%

The hypnograms of CF034 in Figure 7.E show that the NHMM for CF034 was

better able to detect the transitions between NREM 2 and NREM 3 and discriminate between the wake state and NREM 1, which is indicated by the fewer transitions to NREM 1 in the wake state portions at the beginning and end of the PSG recording. The increased number of transitions between the wake and NREM 1 states for the HMM improves the overall classification of the NREM 1 state, but at the cost of decreasing the accuracy for the wake and REM states. Despite the decrease in performance for CF022 from HMM to NHMM, the NHMM for both CF022 and CF034 was better at effectively classifying the active sleep states wake and REM.

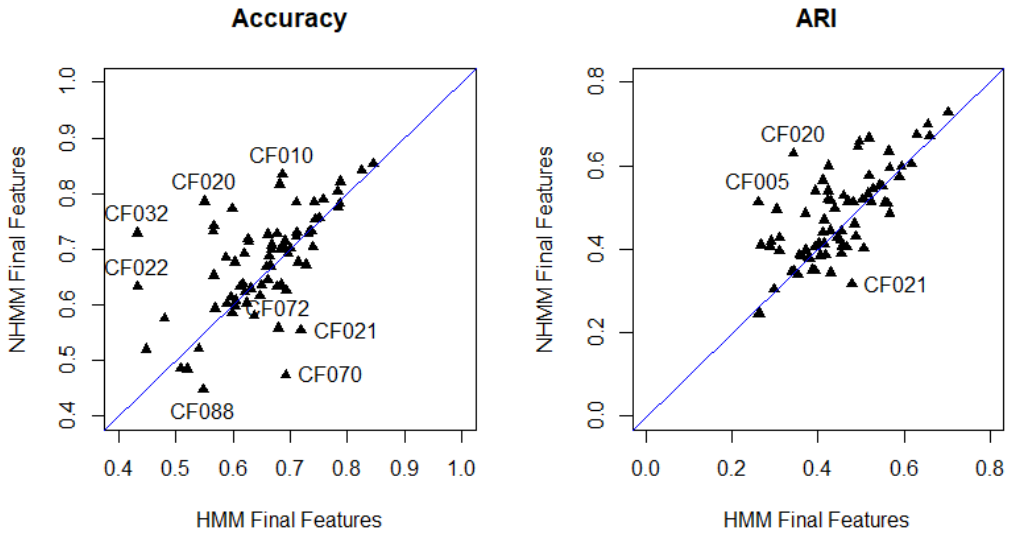


Figure 7.F: Performance of HMM vs NHMM using the final selected features for patient data, with CF069 removed. Blue line is where HMM performance = HMM performance, ie. the line y=x on the domain [0,1]

The overall trend from Figure 7.C can also be seen in Figure 7.F for HMM vs NHMM with the final features, but with more variation around the line. The patients CF010, CF020, CF022, and CF032 had a relatively larger increase in accuracy from HMM to NHMM, however, CF021, CF070, CF072, and CF088 had a significant decrease in accuracy. The hypnograms for patients CF021 and CF020, who saw the the worst and best improvement from HMM to NHMM using the final features, can be seen in Figures 7.G and 7.H. Patient CF021 spent the majority of the PSG recording in the wake state and had a very small proportion of REM

sleep, which indicates that the patient did not sleep very well during the laboratory PSG. The extended periods of being awake drastically hurt the performances of the HMM and NHMM, but the HMM was better able to classify the wake state, which increased the overall performance compared to the NHMM. Despite the decrease in performance measures for CF021 from NHMM to HMM, both the HMM and NHMM classification performances were sub-optimal for patient CF021.

The hypnograms for CF020 in Figure 7.H tell a different story, which is that the NHMM was better able to detect the transitions between NREM 2 and NREM 3 from the more active sleep states. Furthermore, the NHMM was better able to discriminate between the NREM 1 state and the wake and REM states for patient CF020, which is why there was such a big increase in the ARI from HMM to NHMM. The inability to effectively discriminate between all the active sleep states, NREM 1, REM, and wake, is a problem that plagues many sleep state classification algorithms. However, the hypnograms for CF020 show that the NHMM is able to accomplish this task for some patients without sacrificing accuracy of the other states.

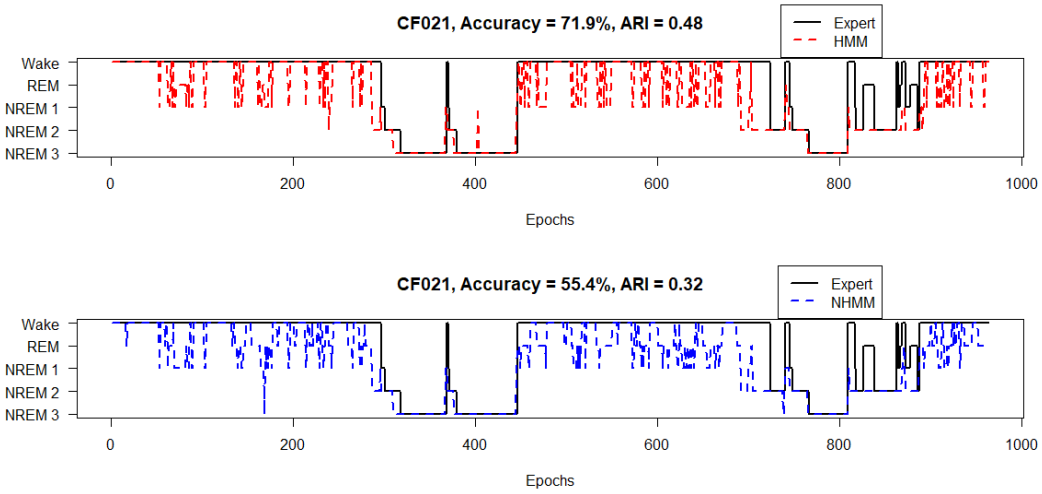


Figure 7.G: Hypnograms of patient CF021 comparing the HMM and the NHMM classification using the final features. Accuracy of each state for HMM: Wake 67.7%, REM 0%, NREM 1 0%, NREM 2 87.2%, NREM 3 98.1%. Accuracy of each state for NHMM: Wake 44.1%, REM 0%, NREM 1 0%, NREM 2 85.1%, NREM 3 98.7%

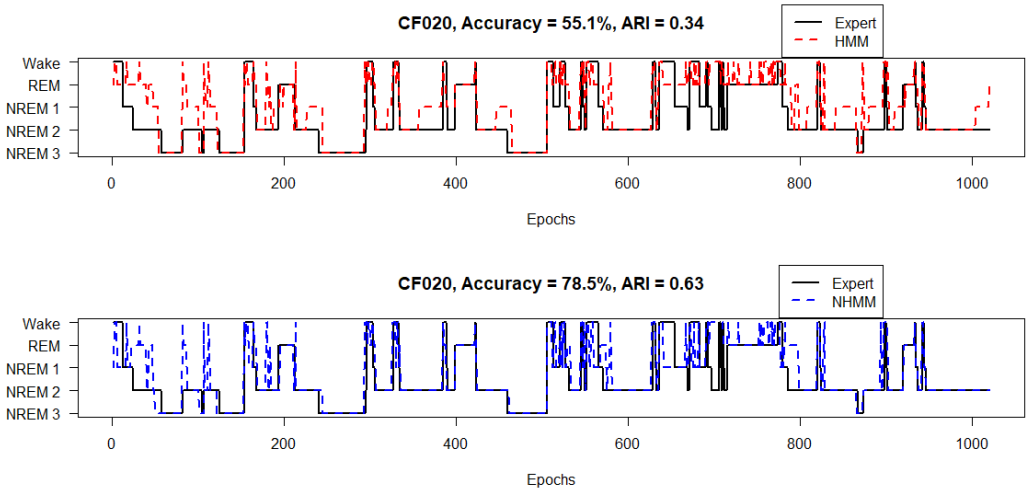


Figure 7.H: Hypnograms of patient CF020 comparing the HMM and the NHMM classification using the final features. Accuracy of each state for HMM: Wake 54.1%, REM 76.7%, NREM 1 1.2%, NREM 2 47.5%, NREM 3 92.1%. Accuracy of each state for NHMM: Wake 55.7%, REM 81.9%, NREM 1 58.5%, NREM 2 81.1%, NREM 3 94.5%

It appears overall that within both feature sets the NHMM does improve classification performance over the HMM. However, a one-sample t-test on the differences between NHMM and HHM performance measures was performed to be certain that the increase in accuracy and ARI is statistically significant. To clarify, in each feature set the difference between HMM and NHMM accuracy was calculated for each patient and then the same was done for ARI. Then a one sample t-test of the differences was performed and the results are presented in Table 7.6. With 95% confidence, the NHMM accuracy was on average between 0.5% and 3.1% higher than the HMM accuracy when using the Renyi entropy features. Similarly, when using the final features data the NHMM accuracy was on average between 0.4% and 4.2% higher than the HMM accuracy. Furthermore, again with 95% confidence, the ARI values for the NHMM are on average between 0.015 and 0.045 higher than the HMM ARI values when using the Renyi entropy features. The ARI values for the NHMM are on average between 0.013 and 0.051 larger than the HMM ARI values when using the final features. Hence, the NHMM model outperforms the HMM model in terms of sleep state classification for both feature sets.

Features	Performance Measure					
	Accuracy			ARI		
	t , df = 73	p-value	95% CI	t , df=73	p-value	95% CI
Renyi	2.75	0.008	(0.5, 3.1)	3.90	0.0002	(0.015, 0.045)
Final	2.47	0.016	(0.4, 4.2)	3.39	0.001	(0.013, 0.051)

Table 7.6: Table of results from the one-sample t-tests for the mean difference between NHMM and HMM performance metrics. Differences are calculated patient-wise, NHMM measure - HMM measure.

7.5 Performance Across Age and OSA groups

The performance measures of the NHMM models were investigated further across demographic groups using MANOVA. Table 7.7 gives the number of patients in each combination of age, OSA, and gender groups (patient CF069 removed) that were used. The gender groups are still very imbalanced and have quite low cell counts for the majority of female OSA and age group combinations, which can be problematic for modelling a 3-way interaction term. Again, as in Chapter 6, the gender variable will not be included in the MANOVA model, which leaves each performance measure to be analyzed using a 2-way MANOVA ($n = 74, p = 2$). Once again, as in any hypothesis test, verification of the MANOVAs assumptions is required before proceeding with the results.

Gender	Low OSA		High OSA		Totals
	Under 13	Over 13	Under 13	Over 13	
Male	19	14	15	9	57
Female	2	9	4	2	17
Totals	21	23	19	11	74

Table 7.7: Number of Patients in all demographic group combinations, with CF069 removed.

The independence of observations within the demographic group combinations is intact, as patients cannot belong to both OSA groups or both age groups. Homoscedasticity across OSA and age group combinations was tested using Box's M-test (Box, 1949) and the results are presented in Table 7.8. The p-values for accuracy and ARI indicated that the equal covariances assumption was valid for both MANOVA models. The marginal normality of the residuals was verified using QQ plots in Figure 7.I for each performance measure. In each feature set, the majority of residuals fell on the line, which indicates that the quantiles of the residuals match those of a univariate normal distribution. The multivariate normality of residuals is checked using the Chi-square QQ plots shown in Figure 7.J. The Chi-square plot for the ARI indicates that the multivariate normality of residuals assumption is met for that MANOVA, however, the plot for accuracy shows that there could be potential outliers: CF020, CF034, CF054, CF070. The majority of accuracy residual quantiles matched that of the Chi-square, so the assumption of multivariate normality for the accuracy residuals was considered valid. Lastly, Figure 7.K shows the residuals versus fitted values in each performance measure across the two feature sets. In each plot the residuals are well balanced around the horizontal line y=0, hence, the linear MANOVAs model were considered appropriate. Thus, all of the MANOVA assumptions were met for each 2-way MANOVA model with the results presented in Tables 7.9 and 7.10.

Measure	Df	χ^2 statistic	p-value
Accuracy	9	14.79	0.097
ARI	9	8.93	0.443

Table 7.8: Output from Box's M-test for assessment of homoscedasticity assumption.

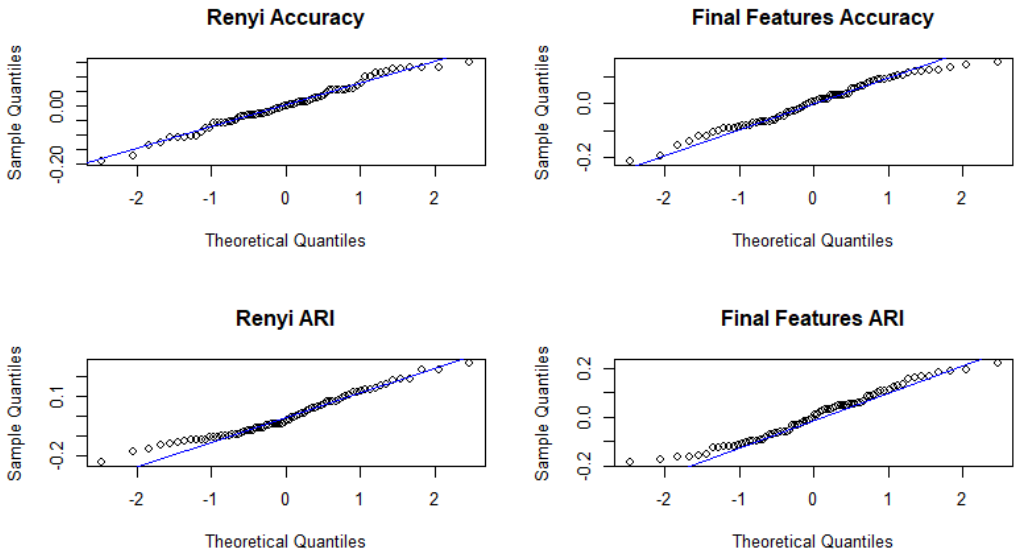


Figure 7.I: QQ plots of the residuals for accuracy and ARI across both feature sets.

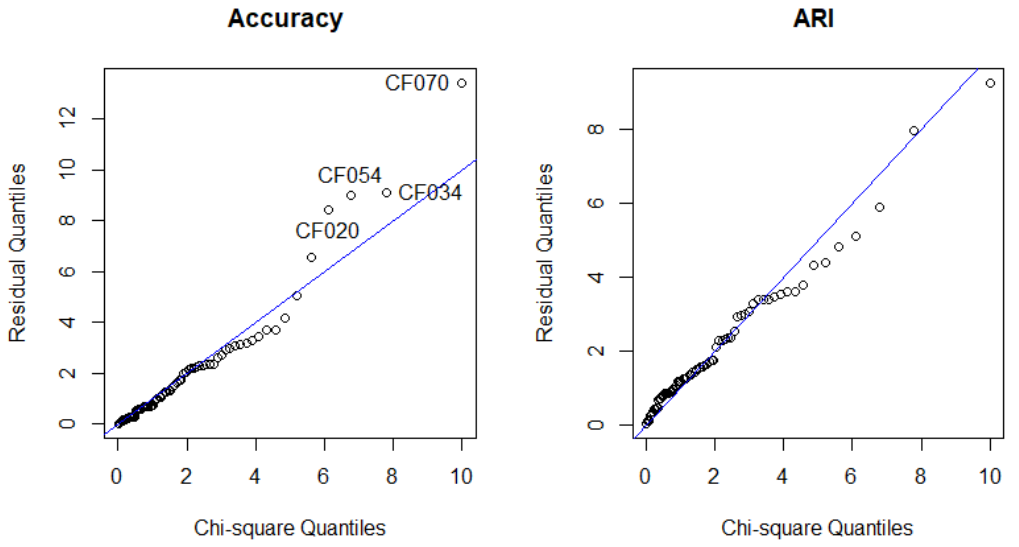


Figure 7.J: Chi-square QQ plot to assess multivariate normality of residuals for each performance measure.

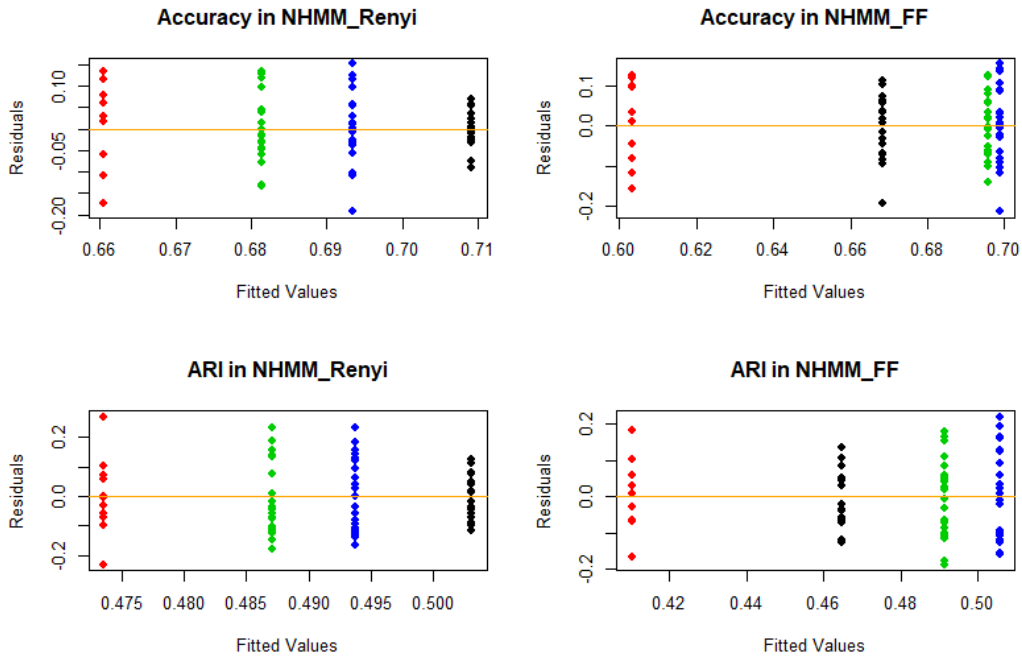


Figure 7.K: Residual vs fitted value plots of each performance measure across each feature set. Group colors: Green (Under 13:Low OSA), Black (Under 13:High OSA), Blue (Over 13:Low OSA) , Red (Over 13:High OSA)

MANOVA Results

Factor	≈ F-statistic	Dfn	Dfd	p-value
Age	0.293	2	69	0.747
OSA	6.49	2	69	0.0026
Age:OSA	1.58	2	69	0.214

Table 7.9: MANOVA table for NHMM accuracy

Factor	≈ F-statistic	Dfn	Dfd	p-value
Age	0.0434	2	69	0.958
OSA	5.84	2	69	0.0045
Age:OSA	1.04	2	69	0.360

Table 7.10: MANOVA table for NHMM ARI.

The first hypothesis tested in each of the 2-way MANOVAs was for the interaction effect. With p-values of 0.214 and 0.360 and a significance level of 5%, the data did not provide sufficient evidence of an interaction effect between age and OSA status.

Next, the hypothesis testing for the main effects of age and OSA was performed. Again, at a 5% significance level, the data did not provide sufficient evidence of a difference between the mean accuracies, or the ARI measures, of the age groups. However, the p-values of 0.0026 and 0.0045 for the main effect of OSA indicate that the data did provide sufficient evidence of a difference between the mean accuracies, and ARI measures, of the OSA groups. Since OSA status was found to have a significant effect in both MANOVA models, the analysis was extended to performing a 2-way ANOVA on each accuracy and ARI measure in each feature set. In total, four 2-way ANOVAs were performed. Only in the ANOVAs for the NHMMs that used the final features was the OSA status, or any factor effect, found to be significant. The results of the 2-way ANOVAs for NHMM accuracy and ARI using the final features are presented in Tables 7.11 and 7.12.

Factor	Df	Mean SS	F-statistic	p-value
Age	1	40.7	0.543	0.464
OSA	1	553.16	7.38	0.0083
Age:OSA	1	200.24	2.67	0.107
Residuals	70	74.92	-	-

Table 7.11: ANOVA table for NHMM accuracy using final features.

Factor	Df	Mean SS	F-statistic	p-value
Age	1	0.00029	0.026	0.872
OSA	1	0.0547	4.87	0.031
Age:OSA	1	0.0201	1.79	0.185
Residuals	70	0.0112	-	-

Table 7.12: ANOVA table for NHMM ARI using final features.

The first hypothesis tested in the ANOVAs was for the interaction effect between age and OSA. At a 5% significance level, the data did not provide evidence to suggest an interaction effect between the age and OSA groups on the mean accuracies and ARI values. Similarly, for the main effect of age, the data did not provide sufficient evidence of a difference between the mean accuracies and ARI values of the age groups. Lastly, the hypothesis test for main effect of OSA found that, at a 5% significance level, the data provided sufficient evidence there was a difference between the mean accuracies, and the mean ARIs, of patient OSA groups. Since

OSA was found to be significant in both ANOVAs presented here, a Tukey's HSD test (Tukey, 1949) was performed with an overall significance level of 5%.

Measure	OSA low - OSA high	Lower Bound	Upper Bound	p-value
Accuracy	5.50	1.41	9.6	0.009
ARI	0.0547	0.0047	0.105	0.033

Table 7.13: Table of results from Tukey's honest symmetric differences test.

The results of the Tukey's HSD tests are presented in Table 7.13. The data showed, with 95% confidence, the mean accuracy of patients with low OSA status is between 1.41% and 9.6% higher than patients with high OSA status. Furthermore, with 95% confidence, the mean ARI measure for patients with low OSA status is between 0.0047 and 0.105 higher than patients with high OSA status. This means that the classification performance of the NHMM using the final features is sensitive to patient OSA status. Furthermore, the findings indicate that, for methods clustering the individual patient's sleep epochs, care should be taken when analyzing classification performance of patients versus healthy subjects, since patient status is the primary recommendation for a sleep PSG, which means that for a classification algorithm to be successful in a clinical setting, it must be robust in relation to patient status.

7.6 Extended HMM Analysis

A key benefit of the HMM and, by extension, NHMM is that not only can they be effective for sleep state classification, but inspection of their transition matrices can also provide insight into the dynamics of a patient's sleep. The HMM transition matrices of CF003 and CF057 from Section 7.4 are presented in Tables 7.14 and 7.15. For simplicity, the transition matrices of the corresponding NHMM performances of patients CF003 and CF057 are not presented in this work, as the transition probabilities change with time and would require a transition matrix to be presented for every t.

Patient CF003: Renyi Entropy Features					
States	NREM 1	NREM 2	NREM 3	REM	Wake
NREM 1	0	0	0.333	0.333	0.333
NREM 2	0.008	0	0.012	0.098	0.882
NREM 3	0	0.25	0.25	0.25	0.25
REM	0.985	0	0	0	0.015
Wake	0	.333	0.333	0.333	0

Table 7.14: Transition matrix of HMM using Renyi entropy features for CF003. Values are the probabilities for transition to row state to column state.

The transition probabilities for CF003 of the HMM that used the Renyi entropy features indicated that when CF003 was in NREM 2 sleep, they had a high probability of transitioning to the wake state but then would transition back to NREM 2, NREM 3, or REM sleep. Also, patient CF003 had equal probabilities to transition to any state other than NREM 1 when in NREM 3, but if CF003 was in the REM state a transition to the NREM 1 state was almost certain. More interesting is the probabilities in Table 7.14 that are 0, like for the transitions from NREM 1 to NREM 2, as these are thought to be successive sleep states that help a person transition to deep sleep. Furthermore, the majority of transition probabilities that pertain to staying in the current state (the diagonal) are 0, except for NREM 3. This means that when in sleep states other than NREM 3, CF003 was found to certainly transition to another state.

Patient CF057: Final Features					
States	NREM 1	NREM 2	NREM 3	REM	Wake
NREM 1	0	0.27	0.558	0.11	0.062
NREM 2	0.16	0	0.022	0	0.962
NREM 3	0.967	0	0.027	0	0.006
REM	0	0	0.051	0.910	0.039
Wake	0	0.333	0.333	0.333	0

Table 7.15: Transition matrix of HMM using the final selected features for patient CF057. Values are the probabilities for transition to row state to column state.

The transition probabilities for patient CF057 of the HMM that used the final selected features tells a slightly different story. Patient CF057 was found to only have a high probability of transitioning to the wake state when in NREM 2, but

had the same equal probabilities of transitioning from the wake state to NREM 2, NREM 3, and REM states, like patient CF003. The transition probabilities between states NREM 1 and NREM 2 are larger than for patient CF003. This means that patient CF057 was found more likely to alternate between NREM 1 and NREM 2 states, but again had 0 probability of staying NREM 1 or NREM 2 for consecutive epochs. Furthermore, patient CF057 was more likely to transition to lighter sleep than staying in NREM 3, but when patient CF057 hit the REM state there was a high probability to stay in the REM state for consecutive epochs.

The transition matrices presented in this work could be further explored and analyzed with greater scrutiny, especially for a comparative analysis of the sleep dynamics of between patients with Low OSA status and patients with High OSA status. Similarly, a comparative analysis between patients 12 years or younger and those 13 years or older may provide experts with deeper insight as to how transitioning between sleep states changes with age. Moreover, these possible analyses might provide a starting point for building HMMs and NHMMs that can forecast the sleep states of new patients.

Chapter 8

Conclusion

8.1 Results

The first result, although somewhat unexpected, of this work was the range of Hurst exponent values for EEG and non-EEG PSG channels. As stated in Chapter 4, Vorobyov and Cichocki (2002, p. 296) reported that "the Hurst exponent takes values between 0.70-0.76 for most human phenomena" and the study by Bian et al. (2006) used a range of 0.7 - 0.9 to identify source signals that pertained to neural activity. In this work, the Hurst exponent was found to take a different range for EEG signals, 0.6-0.8, and, for all human phenomena included, the Hurst exponents of the non-EEG information was typically below 0.6, except for the abdomen EMG and EOG channels. The difference between the range found by Bian et al. (2006) and this work could be due to other factors, such as the sampling frequency of the PSG, the sleep laboratory and equipment used to perform the PSG, and/or possibly even the demographic groups of the subjects included in the study.

The random forest classification performance for the CF00N data was lower than what was found by the studies by Fraiwan et al. (2012) and da Silveira et al. (2017) that used the *Sleep EDF 2002* and *Sleep EDF Expaned* data sets, however, this work extended the cross validation analysis to a LOPOCV. In the LOPOCV it

was found that the classification performance measures decreased when the training data did not contain epochs from the same patients in the testing data. On the upside, there was no significant difference in the mean classification performance measures between the age and OSA status groups. The same could not be said for the NHMMs that used the final selected features, as patients in the High OSA status group had a lower mean performance measures than patients in the Low OSA group.

8.2 HMM and NHMM vs Random Forest

The classification results of Chapter 6 found that in the random forest LOPOCV the mean classification performance of all patients was best when using the DWC features, while the mean performance when using the final selected features was not far behind. Besides the non-EEG features, the Renyi entropy features had the worst mean classification performance in the random forest LOPOCV analysis. In Chapter 7, the NHMM was found, on average, to improve classification performance over the HMM using either the Renyi entropy or final selected features. Before the comparison between the classification performances of the random forests with the HMM and NHMM can be made, it should be noted that even though the goal of both approaches was to classify the epochs of individual patients, there is a key difference between the approaches. The random forest is a supervised method where the algorithm uses observations with known class labels to learn rules for classifying observations with unknown labels. Furthermore, the researcher has complete control in a cross validation setting over what information is used to train the random forest and what information is used to verify it, whereas the HMM and NHMM in this work are semi-supervised. They are semi-supervised because the HMM and NHMM were given a specified number of states, 5, to cluster the data.

The scatter plots in Figure 8.A compare the accuracies for individual patients of the random forest and the HMM or NHMM using the Renyi entropy and final selected features. The line in the plots is where the accuracies for random forest

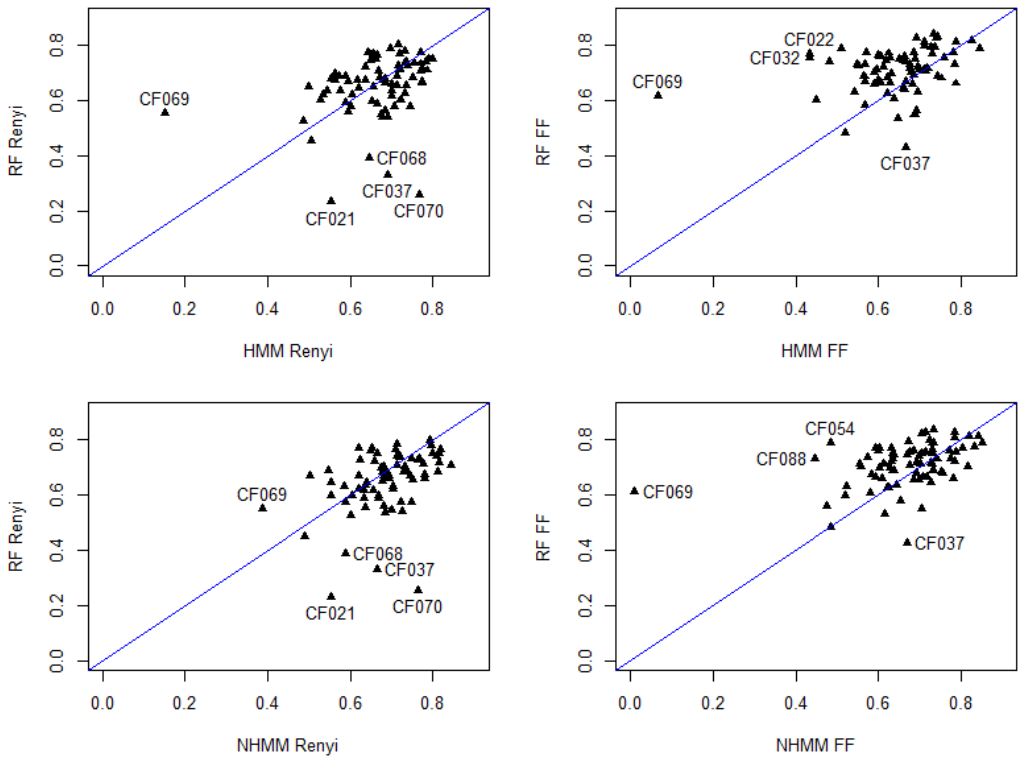


Figure 8.A: Accuracy scatter plots comparing random forest LOPOCV classification to HMM and NHMM

and HMM or NHMM would be equal. The first major difference was the classification performance of patient CF069, which stems from the key difference between the random forest and the HMM and NHMM approaches. When the Renyi entropy features were used, the random forest accuracies of patients CF021, CF037, CF068 and CF070 were found to be quite poor compared with the other patients, but the HMM and NHMM were able to classify these patients' epochs with accuracies more consistent with all other patients. Moreover, there are more patients below the lines of the comparative scatter plots for the Renyi entropy features, which indicates that the HMM and NHMM were more accurate across individual patients for this feature set. However, in the scatter plots that pertain to the Renyi entropy features the random forest appeared to have performed better than HMM and NHMM, since the majority of observations were above the line. Despite this trade off in better classification performance between random forest and HMM or NHMM across feature

sets, the HMM and NHMM provide insight into the dynamics of the sleep state dependence structure via the transition probabilities, which is something random forest and many other supervised algorithms cannot do. Understanding the dependence structure between sleep states may have more benefits than just improving sleep state classification: it may provide sleep experts with new information about the dependence structure of sleep states for different age, gender, and/or patient groups.

8.3 Future Work

The CF00N data presented in this work has many possibilities for future studies. The first place the author would start is right at the beginning: instead of band pass filtering and then forward-backward filtering with the Butterworth filter, the author would skip the latter to retain as much information as possible before cleaning the data with the ICA. To add to the previous statement, the author would consider performing the artifact removal using Matlab instead of `R`. The reason for this being that Matlab has many plug-ins available that are better suited to handle EEG data, so that, instead of cleaning each epoch individually, the author could clean the entire recording all at once.

All of the ideas mentioned above would have big implications on the effectiveness of the statistical features to discriminate against sleep states. The statistical features calculated in this work have been proven effective for the CF00N data, but there might be room for improvement. The first one being the statistical features of the non-EEG channels. There was relevant information present in those PSG channels that was not captured by the features calculated in this work. The wavelet transformations used to calculate the Renyi entropy and DWC moments features could be used in a similar manner for the non-EEG channels. Furthermore, the frequency bands used by Fraiwan et al. (2012) could be refined, that is, use smaller frequency bands and more of them to cover the range 0-35Hz. This refinement could lead to a better distinction between which frequencies are dominating

the EEG signal in an epoch and help algorithms better discriminate between sleep states. Along with the refinement, exploration using Daubechies wavelets of varying order may also find features that are better able to discriminate between sleep states.

Lastly, further HMM and NHMM analysis using new features that better discriminate between sleep states could be used to construct an HMM or NHMM that can accurately predict the sleep states of a new patient. Also, instead of using 30s epochs, use features calculated on smaller epochs, say 1s, and then model the data with HMM or NHMM to determine when the transitions between sleep states are happening and how long the patient stays in that sleep state, which would give a better estimate of the proportion of REM and NREM sleep for humans and non-humans. This might also provide better estimates for the transition probabilities and, consequently, a better understanding of the dynamics of sleep cycles in healthy subjects and patients.

References

Acharya, U. R., Chua, E. C.-P., Chua, K. C., Min, L. C., & Tamura, T. (2010). Analysis and automatic identification of sleep stages using higher order spectra. *International Journal of Neural Systems*, *20*(06), 509-521. Retrieved from https://doi.org/10.1142/S0129065710002589 (PMID: 21117273) doi: 10.1142/S0129065710002589

Ailliot, P., Bessac, J., Monbet, V., & Pène, F. (2015). Non-homogeneous hidden markov-switching models for wind time series. *Journal of Statistical Planning and Inference*, *160*, 75 - 88. Retrieved from http://www.sciencedirect.com/science/article/pii/S0378375814002018 doi: https://doi.org/10.1016/j.jspi.2014.12.005

Baum, L. E., Petrie, T., Soules, G., & Weiss, N. (1970, 02). A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *Ann. Math. Statist.*, *41*(1), 164–171. Retrieved from https://doi.org/10.1214/aoms/1177697196 doi: 10.1214/aoms/1177697196

Bian, N.-Y., Wang, B., Cao, Y., & Zhang, L. (2006). Automatic removal of artifacts from eeg data using ica and exponential analysis. In J. Wang, Z. Yi, J. M. Zurada, B.-L. Lu, & H. Yin (Eds.), *Advances in neural networks - isnn 2006* (pp. 719–726). Berlin, Heidelberg: Springer Berlin Heidelberg.

Boostani, R., Karimzadeh, F., & Nami, M. (2017). A comparative review on sleep stage classification methods in patients and healthy individuals. *Computer Methods and Programs in Biomedicine*, *140*, 77 - 91. Retrieved from http://www.sciencedirect.com/science/article/pii/S0169260716308276 doi: https://doi.org/10.1016/j.cmpb.2016.12.004

Box, G. E. P. (1949, 12). A General Distribution Theory For a Class of Likelihood Criteria. *Biometrika*, *36*(3-4), 317-346. Retrieved from https://doi.org/10.1093/biomet/36.3-4.317 doi: 10.1093/biomet/36.3-4.317

Breiman, L. (2001, Oct 01). Random forests. *Machine Learning*, *45*(1), 5–32. Retrieved from https://doi.org/10.1023/A:1010933404324 doi: 10.1023/A:1010933404324

Bridle, J. S. (1990). Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In F. F. Soulié & J. Hérault (Eds.), *Neurocomputing* (pp. 227–236). Berlin, Heidelberg: Springer Berlin Heidelberg.

Chaumon, M., Bishop, D. V., & Busch, N. A. (2015). A practical guide to the selection of independent components of the electroencephalogram for artifact correction. *Journal of Neuroscience Methods*, *250*, 47 - 63. Retrieved from http://www.sciencedirect.com/science/article/pii/S0165027015000928 (Cutting-edge EEG Methods) doi: https://doi.org/10.1016/j.jneumeth.2015.02.025

Chen, Y., Zhu, X., & Chen, W. (2015, 08). Automatic sleep staging based on ecg signals using hidden markov models. In (Vol. 2015, p. 530-533). doi: 10.1109/EMBC.2015.7318416

Chokroverty, S. (2017). Sleep disorders medicine. In (pp. 5–27). Springer.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, *20*(1), 37-46. Retrieved from https://doi.org/10.1177/001316446002000104 doi: 10.1177/001316446002000104

Cushner, K., Fish, B., & Wilson, A. (2019). *Tuck advancing better sleep.* http://www.tuck.com/sleep-spindles.

Danker-Hopfe, H., Anderer, P., Zeitlhofer, J., Boeck, M., Dorn, H., Gruber, G., ... Dorffner, G. (2008). Interrater reliability for sleep scoring according to the rechtschaffen & kales and the new aasm standard. *Journal of Sleep Research*, *18*(1), 74-84. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-2869.2008.00700.x doi: 10.1111/j.1365-2869.2008.00700.x

da Silveira, T. L. T., Kozakevicius, A. J., & Rodrigues, C. R. (2017, Feb 01). Single-channel eeg sleep stage classification based on a streamlined set of statistical features in wavelet domain. *Medical & Biological Engineering & Computing*, *55*(2), 343–352. Retrieved from https://doi.org/10.1007/s11517-016-1519-4 doi: 10.1007/s11517-016-1519-4

Delorme, A., & Makeig, S. (2004). Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis. *Journal of Neuroscience Methods*, *134*, 9–21.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, *39*(1), 1–38.

Ephraim, Y., & Merhave, N. (2002). Hidden markov processes. *IEEE Transactions on Information Theory*.

Flexer, A., Dorffner, G., Sykacekand, P., & Rezek, I. (2002, 03). An automatic, continuous and probabilistic sleep stager based on a hidden markov model.

Applied Artificial Intelligence, *16*. doi: 10.1080/088395102753559271

Flexer, A., Gruber, G., & Dorffner, G. (2005). A reliable probabilistic sleep stager based on a single eeg signal. *Artificial Intelligence in Medicine*, *33*(3), 199 - 207. Retrieved from http://www.sciencedirect.com/science/article/pii/S093336570400079X doi: https://doi.org/10.1016/j.artmed.2004.04.004

Forney, G. D. (1973, March). The viterbi algorithm. *Proceedings of the IEEE*, *61*(3), 268-278. doi: 10.1109/PROC.1973.9030

Fraiwan, L., Lweesy, K., Khasawneh, N., Fraiwan, M., Wenz, H., & Dickhaus, H. (2010, 01). Classification of sleep stages using multi-wavelet time frequency entropy and lda. *Methods of information in medicine*, *49*, 230-7. doi: 10.3414/ME09-01-0054

Fraiwan, L., Lweesy, K., Khasawneh, N., Wenz, H., & Dickhaus, H. (2012). Automated sleep stage identification system based on time–frequency analysis of a single eeg channel and random forest classifier. *Computer Methods and Programs in Biomedicine*, *108*(1), 10 - 19. Retrieved from http://www.sciencedirect.com/science/article/pii/S0169260711003105 doi: https://doi.org/10.1016/j.cmpb.2011.11.005

G Doroshenkov, L., Konyshev, V., & Selishchev, S. (2007, 01). Classification of human sleep stages based on eeg processing using hidden markov models. *Meditsinskaia tekhnika*, *41*, 24-8. doi: 10.1007/s10527-007-0006-5

Gneiting, T., & Schlather, M. (2004). Stochastic models that separate fractal dimension and the hurst effect. *SIAM Review*, *46*(2), 269–282. Retrieved from http://www.jstor.org/stable/20453506

Goldberger, A. L., Amaral, L. A. N., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., ... Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physio-

logic signals. *Circulation*, *101*(23), e215–e220. (Circulation Electronic Pages: http://circ.ahajournals.org/content/101/23/e215.full PMID:1085218; doi: 10.1161/01.CIR.101.23.e215)

Gupta, M. R., & Chen, Y. (2011, March). Theory and use of the em algorithm. *Found. Trends Signal Process.*, *4*(3), 223–296. Retrieved from http://dx.doi.org/10.1561/2000000034 doi: 10.1561/2000000034

Güneş, S., Polat, K., & Şebnem Yosunkaya. (2010). Efficient sleep stage recognition system based on eeg signal using k-means clustering based feature weighting. *Expert Systems with Applications*, *37*(12), 7922 - 7928. Retrieved from http://www.sciencedirect.com/science/article/pii/S095741741000343X doi: https://doi.org/10.1016/j.eswa.2010.04.043

Hassan, A. R., & Subasi, A. (2017). A decision support system for automated identification of sleep stages from single-channel eeg signals. *Knowledge-Based Systems*, *128*, 115 - 124. Retrieved from http://www.sciencedirect.com/science/article/pii/S095070511730206X doi: https://doi.org/10.1016/j.knosys.2017.05.005

Helwig, N. E. (2018). ica: Independent component analysis [Computer software manual]. Retrieved from https://CRAN.R-project.org/package=ica (R package version 1.0-2)

Ho, T. K. (1995). Random decision forests. In *Proceedings of the third international conference on document analysis and recognition (volume 1) - volume 1* (pp. 278–). Washington, DC, USA: IEEE Computer Society. Retrieved from http://dl.acm.org/citation.cfm?id=844379.844681

Ho, T. K. (1998, August). The random subspace method for constructing decision forests. *IEEE Trans. Pattern Anal. Mach. Intell.*, *20*(8), 832–844. Re-

trieved from https://doi.org/10.1109/34.709601 doi: 10.1109/34.709601

Hsu, Y.-L., Yang, Y.-T., Wang, J.-S., & Hsu, C.-Y. (2013). Automatic sleep stage recurrent neural classifier using energy features of eeg signals. *Neurocomputing*, *104*, 105 - 114. Retrieved from http://www.sciencedirect.com/science/article/pii/S0925231212008387 doi: https://doi.org/10.1016/j.neucom.2012.11.003

Hughes, J. P., Guttorp, P., & Charles, S. P. (1999). A non-homogeneous hidden markov model for precipitation occurrence. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, *48*(1), 15–30. Retrieved from http://www.jstor.org/stable/2680815

Hurst, H. E. (1951). Long-term storage capacity of reservoirs..

Hurst, H. E., Black, R. P., & Simaika, Y. M. (1965). *Long-term storage : an experimental study / by h.e. hurst, r.p. black, y.m. simaika* [Book]. Constable London.

Hyvarinen, A. (1999, May). Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, *10*(3), 626-634. doi: 10.1109/72.761722

Hyvärinen, A., & Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural Networks*, *13*(4), 411 - 430. Retrieved from http://www.sciencedirect.com/science/article/pii/S0893608000000265 doi: https://doi.org/10.1016/S0893-6080(00)00026-5

Iber, C., Ancoli-Israel, S., Chesson, A., & Quan, S. (2007, 01). The aasm manual for the scoring of sleep and associated events: Rules, terminology and technical specifications. *Westchester, IL: American Academy of Sleep Medicine*.

Johnson, R., & Wichern, D. (2007). *Applied multivariate statistical analysis*.

Pearson Prentice Hall. Retrieved from https://books.google.ca/books?id=gFWcQgAACAAJ

Kemp, B., & A. C. Kamphuisen, H. (1986, 02). Simulation of human hypnograms using a markov chain model. *Sleep*, *9*, 405-14. doi: 10.1093/sleep/9.3.405

Koley, B., & Dey, D. (2012). An ensemble system for automatic sleep stage classification using single channel eeg signal. *Computers in Biology and Medicine*, *42*(12), 1186 - 1195. Retrieved from http://www.sciencedirect.com/science/article/pii/S0010482512001588 doi: https://doi.org/10.1016/j.compbiomed.2012.09.012

Lagona, F., Maruotti, A., & Picone, M. (2011). A non-homogeneous hidden markov model for the analysis of multi-pollutant exceedances data. In P. Dymarski (Ed.), *Hidden markov models* (chap. 10). Rijeka: IntechOpen. Retrieved from https://doi.org/10.5772/14749 doi: 10.5772/14749

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, *33*(1), 159–174. Retrieved from http://www.jstor.org/stable/2529310

Levene, H. (1960). Robust tests for equality of variances. In I. Olkin (Ed.), *Contributions to probability and statistics: Essays in honor of harold hotelling* (p. 278-292). Stanford: Stanford University Press. Retrieved from https://books.google.ca/books?id=ZUSsAAAAIAAJ

Liaw, A., & Wiener, M. (2002). Classification and regression by randomforest. *R News*, *2*(3), 18-22. Retrieved from https://CRAN.R-project.org/doc/Rnews/

Längkvist, M., Karlsson, L., & Loutfi, A. (2012). Sleep stage classification using unsupervised feature learning. Retrieved from https://login.ezproxy.library.ualberta.ca/login?url=https://search.ebscohost.com/login.aspx

?direct=true&db=edsoai&AN=edsoai.ocn802847842&site=eds-live&scope=site

McNicholas, P. (2017). *Mixture model-based classification*. New York: Chapman and Hall/CRC. Retrieved from https://doi.org/10.1201/9781315373577

Mognon, A., Jovicich, J., Bruzzone, L., & Buiatti, M. (2010). Adjust: An automatic eeg artifact detector based on the joint use of spatial and temporal features. *Psychophysiology*, *48*(2), 229-240. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-8986.2010.01061.x doi: 10.1111/j.1469-8986.2010.01061.x

Mousavi, S., Afghah, F., & Acharya, U. R. (2019, Jul). Sleepeegnet: Automated sleep stage scoring with sequence to sequence deep learning approach. *Plos One*, *14*(5). doi: 10.1371/journal.pone.0216456

Nolan, H., Whelan, R., & Reilly, R. (2010). Faster: Fully automated statistical thresholding for eeg artifact rejection. *Journal of Neuroscience Methods*, *192*(1), 152 - 162. Retrieved from http://www.sciencedirect.com/science/article/pii/S0165027010003894 doi: https://doi.org/10.1016/j.jneumeth.2010.07.015

Ocañ-Riola, R. (2005). Non-homogeneous markov processes for biomedical data analysis. *Biometrical Journal*, *47*(3), 369-376. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1002/bimj.200310114 doi: 10.1002/bimj.200310114

Pan, S.-T., Kuo, C.-E., Zeng, J.-H., & Liang, S.-F. (2012, Aug 21). A transition-constrained discrete hidden markov model for automatic sleep staging. *BioMedical Engineering OnLine*, *11*(1), 52. Retrieved from https://doi.org/10.1186/1475-925X-11-52 doi: 10.1186/1475-925X-11-52

Peker, M. (2016a). An efficient sleep scoring system based on eeg signal using complex-valued machine learning algorithms. *Neurocomputing*, *207*, 165 - 177. Retrieved from http://www.sciencedirect.com/science/article/pii/S0925231216303289 doi: https://doi.org/10.1016/j.neucom.2016.04.049

Peker, M. (2016b). A new approach for automatic sleep scoring: Combining taguchi based complex-valued neural network and complex wavelet transform. *Computer Methods and Programs in Biomedicine*, *129*, 203 - 216. Retrieved from http://www.sciencedirect.com/science/article/pii/S016926071600002X doi: https://doi.org/10.1016/j.cmpb.2016.01.001

Penny, W., & Roberts, S. (1999, 10). Gaussian observation hidden markov models for eeg analysis.

R Core Team. (2019). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from https://www.R-project.org/

Rabiner, L. R. (1989, Feb). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257-286. doi: 10.1109/5.18626

Rechtschaffen, A., & Kales, A. (1968). A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects: A. rechtschaffen and a. kales (editors). (public health service, u.s. government printing office, washington, d.c.).

Seifpour, S., Niknazar, H., Mikaeili, M., & Nasrabadi, A. M. (2018). A new automatic sleep staging system based on statistical behavior of local extrema using single channel eeg signal. *Expert Systems with Applications*, *104*, 277 - 293. Retrieved from http://www.sciencedirect.com/

science/article/pii/S095741741830160X doi: https://doi.org/10.1016/j.eswa.2018.03.020

Shannon, C. E. (1948, Oct). A mathematical theory of communication. *The Bell System Technical Journal*, *27*(4), 623-656. doi: 10.1002/j.1538-7305.1948.tb00917.x

Steinley, D. (2004a, 09). Properties of the hubert-arabie adjusted rand index. *Psychological methods*, *9*, 386-96. doi: 10.1037/1082-989X.9.3.386

Steinley, D. (2004b, 09). Properties of the hubert-arabie adjusted rand index. *Psychological methods*, *9*, 386-96. doi: 10.1037/1082-989X.9.3.386

Stepnowsky, C., Levendowski, D., Popovic, D., Ayappa, I., & Rapoport, D. M. (2013). Scoring accuracy of automated sleep staging from a bipolar electroocular recording compared to manual scoring by multiple raters. *Sleep Medicine*, *14*(11), 1199 - 1207. Retrieved from http://www.sciencedirect.com/science/article/pii/S1389945713002347 doi: https://doi.org/10.1016/j.sleep.2013.04.022

Sucar, L. (2015). *Probabilistic graphical models: Principles and applications*. Springer London LTD.

Supratak, A., Dong, H., Wu, C., & Guo, Y. (2017, Nov). Deepsleepnet: A model for automatic sleep stage scoring based on raw single-channel eeg. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *25*(11), 1998-2008. doi: 10.1109/TNSRE.2017.2721116

Terzano, M. G., Parrino, L., Sherieri, A., Chervin, R., Chokroverty, S., Guilleminault, C., ... Walters, A. (2001). Atlas, rules, and recording techniques for the scoring of cyclic alternating pattern (cap) in human sleep. *Sleep Medicine*, *2*(6), 537 - 553. Retrieved from http://www.sciencedirect.com/science/article/pii/S1389945701001496 doi: https://doi.org/10.1016/S1389-9457(01)00149-6

DREAMS. (2016). *The dreams subjects database.* Retrieved from http://www.tcfs.fpms.ac.be/devuyst/Databases/DatabaseSubjects/

Trevenen, M. L., Turlach, B. A., Eastwood, P. R., Straker, L. M., & Murray, K. (2019). Using hidden markov models with raw, triaxial wrist accelerometry data to determine sleep stages. *Australian & New Zealand Journal of Statistics*, *0*(0). Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1111/anzs.12270 doi: 10.1111/anzs.12270

Tukey, J. W. (1949). Comparing individual means in the analysis of variance. *Biometrics*, *5*(2), 99–114. Retrieved from http://www.jstor.org/stable/3001913

Venables, W. N., & Ripley, B. D. (2002). *Modern applied statistics with s* (Fourth ed.). New York: Springer. Retrieved from http://www.stats.ox.ac.uk/pub/MASS4 (ISBN 0-387-95457-0)

Visser, I., & Speekenbrink, M. (2010). depmixS4: An R package for hidden markov models. *Journal of Statistical Software*, *36*(7), 1–21. Retrieved from http://www.jstatsoft.org/v36/i07/

Viterbi, A. (1967, April). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, *13*(2), 260-269. doi: 10.1109/TIT.1967.1054010

Vorobyov, S., & Cichocki, A. (2002, Apr 01). Blind noise reduction for multisensory signals using ica and subspace filtering, with application to eeg analysis. *Biological Cybernetics*, *86*(4), 293–303. Retrieved from https://doi.org/10.1007/s00422-001-0298-6 doi: 10.1007/s00422-001-0298-6

Weiss, B., Clemens, Z., Bódizs, R., Vágó, Z., & Halász, P. (2009). Spatio-temporal analysis of monofractal and multifractal properties of the hu-

man sleep eeg. *Journal of Neuroscience Methods*, *185*(1), 116 - 124. Retrieved from http://www.sciencedirect.com/science/article/pii/S0165027009004038 doi: https://doi.org/10.1016/j.jneumeth.2009.07.027

Xu, H. (2005). *Classification of sleep stage based on eeg wave* (Unpublished master's thesis). The University of Chicago, Chicago, Illinois.

Yaghouby, F., Modur, P., & Sunderam, S. (2014, 11). Naive scoring of human sleep based on a hidden markov model of the electroencephalogram. *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2014*, 5028-5031. doi: 10.1109/EMBC.2014.6944754

Yang, M. C. K., & Hursch, C. J. (1973). The use of a semi-markov model for describing sleep patterns. *Biometrics*, *29*(4), 667–676. Retrieved from http://www.jstor.org/stable/2529133

Zucchini, W., MacDonald, I. L., & Langrock, R. (2016). *Hidden markov models for time series: An introduction using r,(2nd edition)*. Chapman & Hall/CRC, Boca Raton. Retrieved from https://doi.org/10.1201/b20790

Zung, W. W. K., Naylor, T. H., Gianturco, D. T., & Wilson, W. P. (1966). Computer simulation of sleep eeg patterns with a markov chain model. In J. Wortis (Ed.), *Recent advances in biological psychiatry: The proceedings of the twentieth annual convention and scientific program of the society of biological psychiatry, new york city, april 30–may 2,1965* (pp. 335–355). Boston, MA: Springer US. Retrieved from https://doi.org/10.1007/978-1-4899-7313-9-36 doi: 10.1007/978-1-4899-7313-9-36

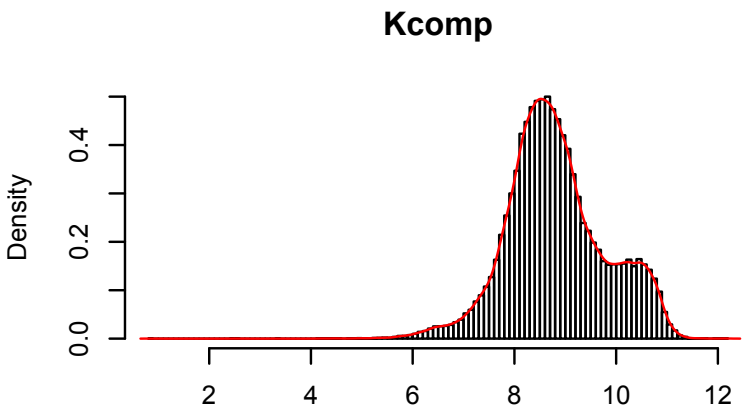
Özşen, S. (2012, 10). Classification of sleep stages using class-dependent sequential feature selection and artificial neural network. *Neural Computing and Applications*, *23*. doi: 10.1007/s00521-012-1065-4

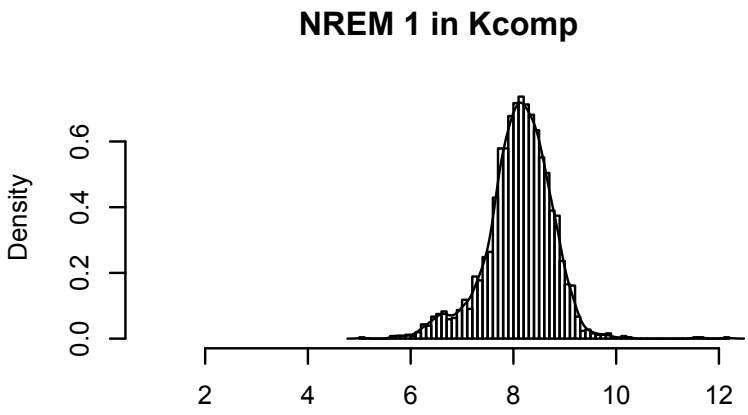
Šušmáková, K., & Krakovská, A. (2008). Discrimination ability of individual measures used in sleep stages classification. *Artificial Intelligence in Medicine*, *44*(3), 261 - 277. Retrieved from http://www.sciencedirect.com/science/article/pii/S0933365708000924 doi: https://doi.org/10.1016/j.artmed.2008.07.005

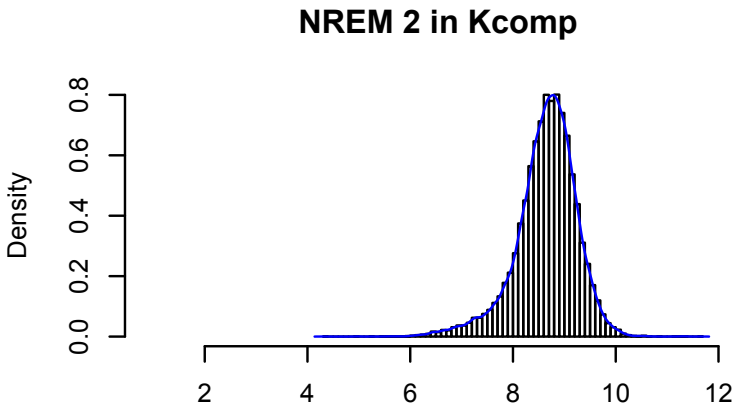
Appendices

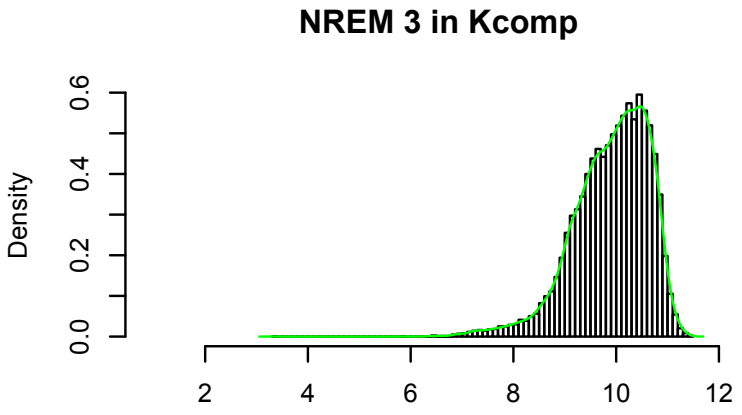
Appendix A.1

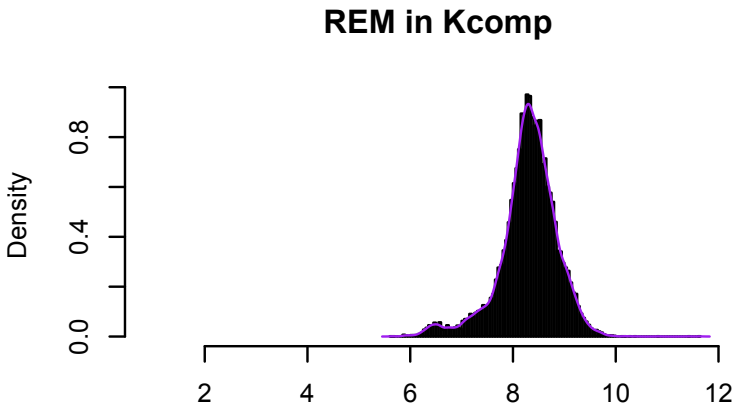
This appendix contains the density and QQ plots of the Renyi Entropy Features.

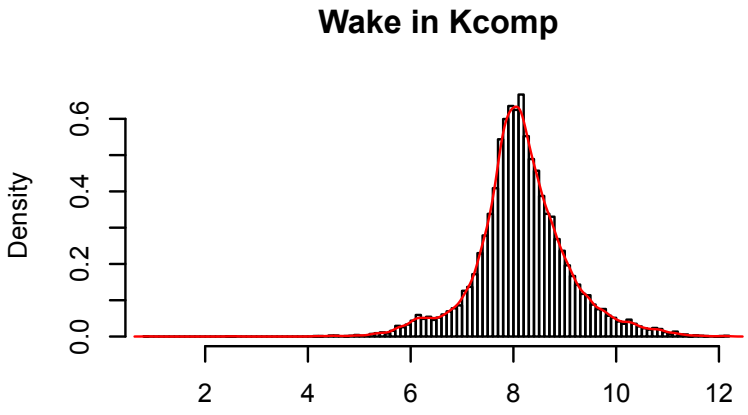

Kcomp
Density
0.0
0.2
0.4
2
4
6
8
10
12

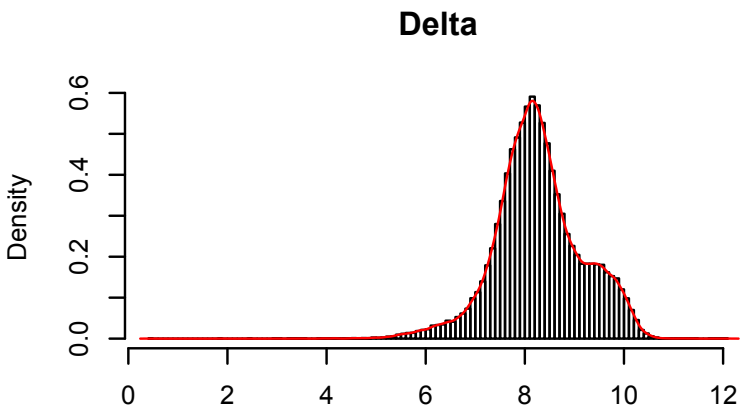

NREM 1 in Kcomp
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

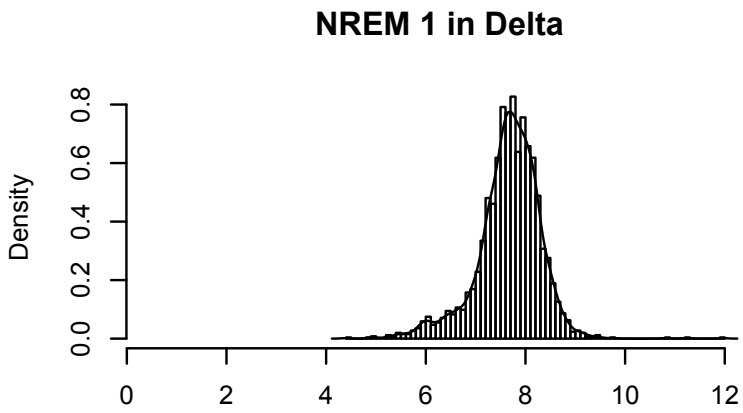

NREM 2 in Kcomp
Density
0.0
0.2
0.4
0.6
0.8
2
4
6
8
10
12

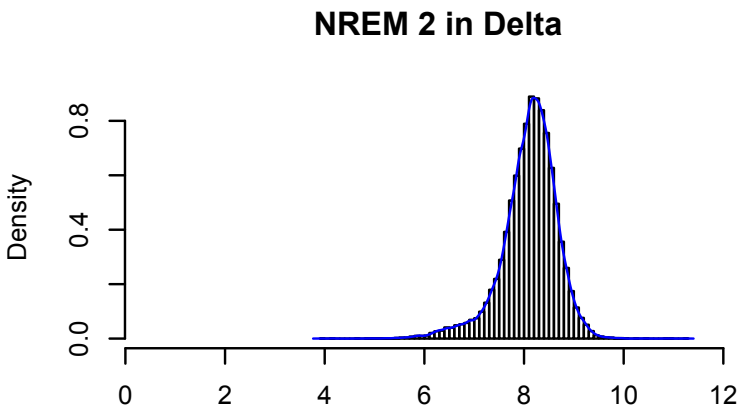

NREM 3 in Kcomp
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

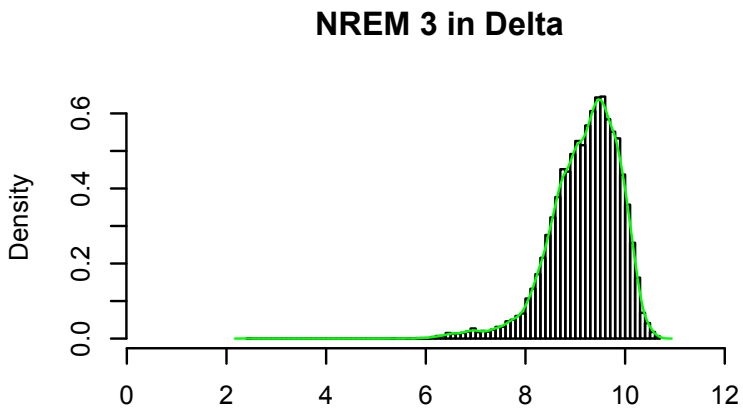

REM in Kcomp
Density
0.0
0.4
0.8
2
4
6
8
10
12

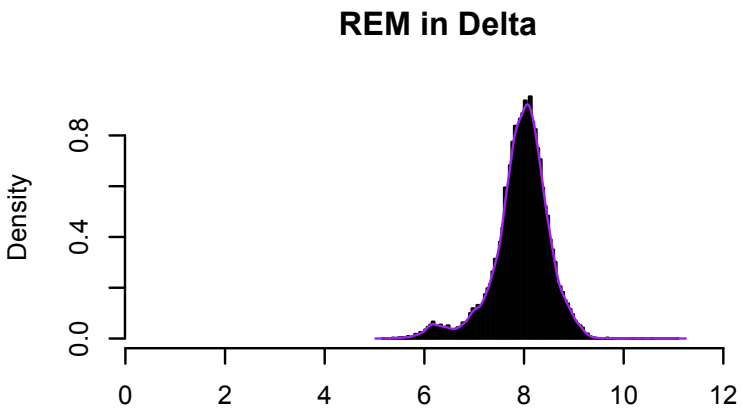

Wake in Kcomp
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

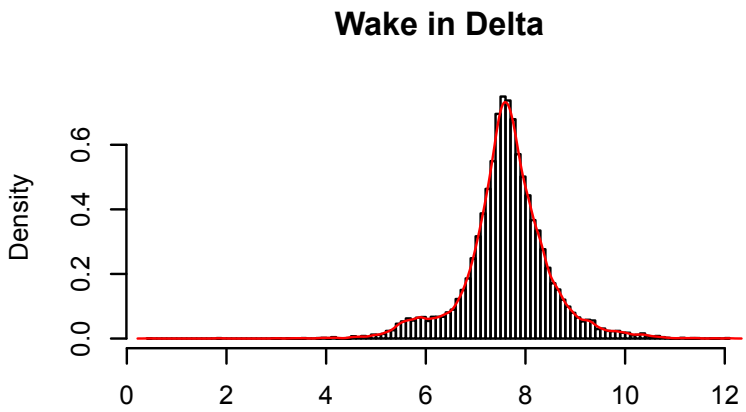

Delta
Density
0.0
0.2
0.4
0.6
0
2
4
6
8
10
12

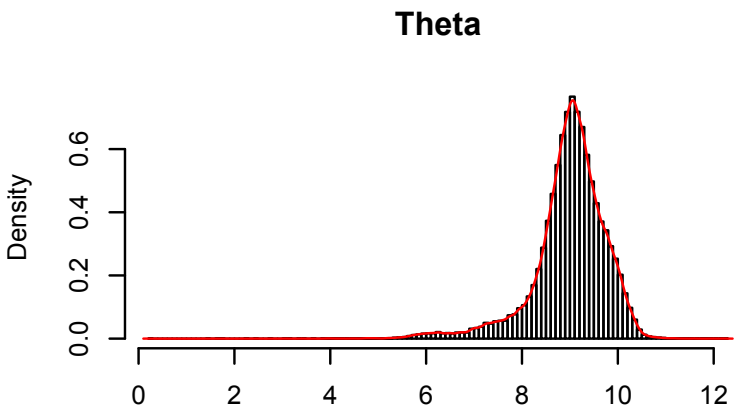

NREM 1 in Delta
Density
0.0
0.2
0.4
0.6
0.8
0
2
4
6
8
10
12

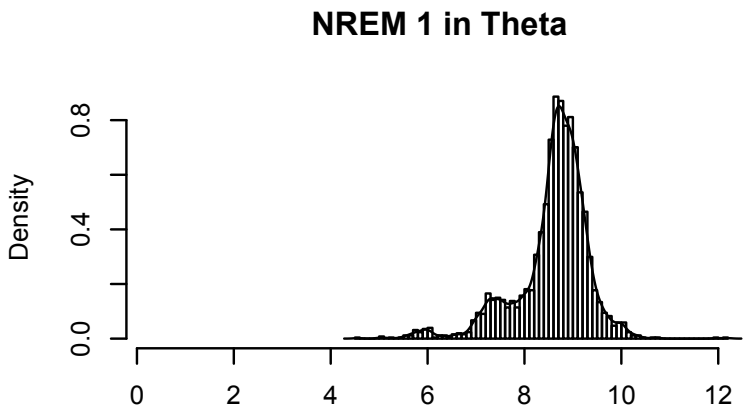

NREM 2 in Delta
Density
0.0
0.4
0.8
0
2
4
6
8
10
12

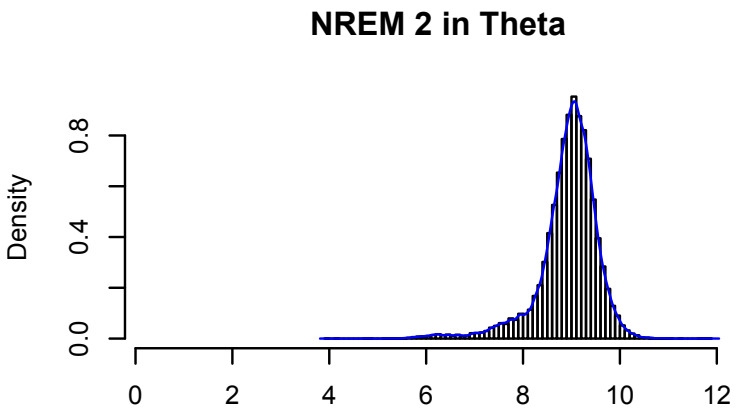

NREM 3 in Delta
Density
0.0
0.2
0.4
0.6
0
2
4
6
8
10
12

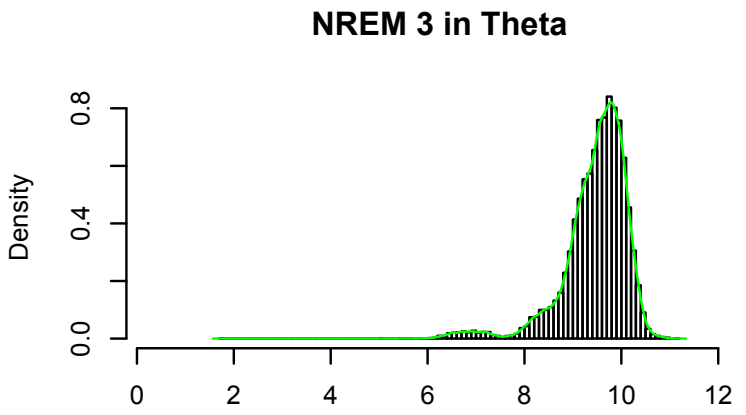

REM in Delta
Density
0.0
0.4
0.8
0
2
4
6
8
10
12

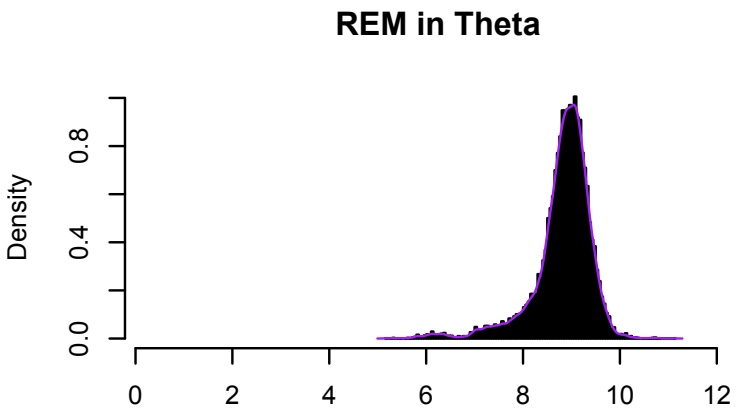

Wake in Delta
Density
0.0
0.2
0.4
0.6
0
2
4
6
8
10
12

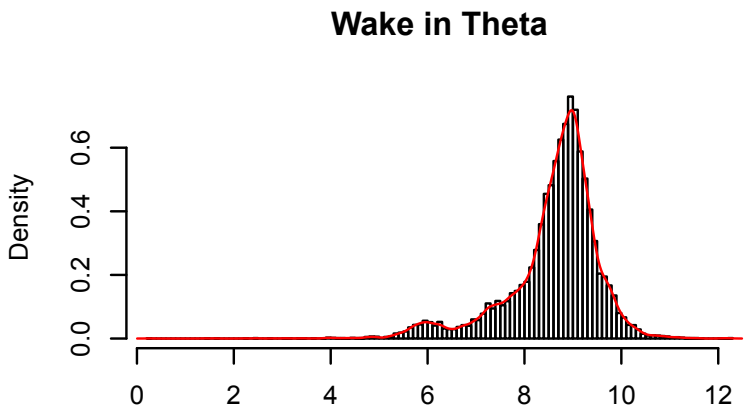

Theta
Density
0.0
0.2
0.4
0.6
0
2
4
6
8
10
12

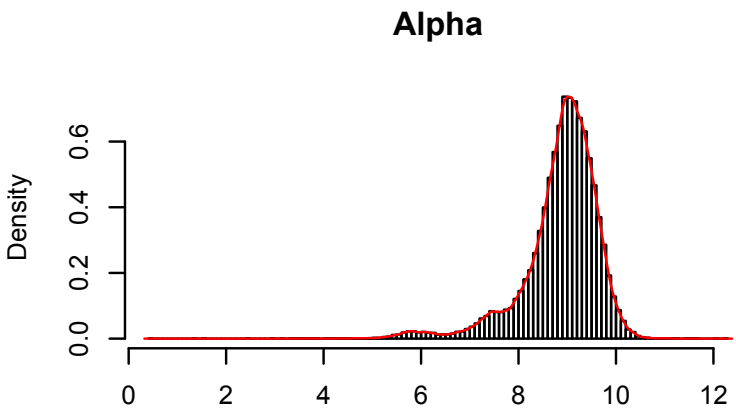

NREM 1 in Theta
Density
0.0
0.4
0.8
0
2
4
6
8
10
12

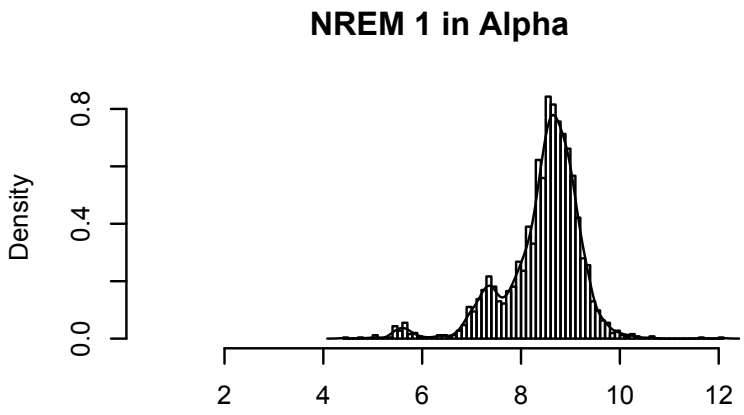

NREM 2 in Theta
Density
0.0
0.4
0.8
0
2
4
6
8
10
12

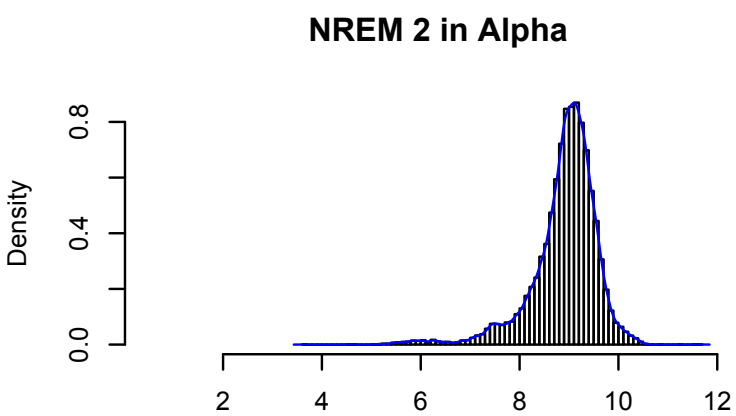

NREM 3 in Theta
Density
0.0
0.4
0.8
0
2
4
6
8
10
12

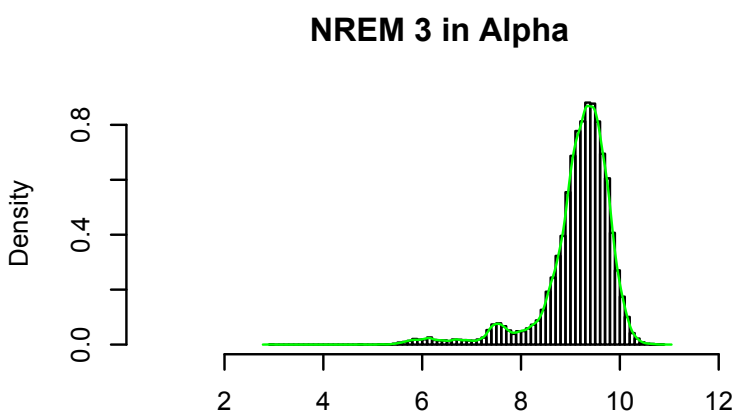

REM in Theta
Density
0.0
0.4
0.8
0
2
4
6
8
10
12

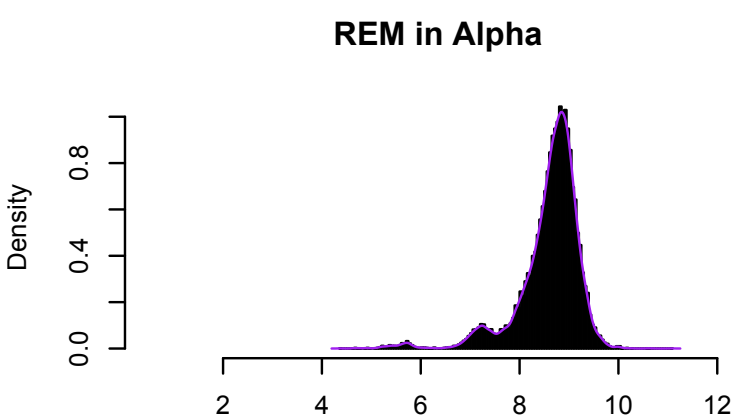

Wake in Theta
Density
0.0
0.2
0.4
0.6
0
2
4
6
8
10
12

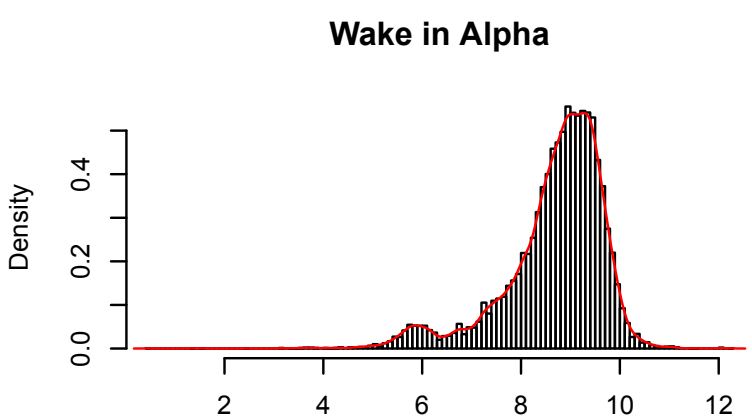

Alpha
Density
0.0
0.2
0.4
0.6
0
2
4
6
8
10
12

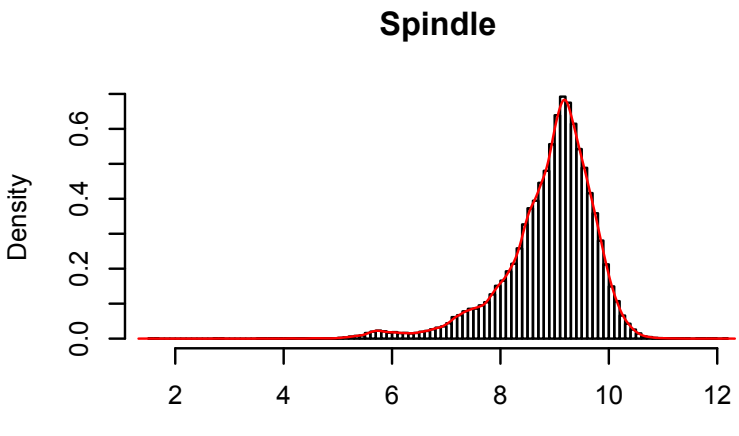

NREM 1 in Alpha
Density
0.0
0.4
0.8
2
4
6
8
10
12

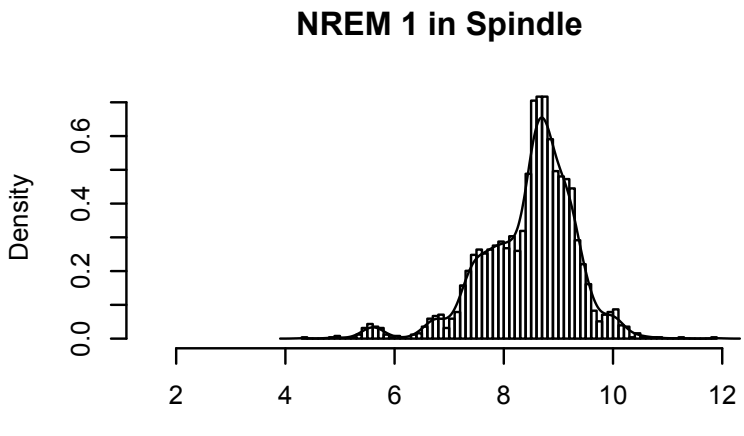

NREM 2 in Alpha
Density
0.0
0.4
0.8
2
4
6
8
10
12

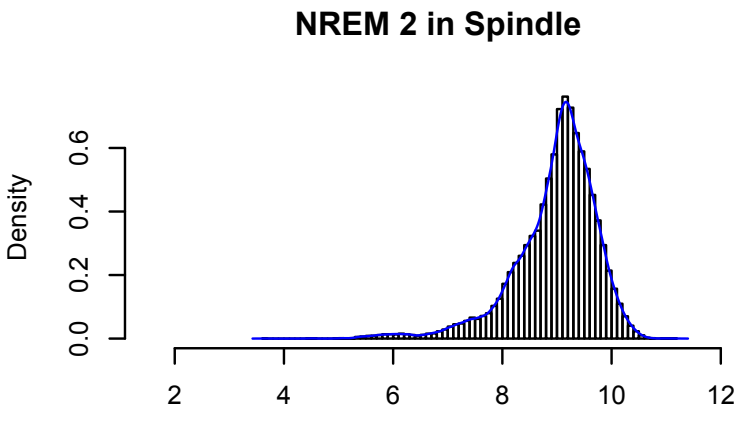

NREM 3 in Alpha
Density
0.0
0.4
0.8
2
4
6
8
10
12

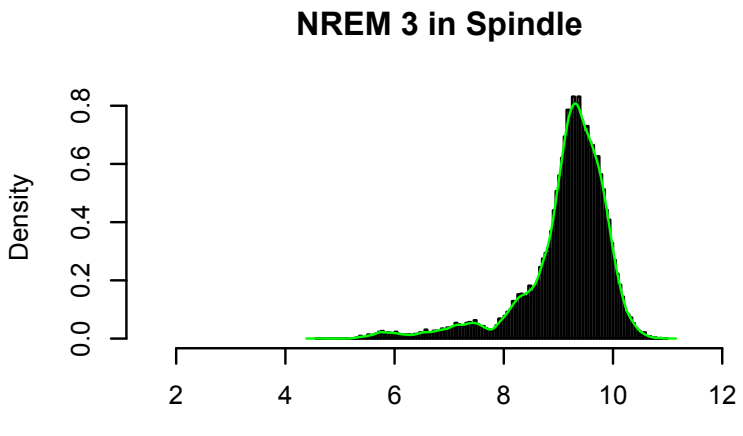

REM in Alpha
Density
0.0
0.4
0.8
2
4
6
8
10
12

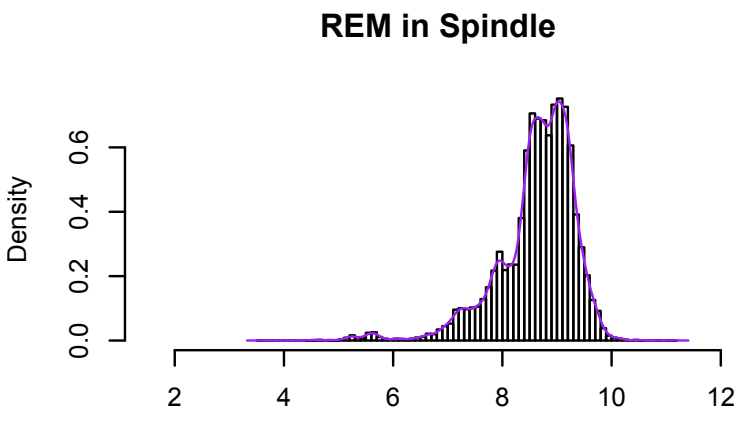

Wake in Alpha
Density
0.0
0.2
0.4
2
4
6
8
10
12

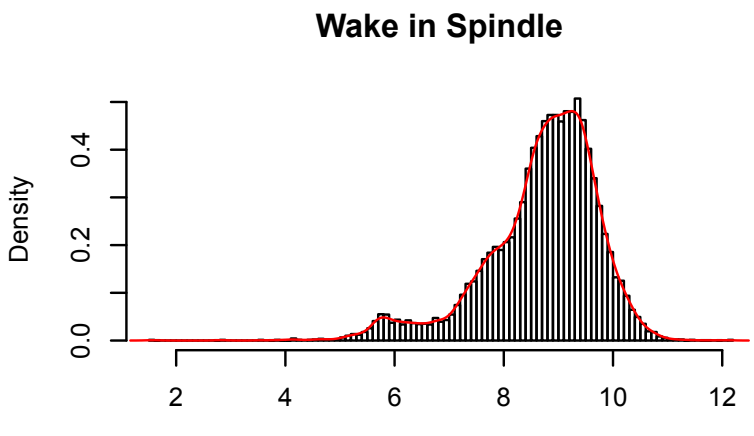

Spindle
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

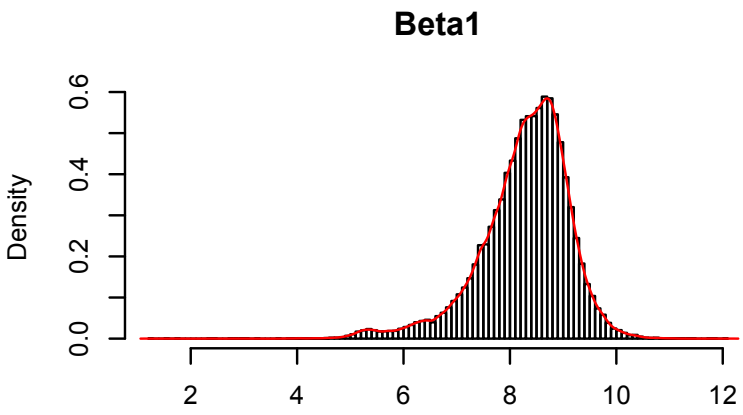

NREM 1 in Spindle
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

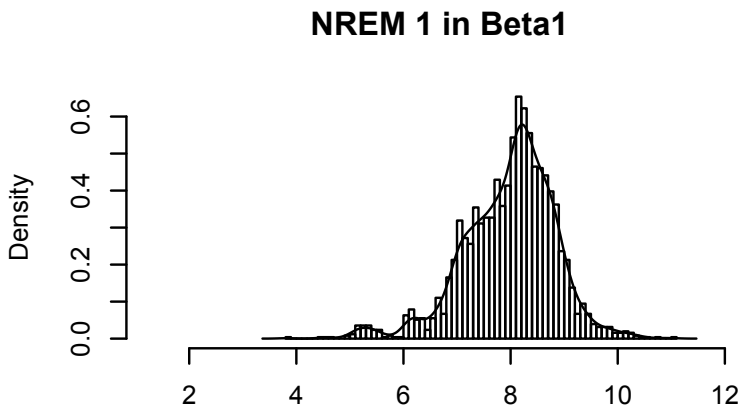

NREM 2 in Spindle
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

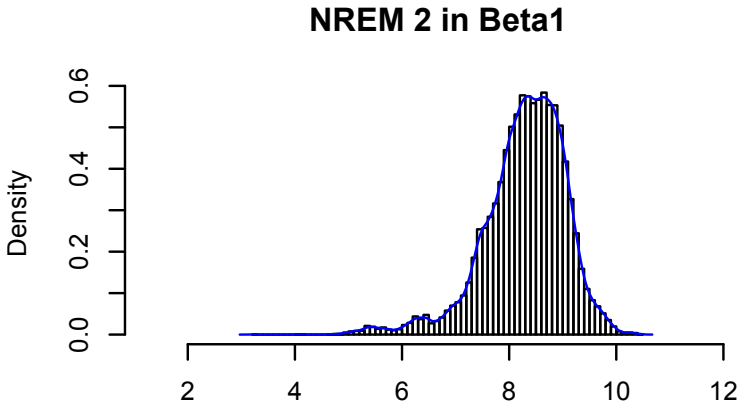

NREM 3 in Spindle
Density
0.0
0.2
0.4
0.6
0.8
2
4
6
8
10
12

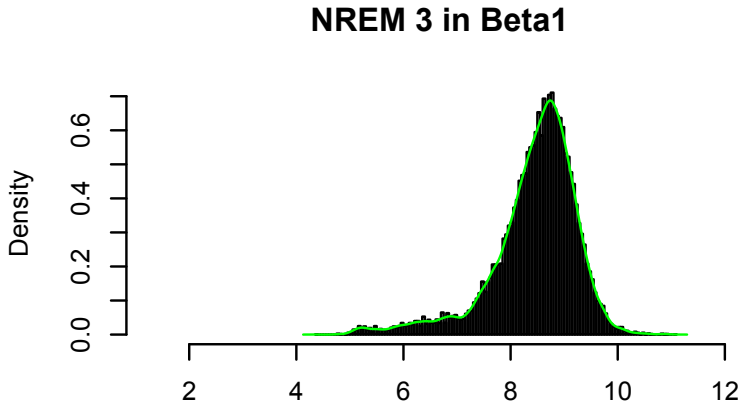

REM in Spindle
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

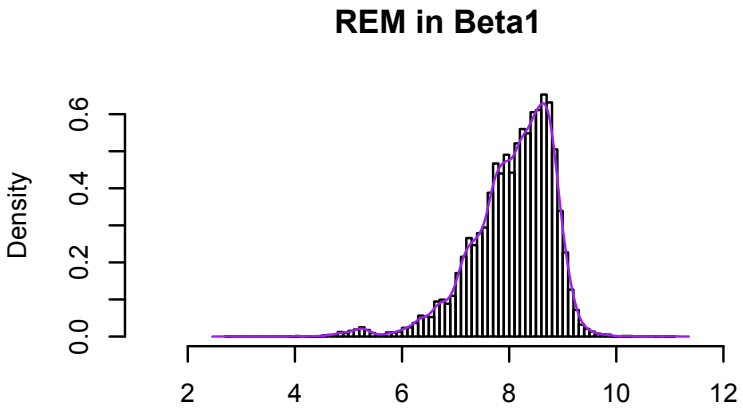

Wake in Spindle
Density
0.0
0.2
0.4
2
4
6
8
10
12

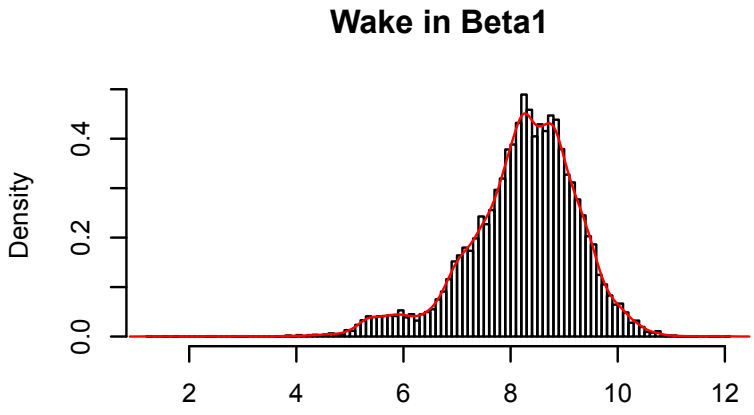

Beta1
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

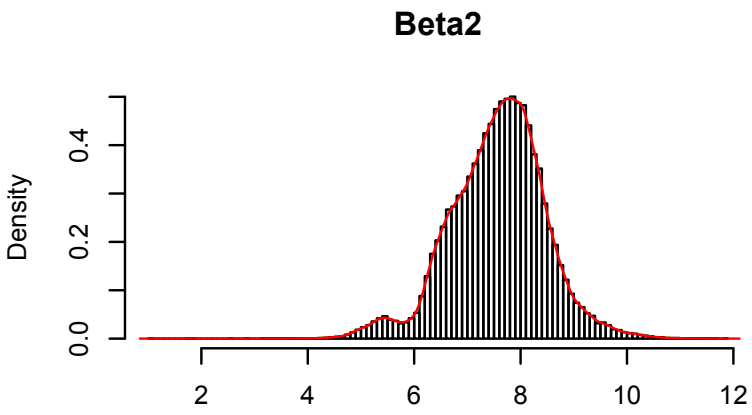

NREM 1 in Beta1
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

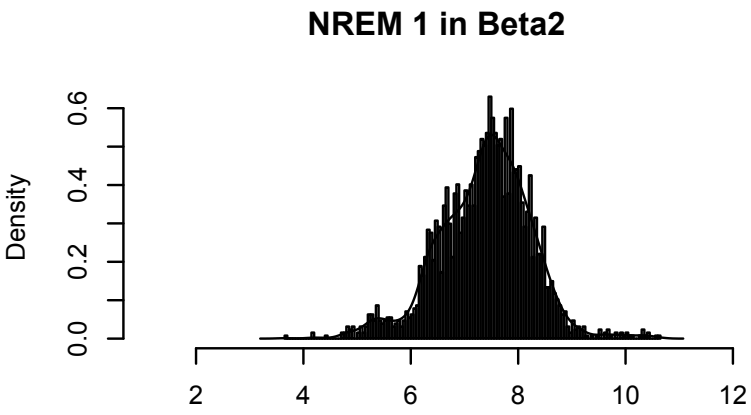

NREM 2 in Beta1
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

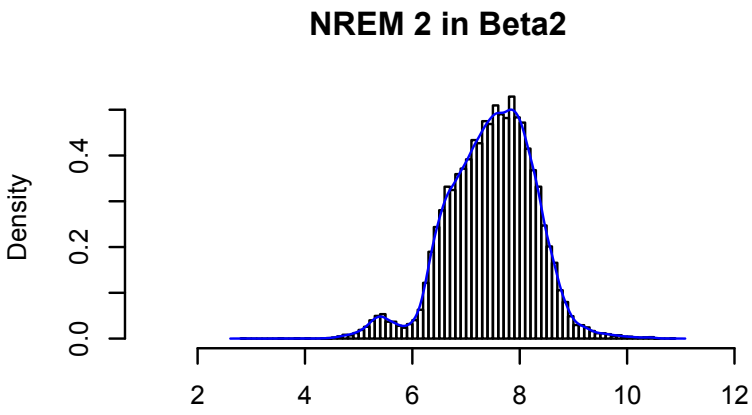

NREM 3 in Beta1
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

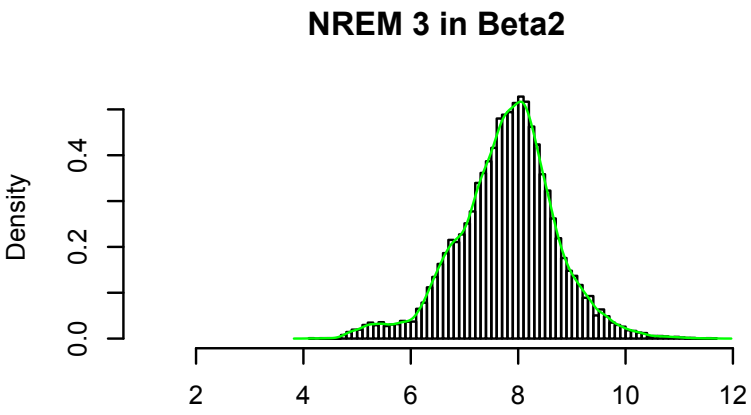

REM in Beta1
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

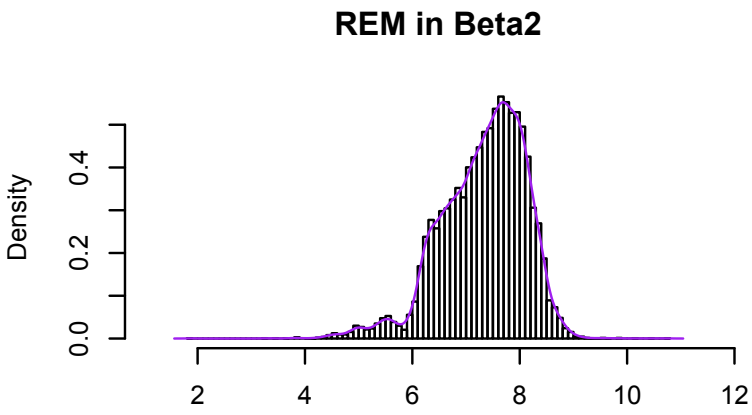

Wake in Beta1
Density
0.0
0.2
0.4
2
4
6
8
10
12

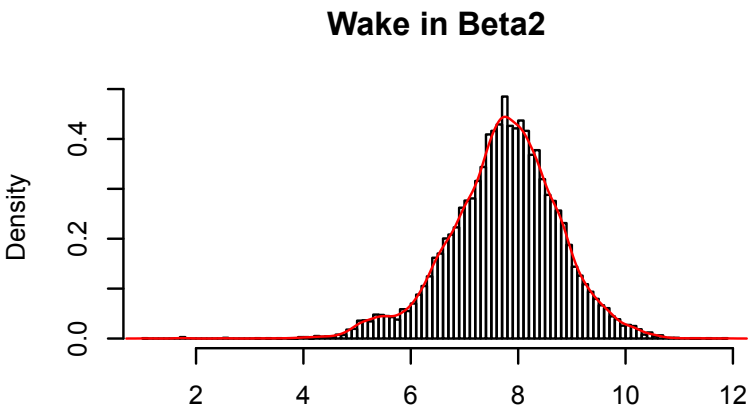

Beta2
Density
0.0
0.2
0.4
2
4
6
8
10
12

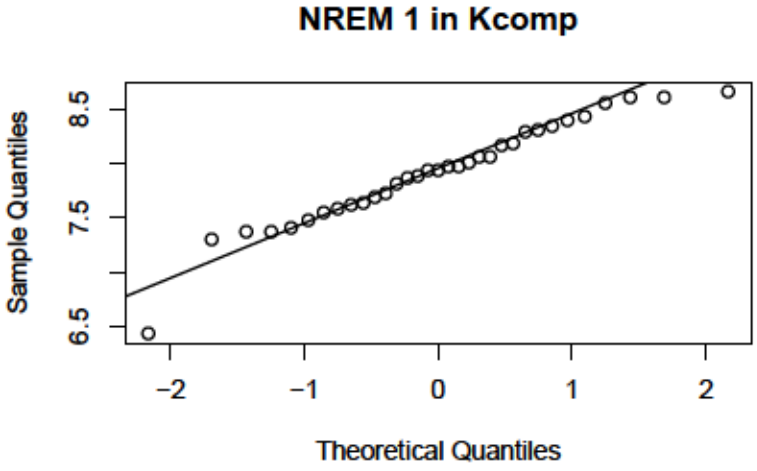

NREM 1 in Beta2
Density
0.0
0.2
0.4
0.6
2
4
6
8
10
12

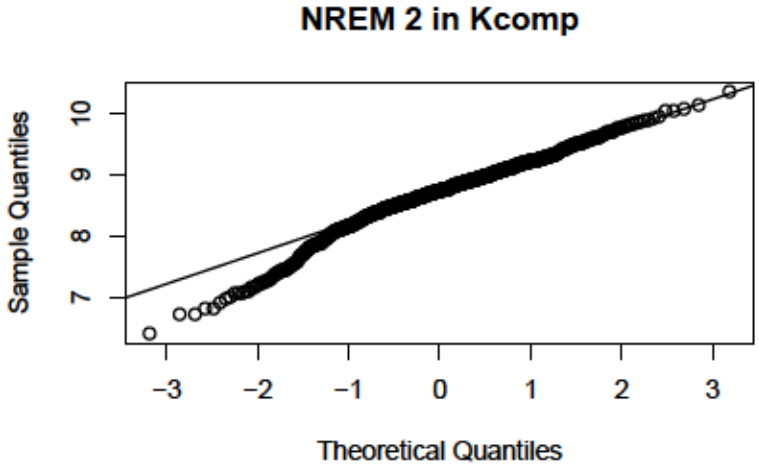

NREM 2 in Beta2
Density
0.0
0.2
0.4
2
4
6
8
10
12

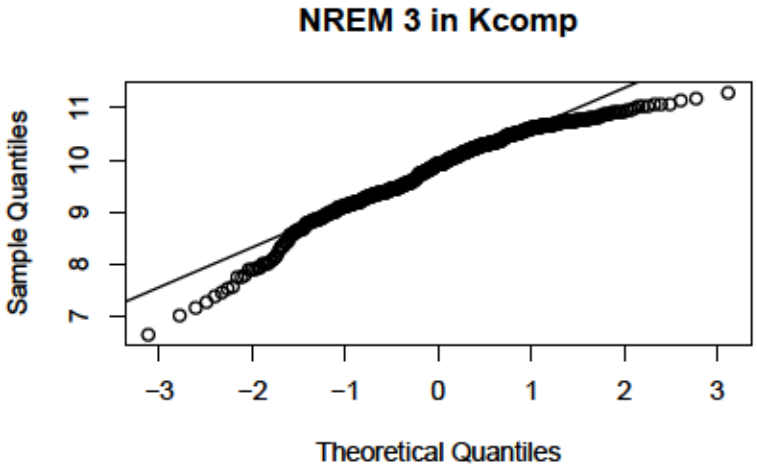

NREM 3 in Beta2
Density
0.0
0.2
0.4
2
4
6
8
10
12

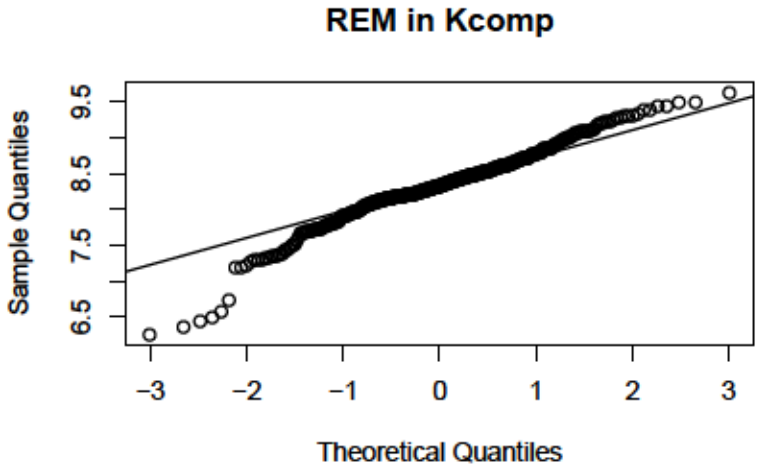

REM in Beta2
Density
0.0
0.2
0.4
2
4
6
8
10
12

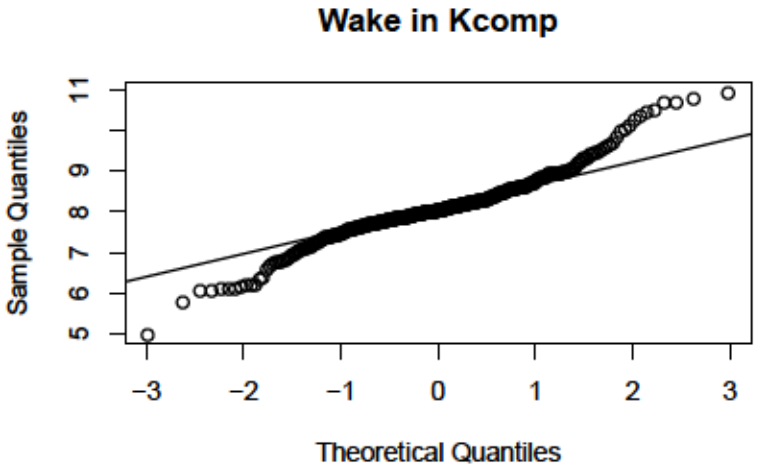

Wake in Beta2
Density
0.0
0.2
0.4
2
4
6
8
10
12

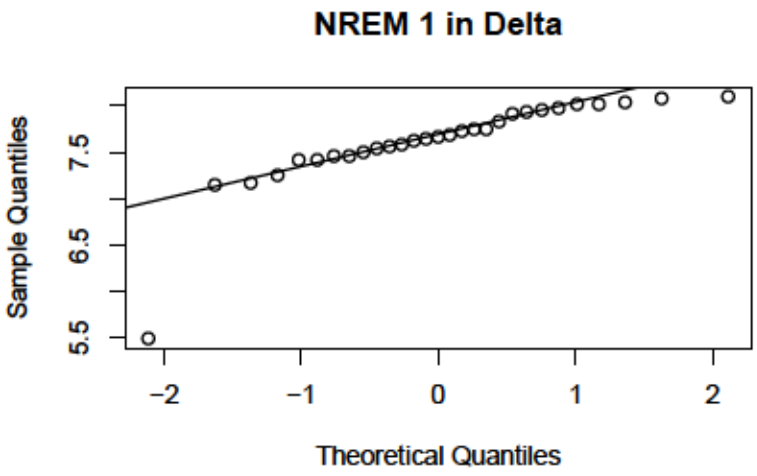

NREM 1 in Kcomp
Sample Quantiles
Theoretical Quantiles

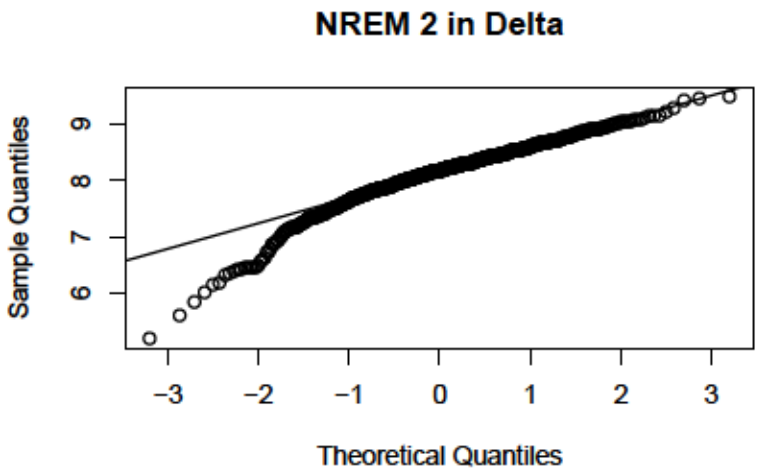

NREM 2 in Kcomp
Sample Quantiles
Theoretical Quantiles

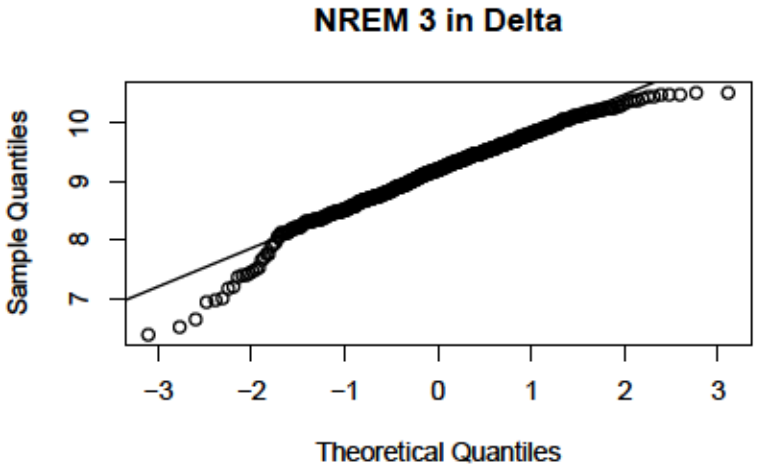

NREM 3 in Kcomp
Sample Quantiles
Theoretical Quantiles

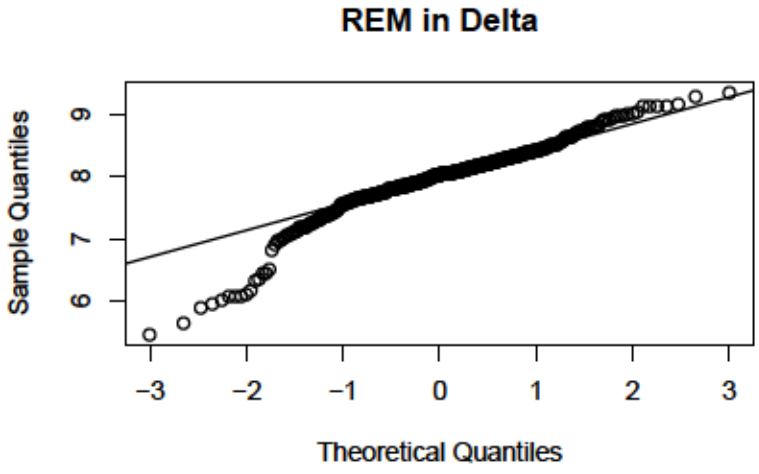

REM in Kcomp
Sample Quantiles
Theoretical Quantiles

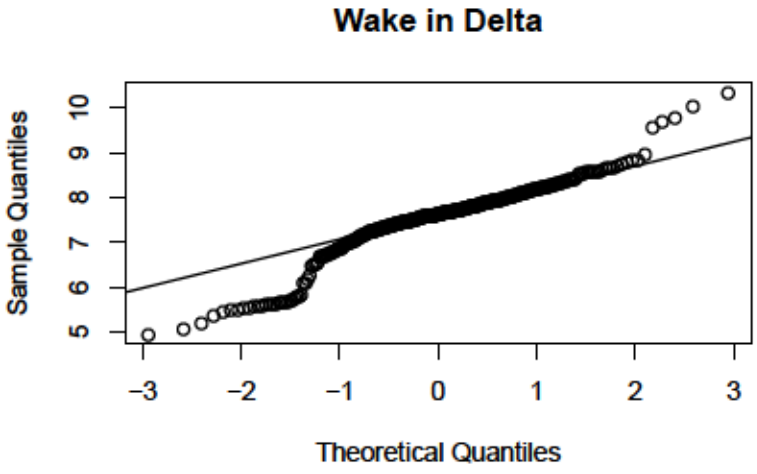

Wake in Kcomp
Sample Quantiles
Theoretical Quantiles

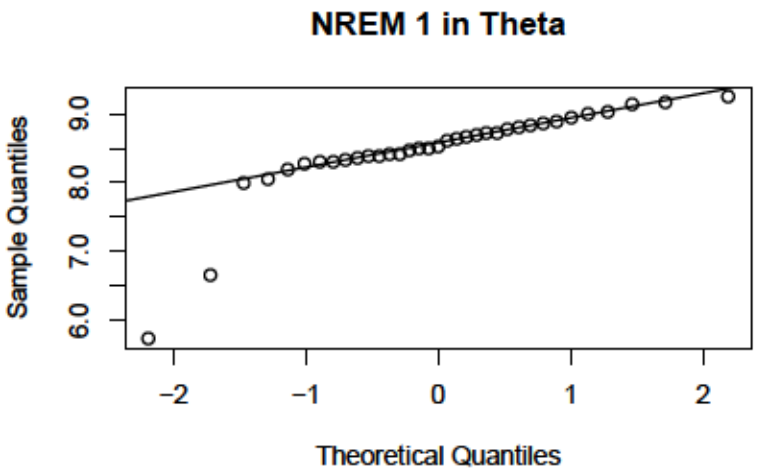

NREM 1 in Delta
Sample Quantiles
Theoretical Quantiles

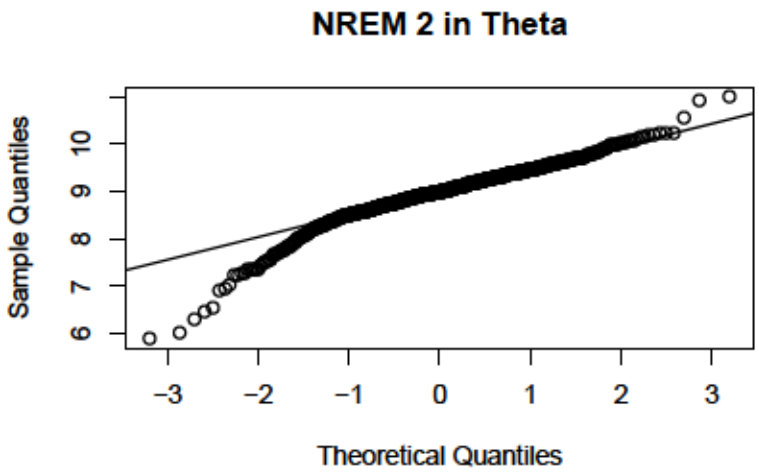

NREM 2 in Delta
Sample Quantiles
Theoretical Quantiles

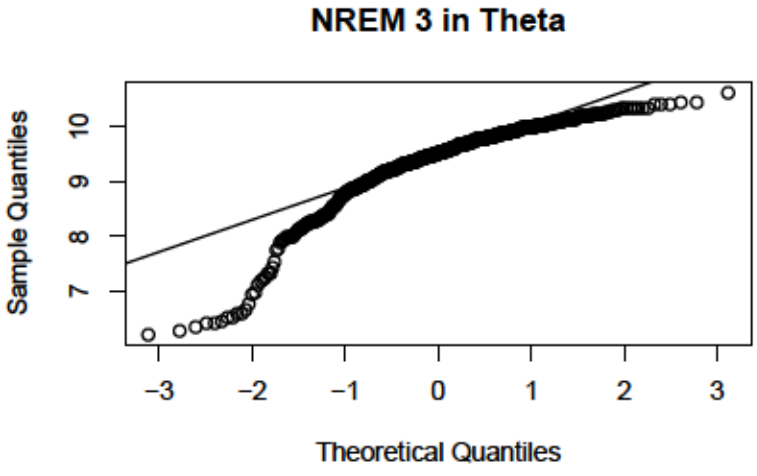

NREM 3 in Delta
Sample Quantiles
Theoretical Quantiles

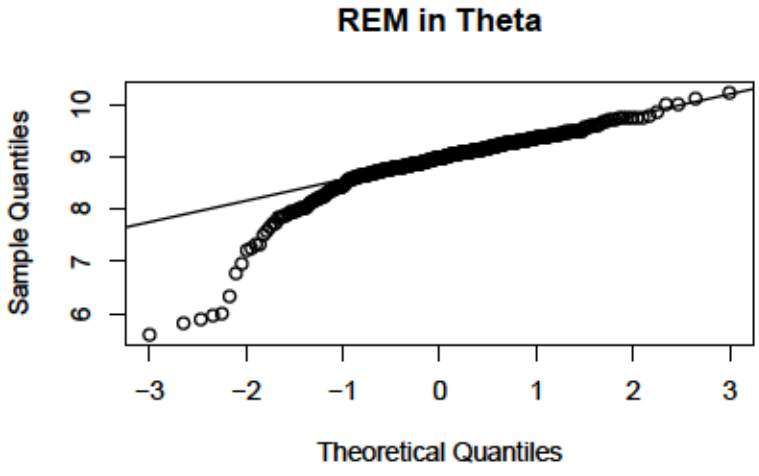

REM in Delta
Sample Quantiles
Theoretical Quantiles

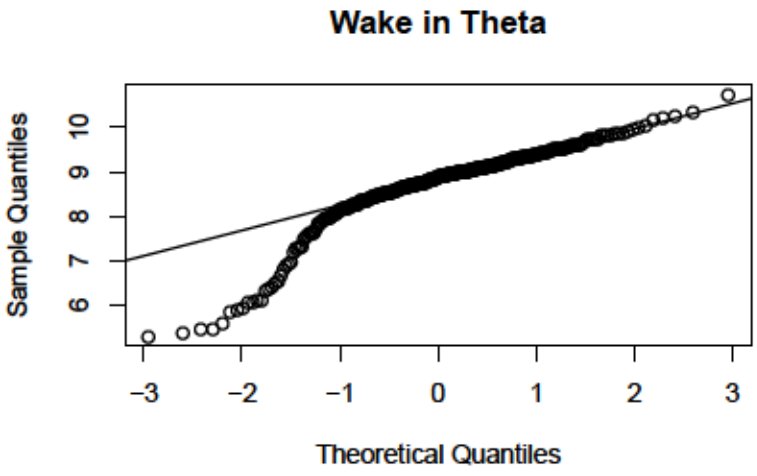

Wake in Delta
Sample Quantiles
Theoretical Quantiles

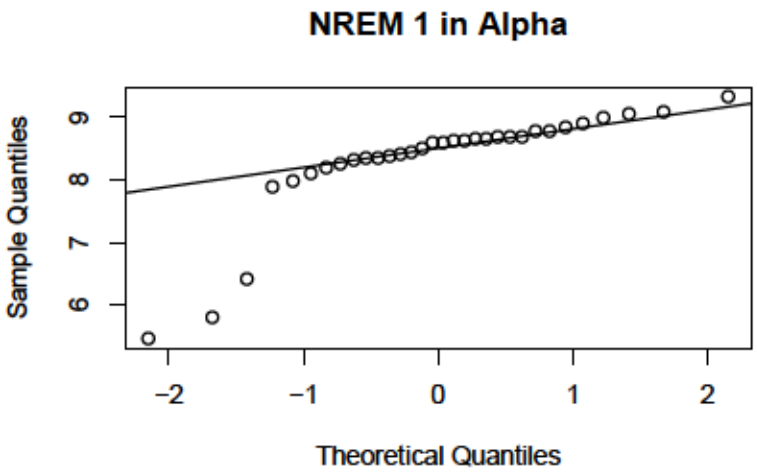

NREM 1 in Theta
Sample Quantiles
6.0
7.0
8.0
9.0
−2
−1
0
1
2
Theoretical Quantiles

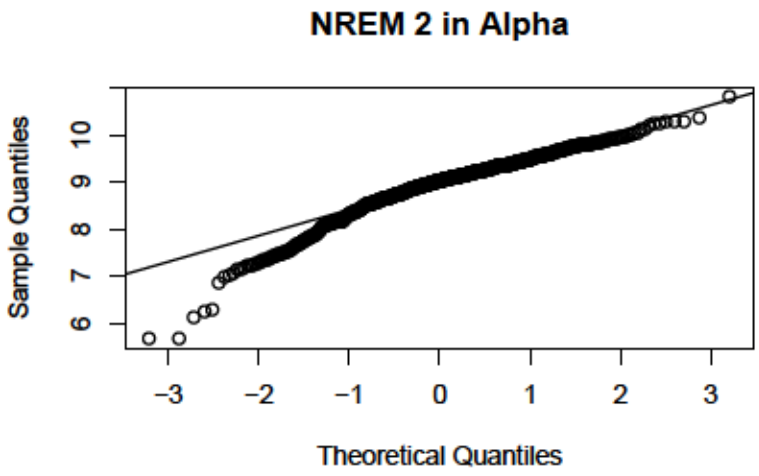

NREM 2 in Theta
Sample Quantiles
6
7
8
9
10
−3
−2
−1
0
1
2
3
Theoretical Quantiles

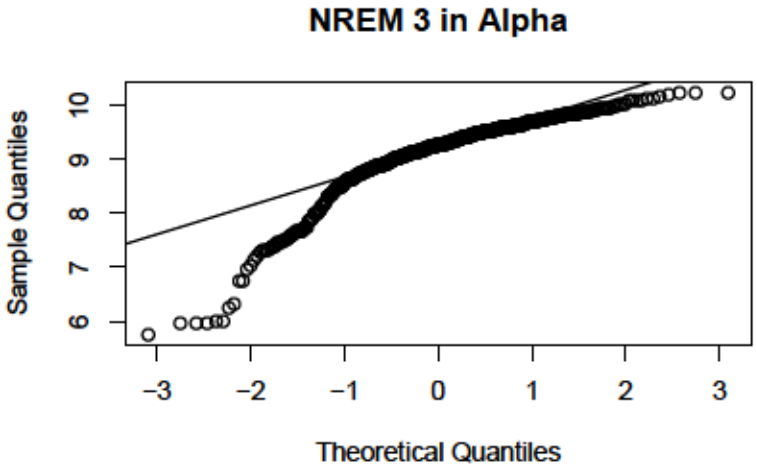

NREM 3 in Theta
Sample Quantiles
7
8
9
10
−3
−2
−1
0
1
2
3
Theoretical Quantiles

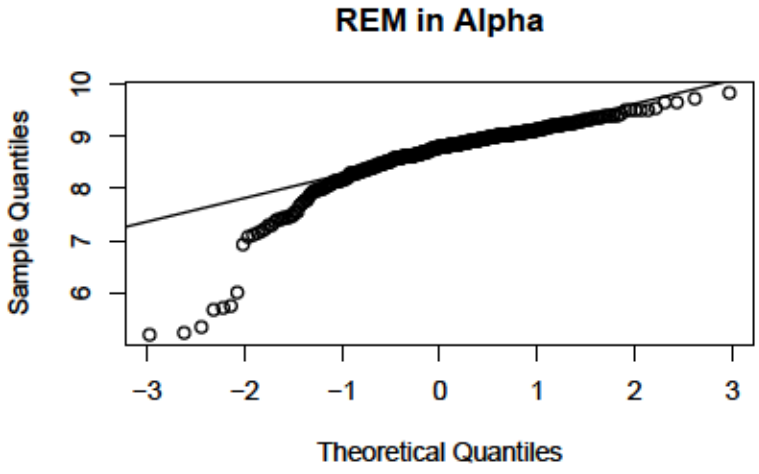

REM in Theta
Sample Quantiles
6
7
8
9
10
−3
−2
−1
0
1
2
3
Theoretical Quantiles

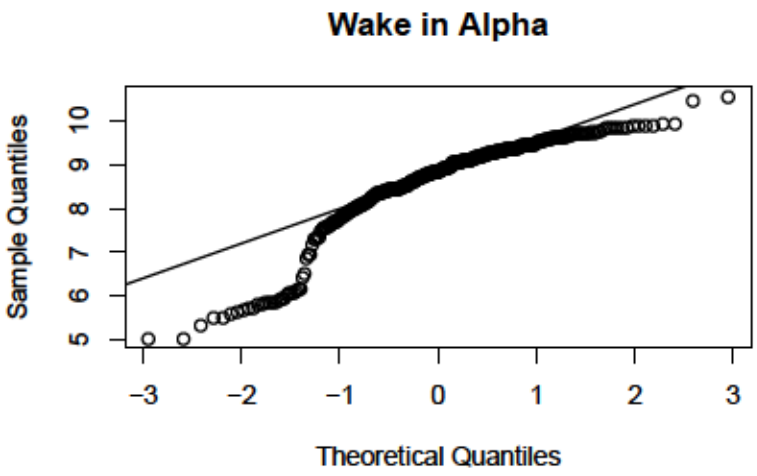

Wake in Theta
Sample Quantiles
6
7
8
9
10
−3
−2
−1
0
1
2
3
Theoretical Quantiles

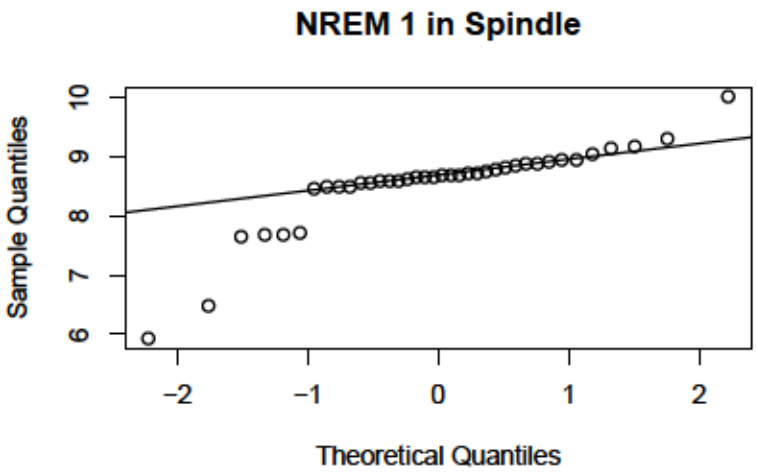

NREM 1 in Alpha
Sample Quantiles
Theoretical Quantiles

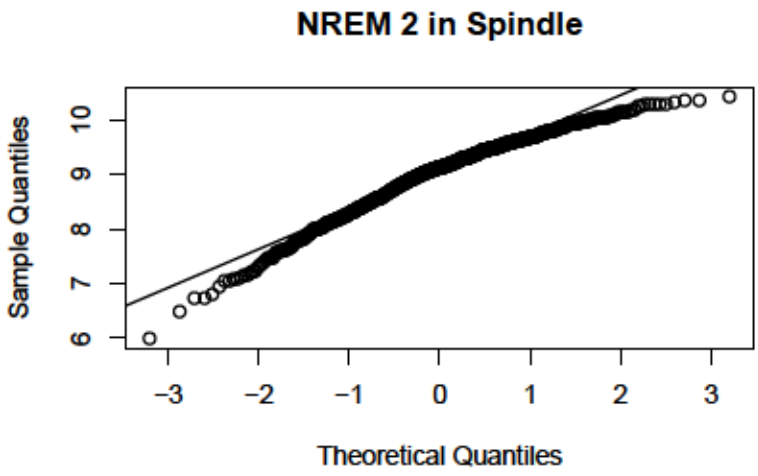

NREM 2 in Alpha
Sample Quantiles
Theoretical Quantiles

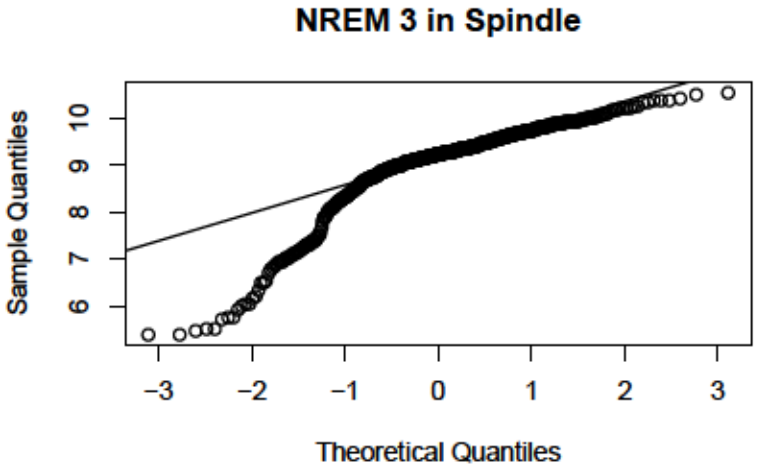

NREM 3 in Alpha
Sample Quantiles
Theoretical Quantiles

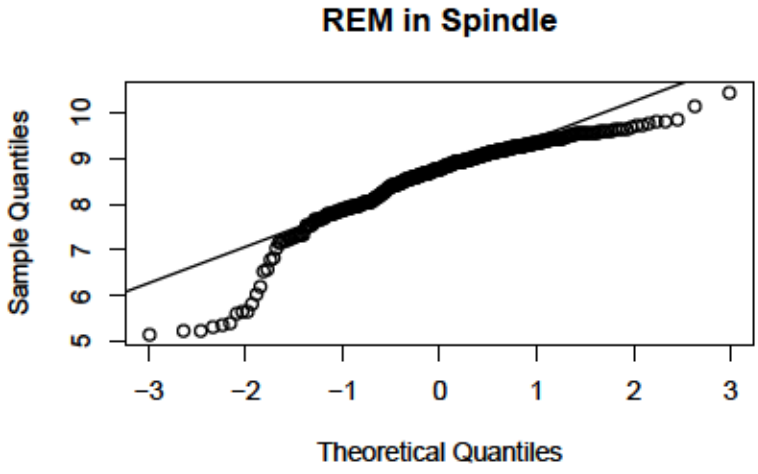

REM in Alpha
Sample Quantiles
Theoretical Quantiles

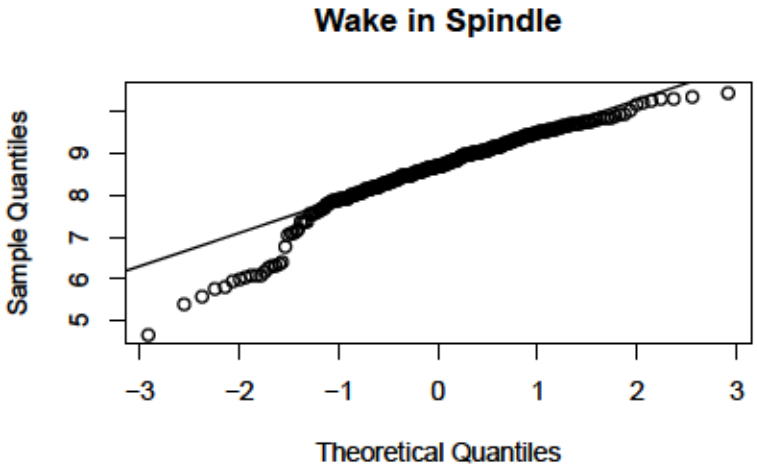

Wake in Alpha
Sample Quantiles
Theoretical Quantiles

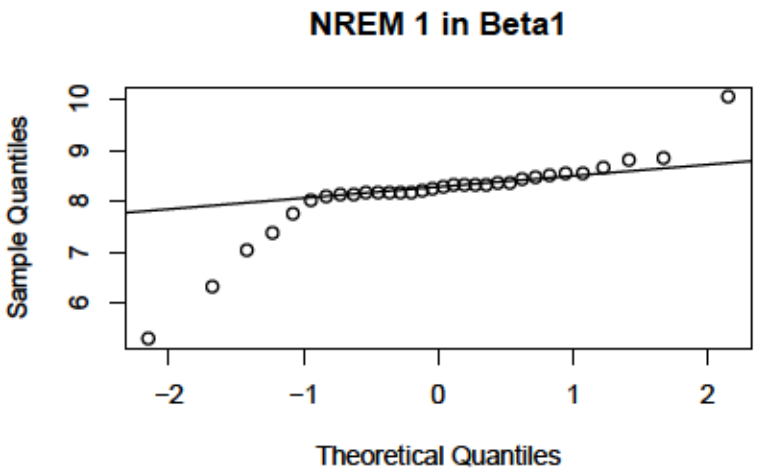

NREM 1 in Spindle
Sample Quantiles
Theoretical Quantiles

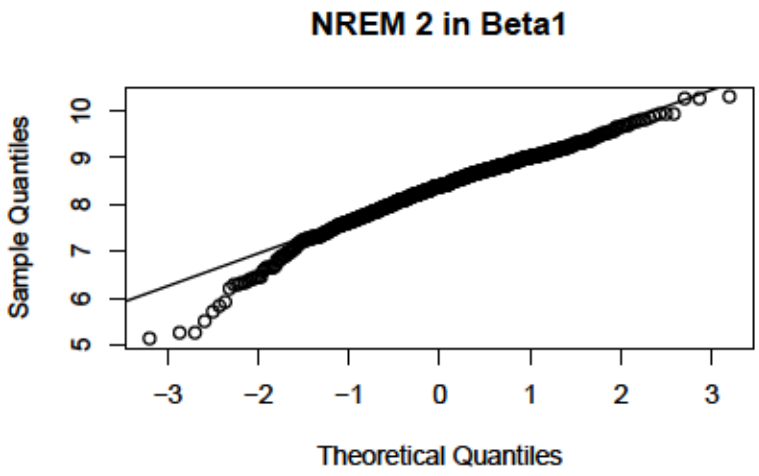

NREM 2 in Spindle
Sample Quantiles
Theoretical Quantiles

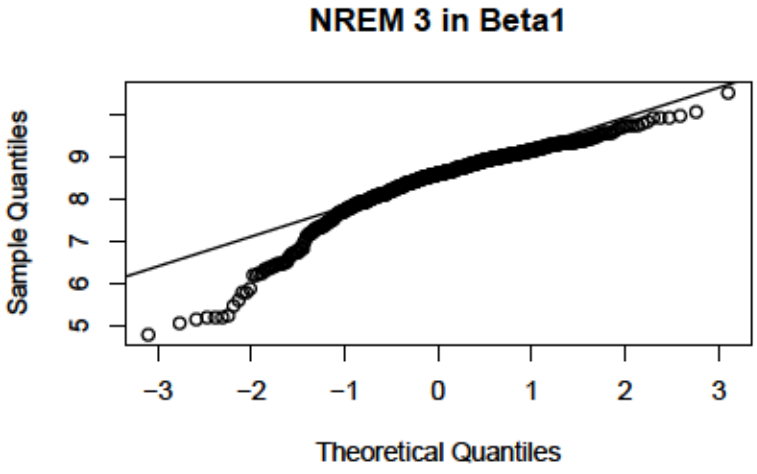

NREM 3 in Spindle
Sample Quantiles
Theoretical Quantiles

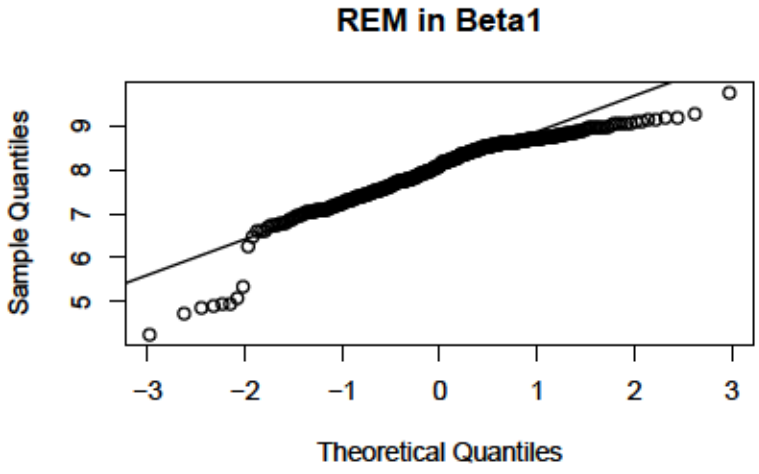

REM in Spindle
Sample Quantiles
Theoretical Quantiles

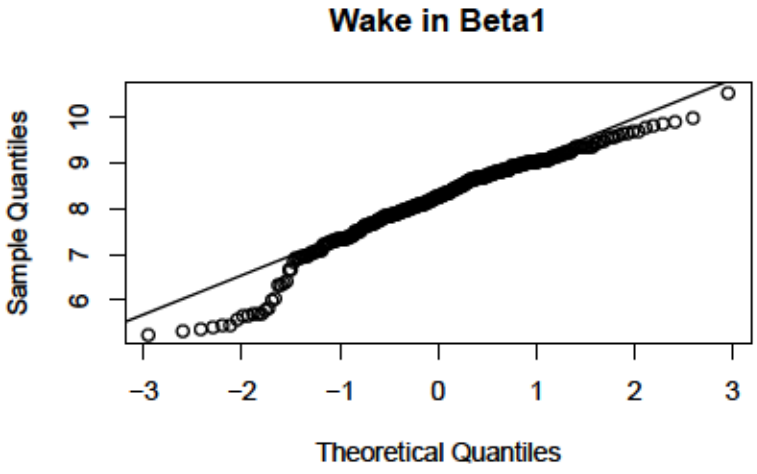

Wake in Spindle
Sample Quantiles
Theoretical Quantiles

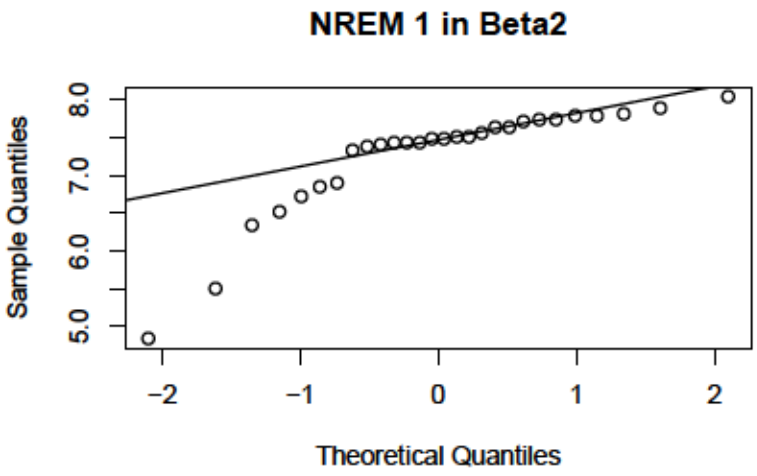

NREM 1 in Beta1
Sample Quantiles
Theoretical Quantiles

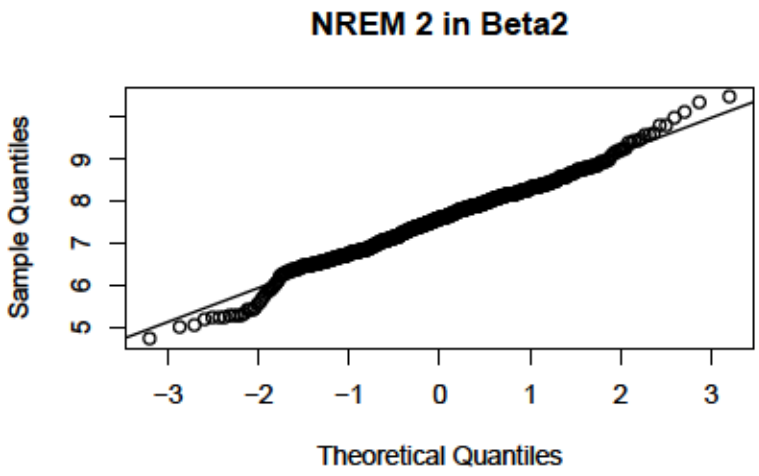

NREM 2 in Beta1
Sample Quantiles
Theoretical Quantiles

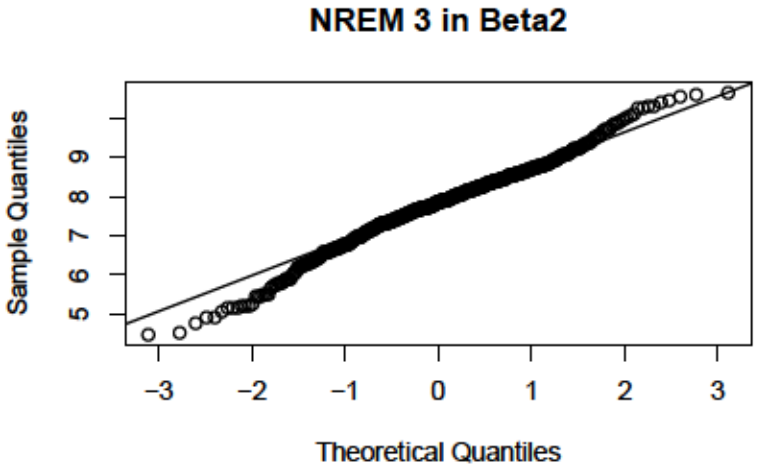

NREM 3 in Beta1
Sample Quantiles
Theoretical Quantiles

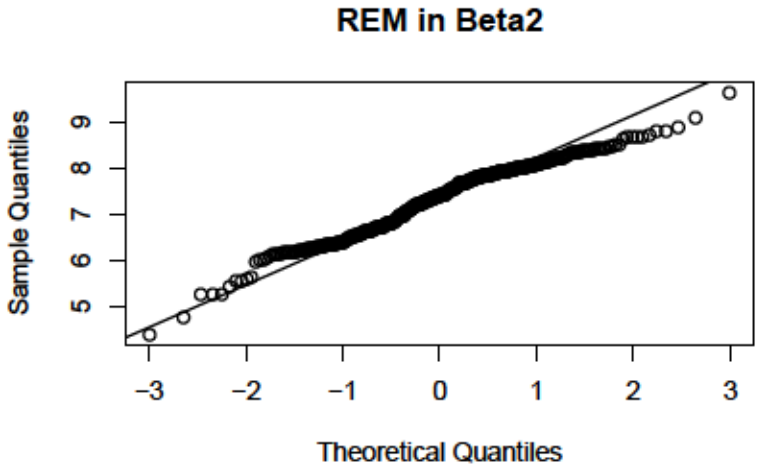

REM in Beta1
Sample Quantiles
Theoretical Quantiles

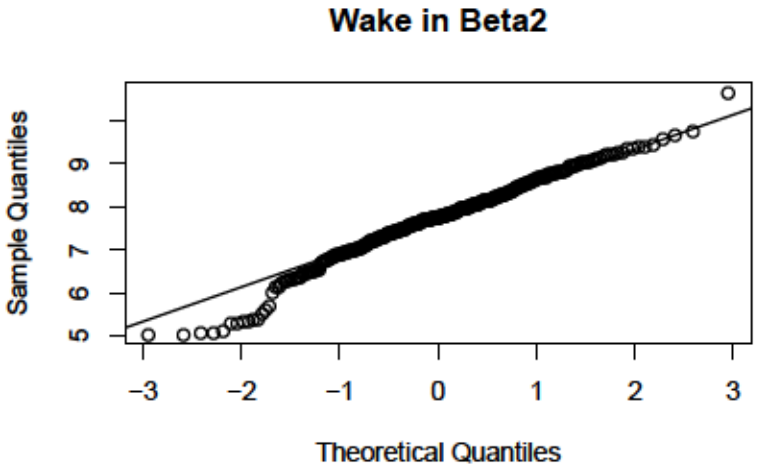

Wake in Beta1
Sample Quantiles
Theoretical Quantiles


NREM 1 in Beta2
Sample Quantiles
Theoretical Quantiles


NREM 2 in Beta2
Sample Quantiles
Theoretical Quantiles

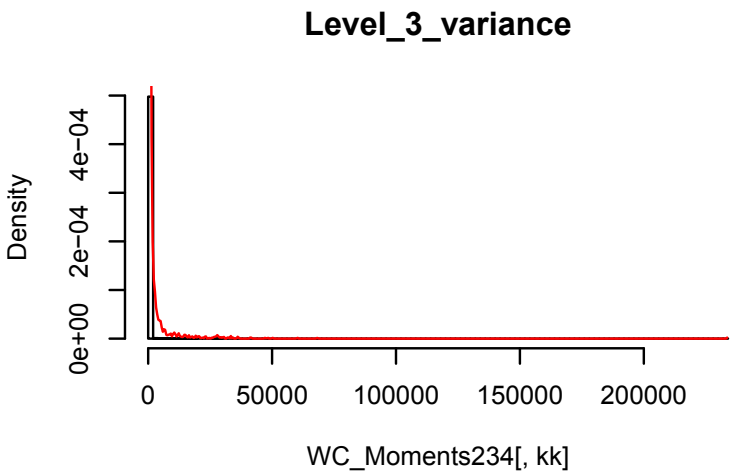

NREM 3 in Beta2
Sample Quantiles
Theoretical Quantiles

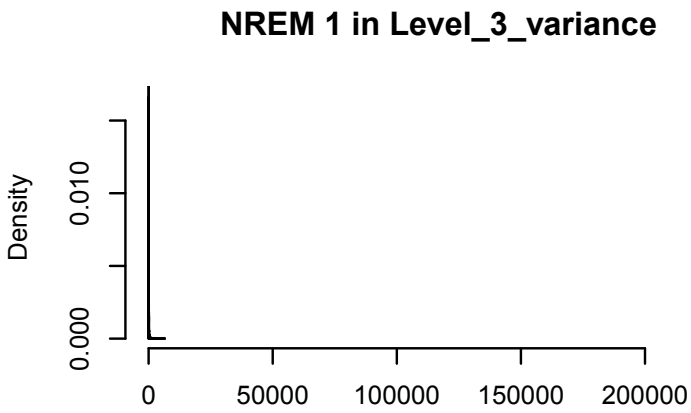

REM in Beta2
Sample Quantiles
Theoretical Quantiles

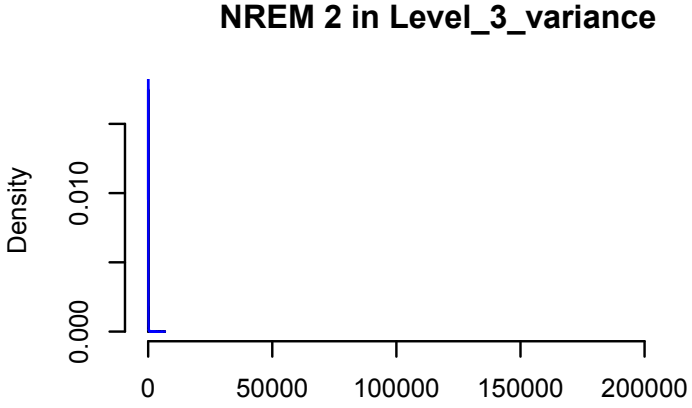

Wake in Beta2
Sample Quantiles
Theoretical Quantiles

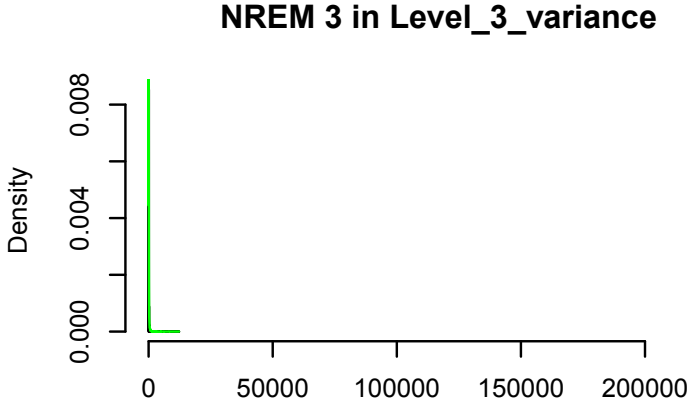
Appendix A.2

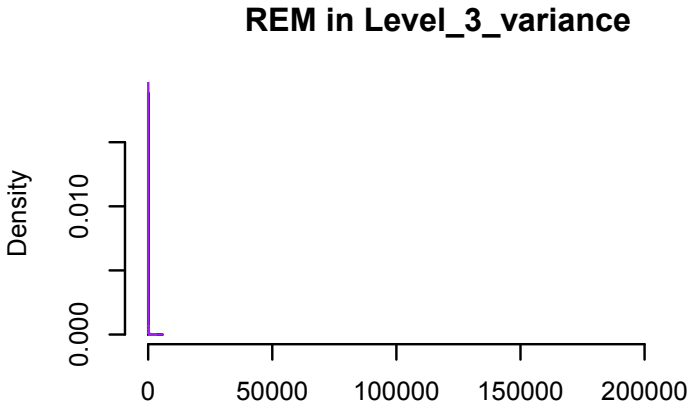
This appendix contains the density plots of all original DWC features, the QQ plots of the transformed DWC variance and kurtosis features, and the original skewness features. Lastly, Density plots of the transformed DWC variance and kurtosis features.

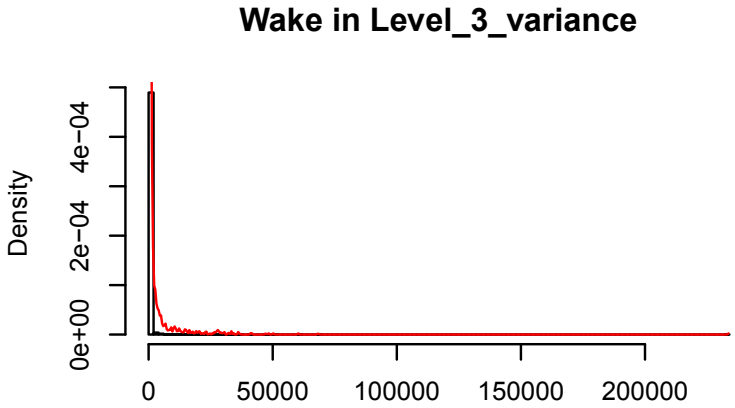

Level_3_variance
Density
0e+00
2e-04
4e-04
0
50000
100000
150000
200000
WC_Moments234[, kk]

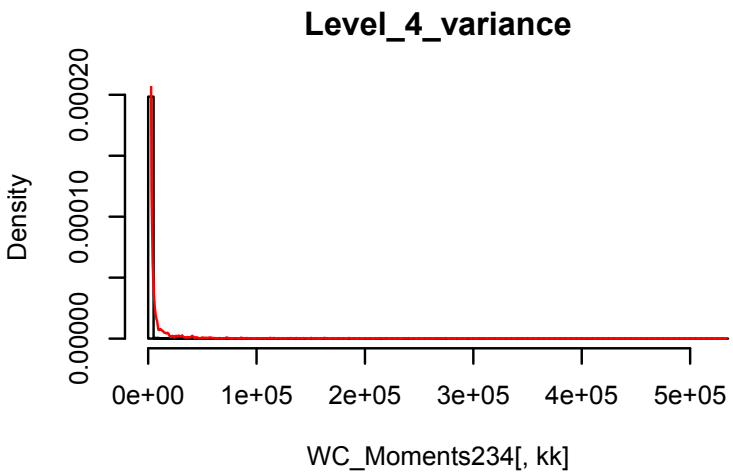

NREM 1 in Level_3_variance
Density
0.000
0.010
0
50000
100000
150000
200000

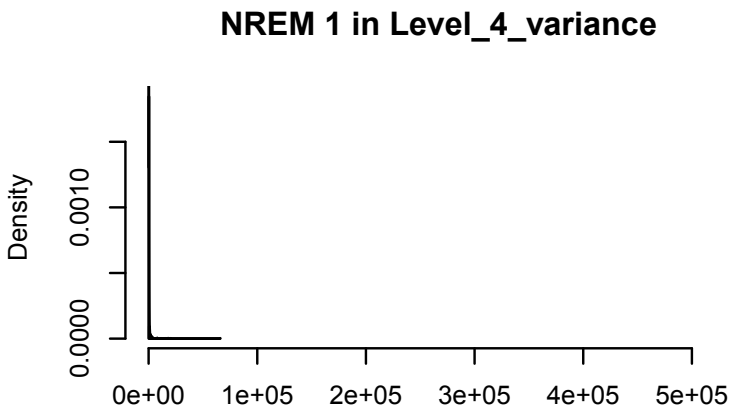

NREM 2 in Level_3_variance
Density
0.000
0.010
0
50000
100000
150000
200000

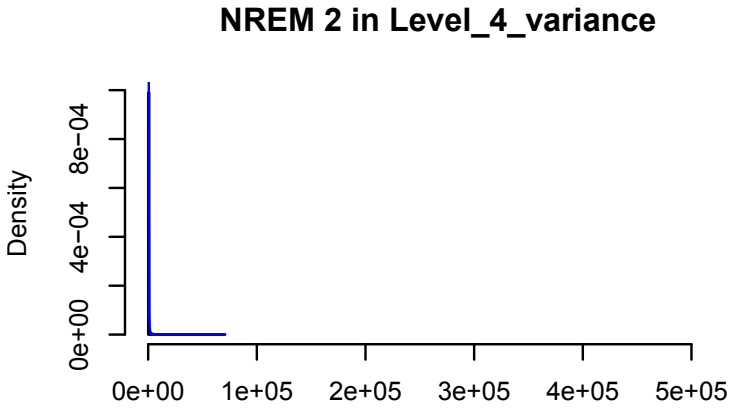

NREM 3 in Level_3_variance
Density
0.000
0.004
0.008
0
50000
100000
150000
200000

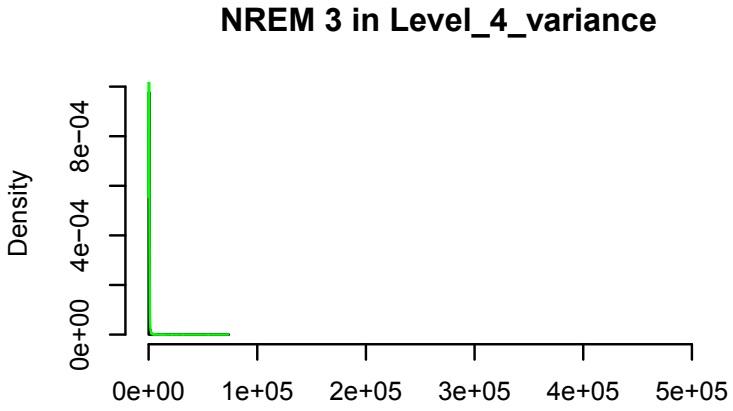

REM in Level_3_variance
Density
0.000
0.010
0
50000
100000
150000
200000

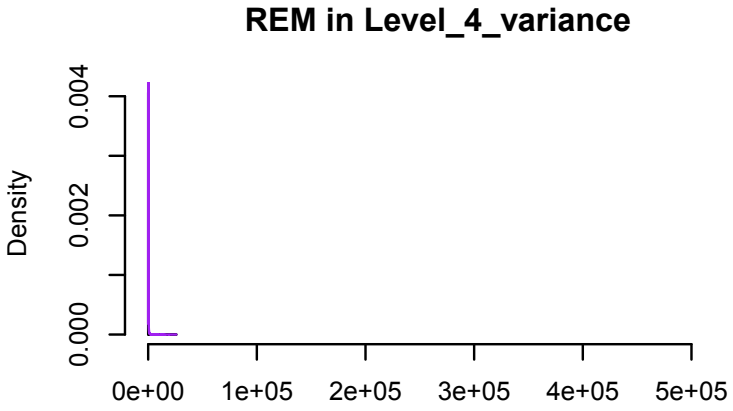

Wake in Level_3_variance
Density
0e+00
2e-04
4e-04
0
50000
100000
150000
200000

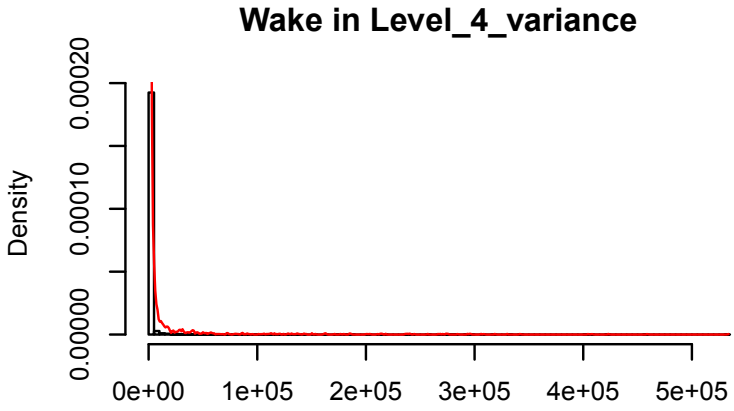

Level_4_variance
Density
0.00000
0.00010
0.00020
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05
WC_Moments234[, kk]

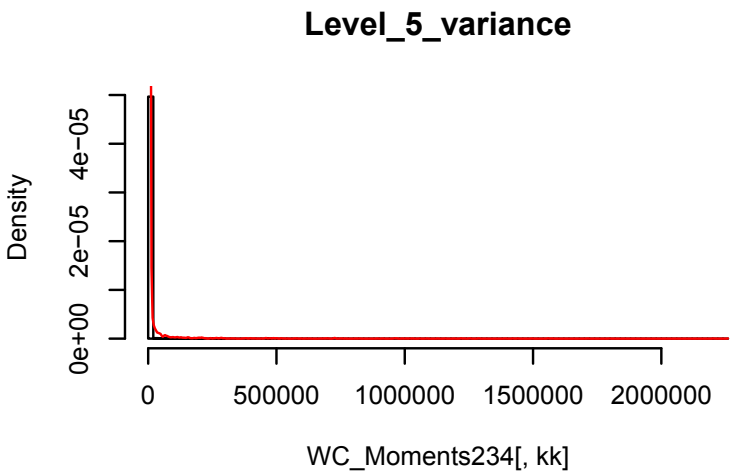

NREM 1 in Level_4_variance
Density
0.0000
0.0010
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05

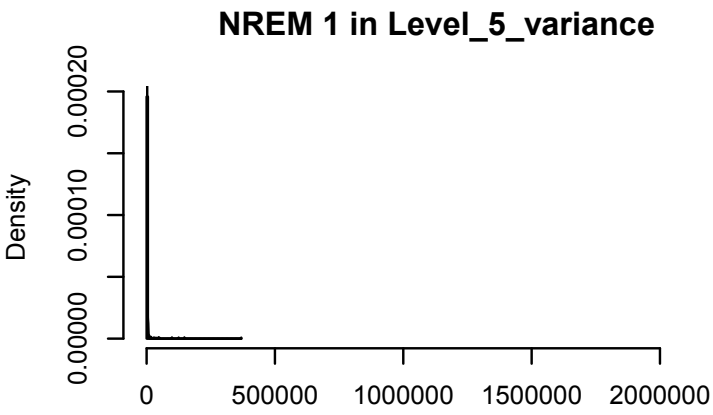

NREM 2 in Level_4_variance
Density
0e+00
4e−04
8e−04
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05

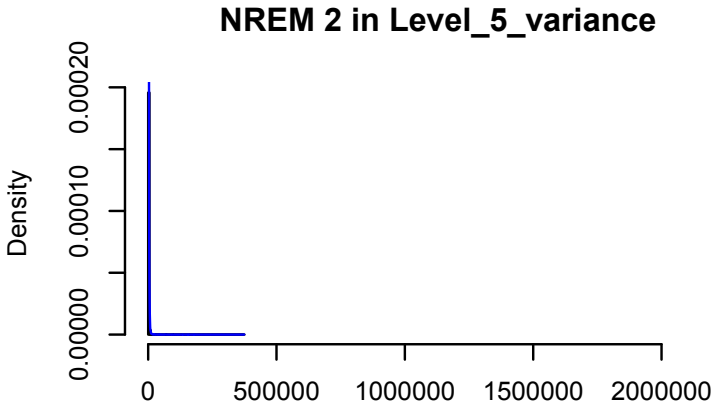

NREM 3 in Level_4_variance
Density
0e+00
4e−04
8e−04
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05

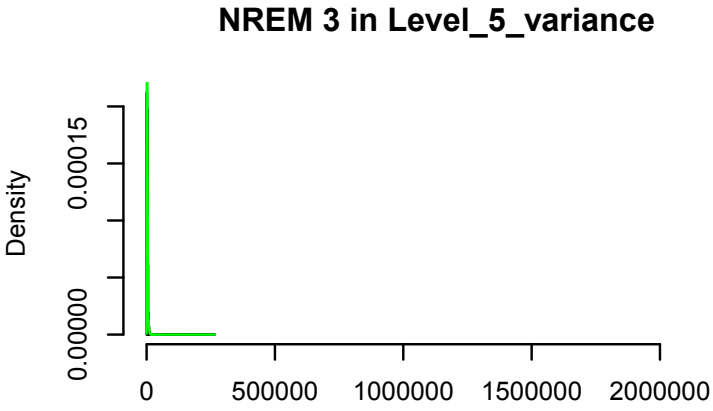

REM in Level_4_variance
Density
0.000
0.002
0.004
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05

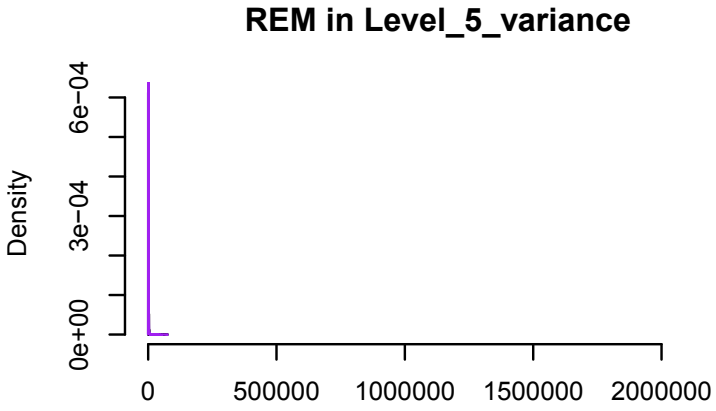

Wake in Level_4_variance
Density
0.00000
0.00010
0.00020
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05

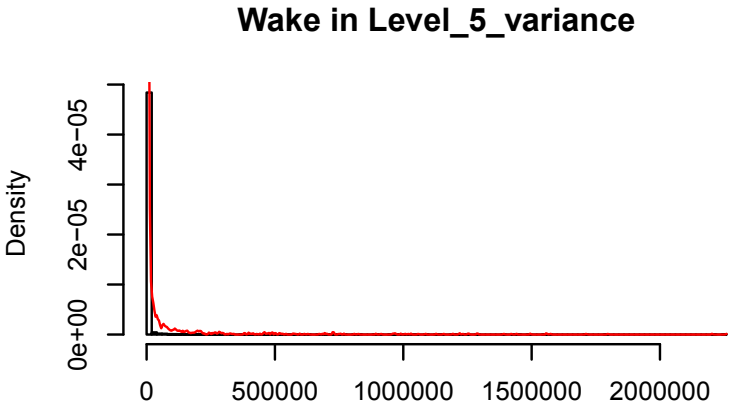

Level_5_variance
Density
0e+00
2e−05
4e−05
0
500000
1000000
1500000
2000000
WC_Moments234[, kk]

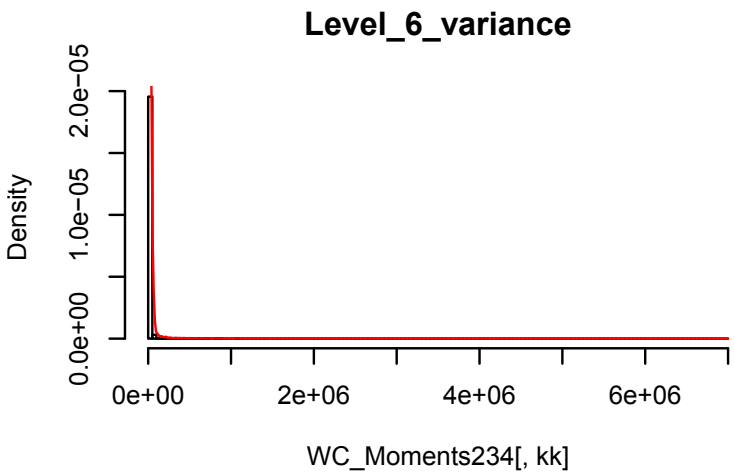

NREM 1 in Level_5_variance
Density
0.00000
0.00010
0.00020
0
500000
1000000
1500000
2000000

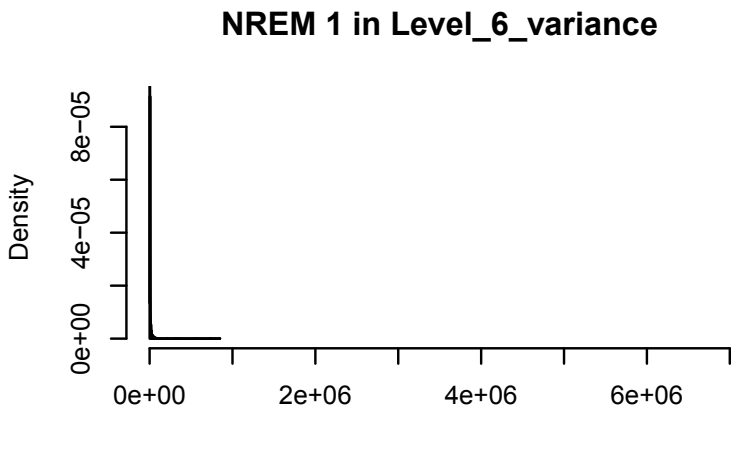

NREM 2 in Level_5_variance
Density
0.00000
0.00010
0.00020
0
500000
1000000
1500000
2000000

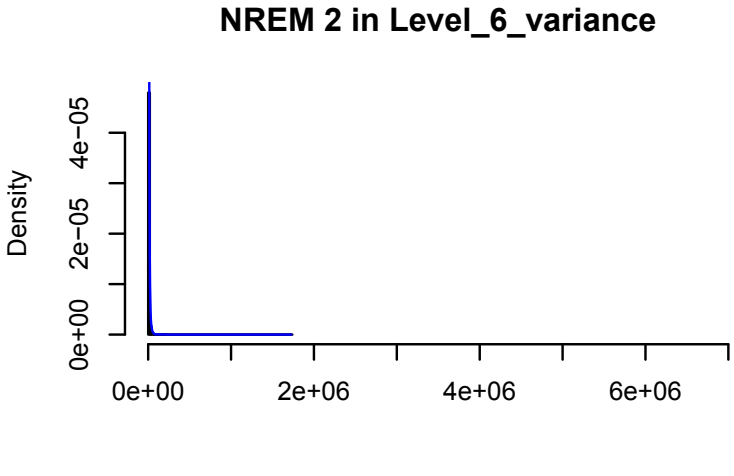

NREM 3 in Level_5_variance
Density
0.00000
0.00015
0
500000
1000000
1500000
2000000

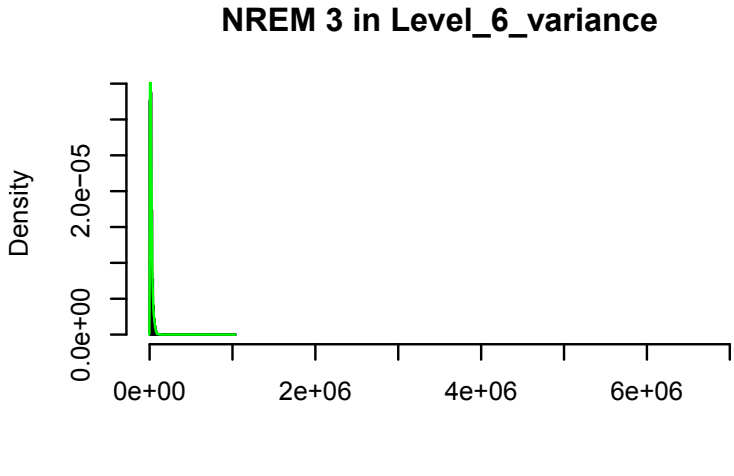

REM in Level_5_variance
Density
0e+00
3e−04
6e−04
0
500000
1000000
1500000
2000000

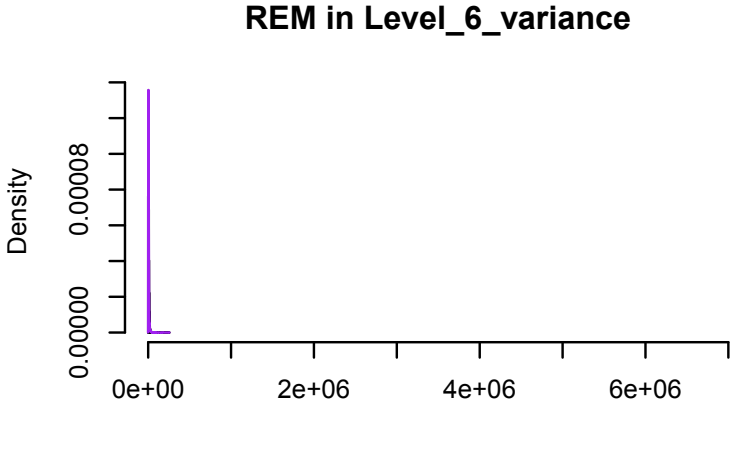

Wake in Level_5_variance
Density
0e+00
2e−05
4e−05
0
500000
1000000
1500000
2000000

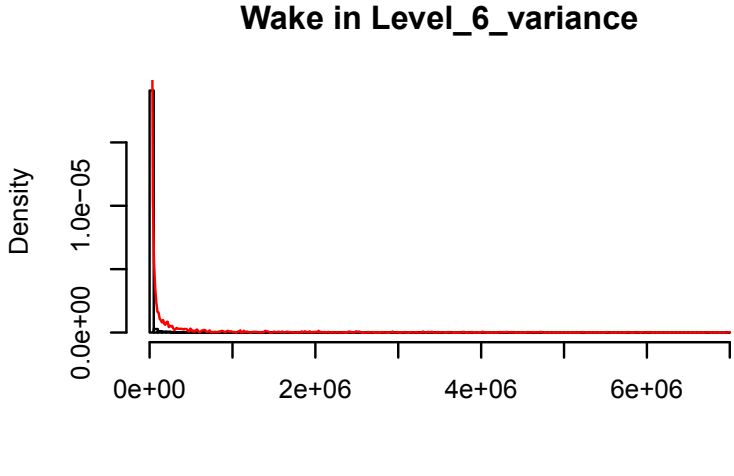

Level_6_variance
Density
0.0e+00
1.0e−05
2.0e−05
0e+00
2e+06
4e+06
6e+06
WC_Moments234[, kk]

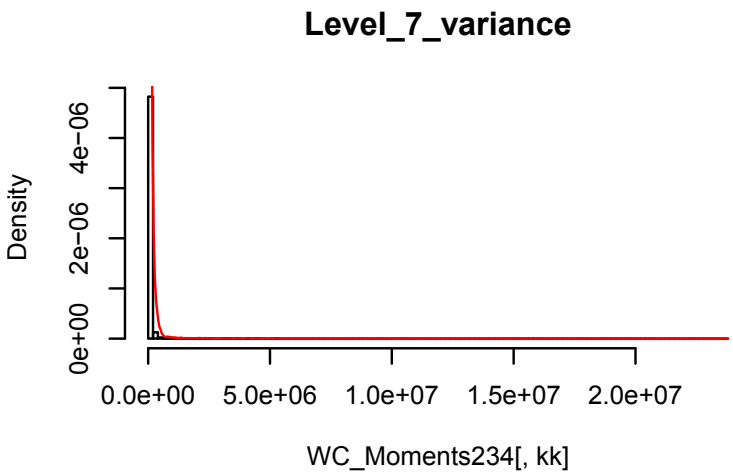

NREM 1 in Level_6_variance
Density
0e+00
4e−05
8e−05
0e+00
2e+06
4e+06
6e+06

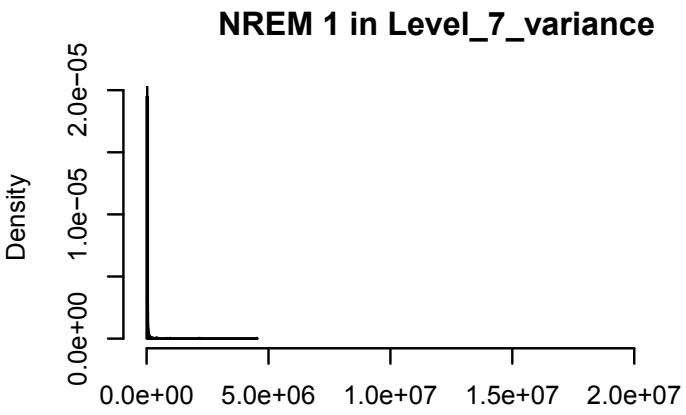

NREM 2 in Level_6_variance
Density
0e+00
2e−05
4e−05
0e+00
2e+06
4e+06
6e+06

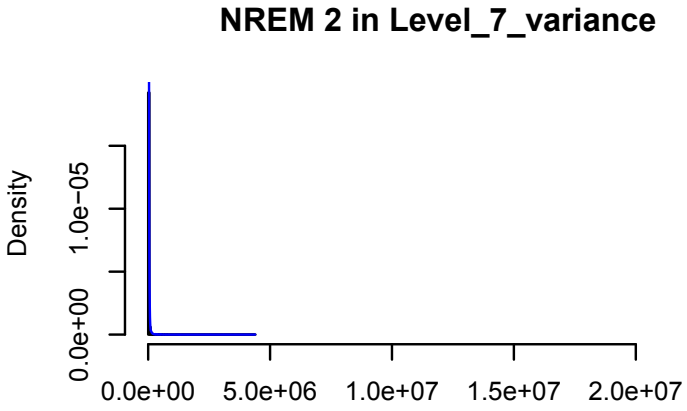

NREM 3 in Level_6_variance
Density
0.0e+00
2.0e−05
0e+00
2e+06
4e+06
6e+06

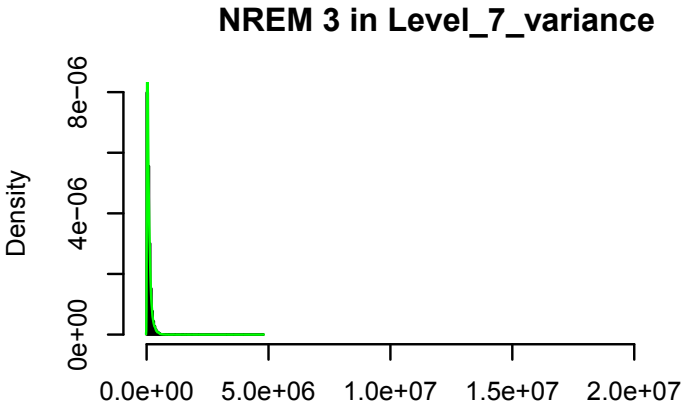

REM in Level_6_variance
Density
0.00000
0.00008
0e+00
2e+06
4e+06
6e+06

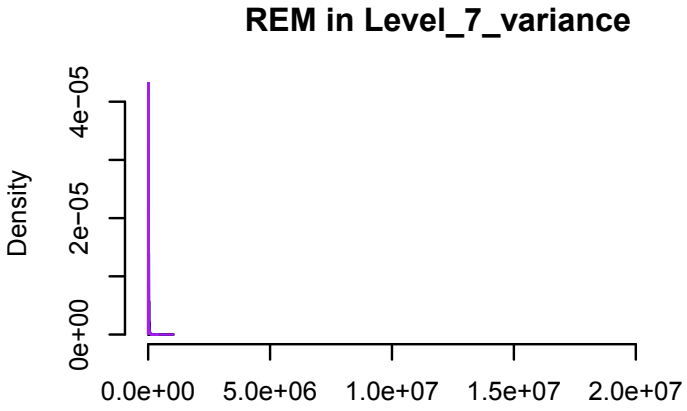

Wake in Level_6_variance
Density
0.0e+00
1.0e−05
0e+00
2e+06
4e+06
6e+06

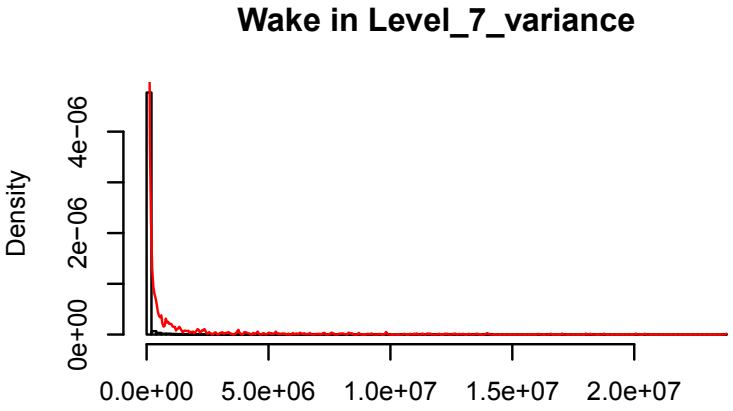

Level_7_variance
Density
0e+00
2e-06
4e-06
0.0e+00
5.0e+06
1.0e+07
1.5e+07
2.0e+07
WC_Moments234[, kk]

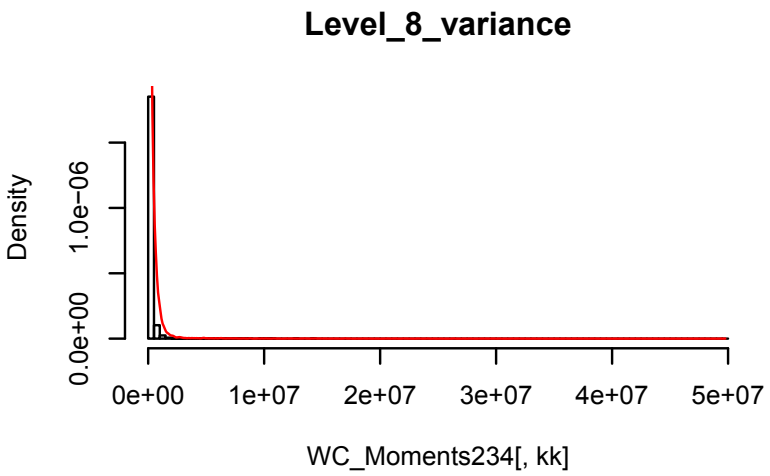

NREM 1 in Level_7_variance
Density
0.0e+00
1.0e-05
2.0e-05
0.0e+00
5.0e+06
1.0e+07
1.5e+07
2.0e+07

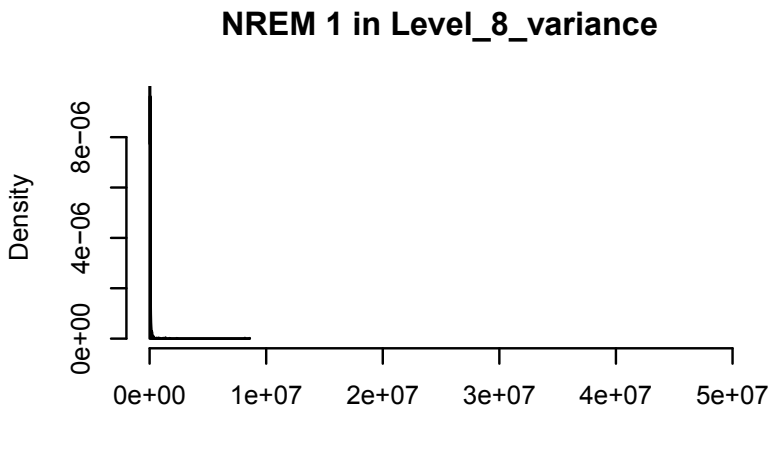

NREM 2 in Level_7_variance
Density
0.0e+00
1.0e-05
0.0e+00
5.0e+06
1.0e+07
1.5e+07
2.0e+07

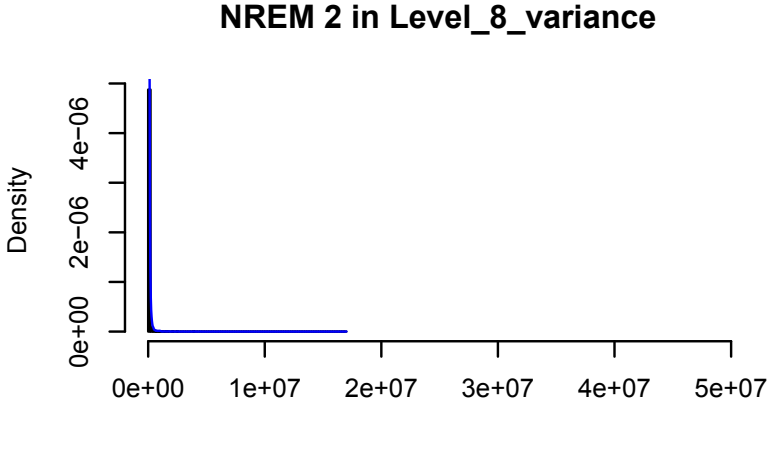

NREM 3 in Level_7_variance
Density
0e+00
4e-06
8e-06
0.0e+00
5.0e+06
1.0e+07
1.5e+07
2.0e+07

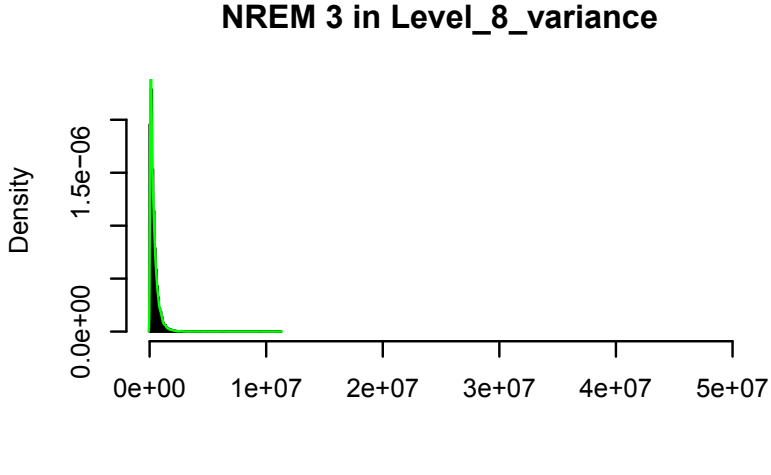

REM in Level_7_variance
Density
0e+00
2e-05
4e-05
0.0e+00
5.0e+06
1.0e+07
1.5e+07
2.0e+07

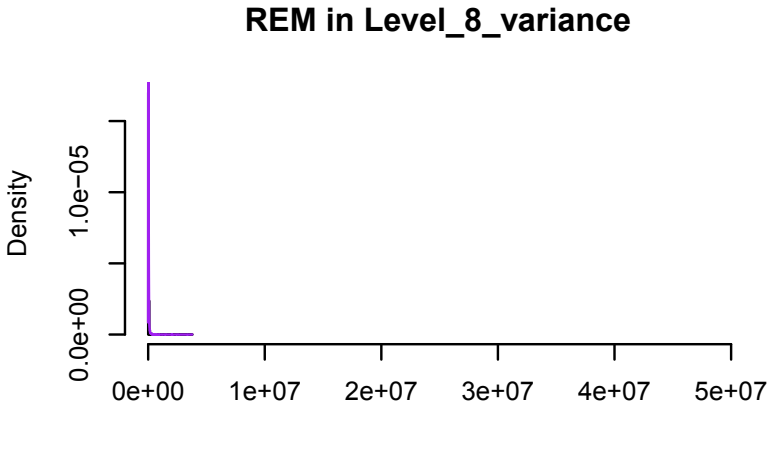

Wake in Level_7_variance
Density
0e+00
2e-06
4e-06
0.0e+00
5.0e+06
1.0e+07
1.5e+07
2.0e+07

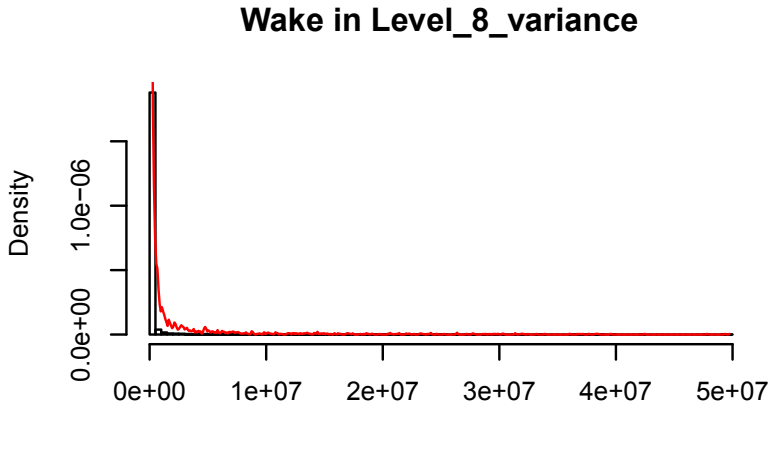

Level_8_variance
Density
0.0e+00
1.0e-06
0e+00
1e+07
2e+07
3e+07
4e+07
5e+07
WC_Moments234[, kk]

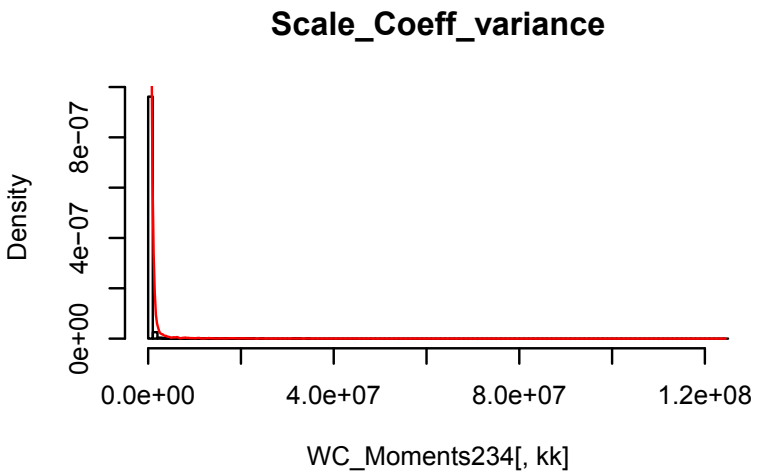

NREM 1 in Level_8_variance
Density
0e+00
4e-06
8e-06
0e+00
1e+07
2e+07
3e+07
4e+07
5e+07

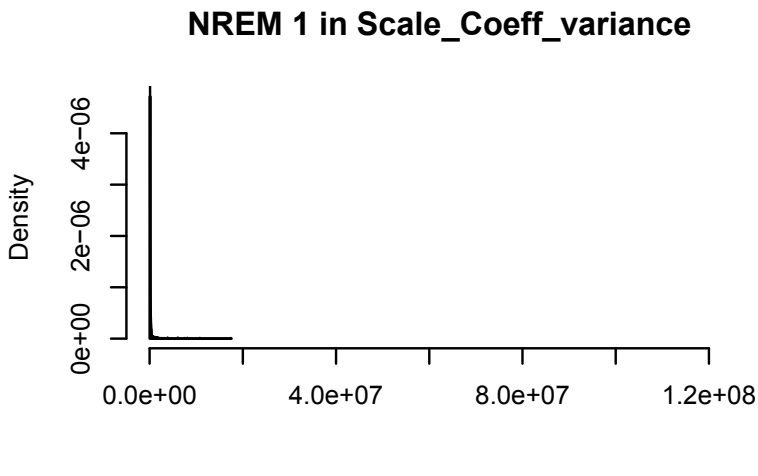

NREM 2 in Level_8_variance
Density
0e+00
2e-06
4e-06
0e+00
1e+07
2e+07
3e+07
4e+07
5e+07

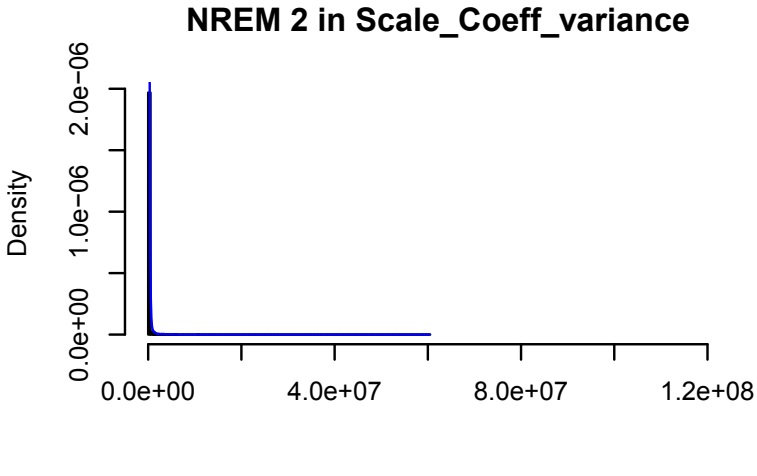

NREM 3 in Level_8_variance
Density
0.0e+00
1.5e-06
0e+00
1e+07
2e+07
3e+07
4e+07
5e+07

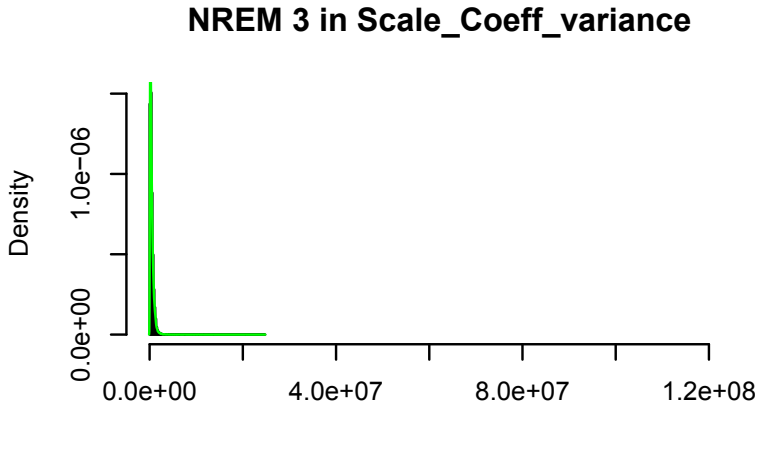

REM in Level_8_variance
Density
0.0e+00
1.0e-05
0e+00
1e+07
2e+07
3e+07
4e+07
5e+07

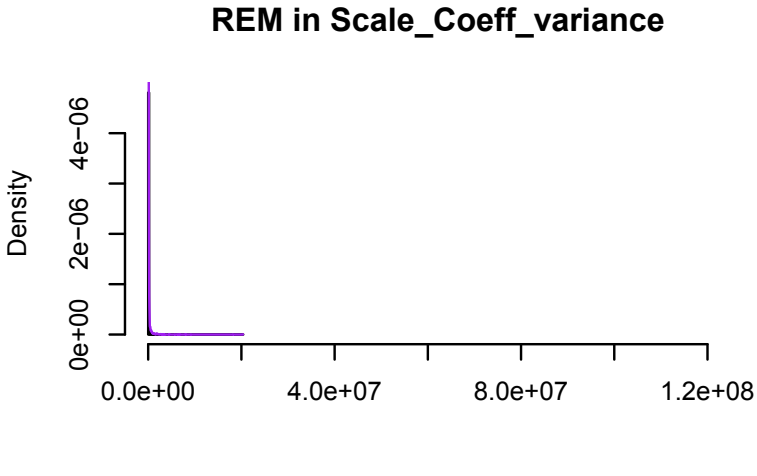

Wake in Level_8_variance
Density
0.0e+00
1.0e-06
0e+00
1e+07
2e+07
3e+07
4e+07
5e+07

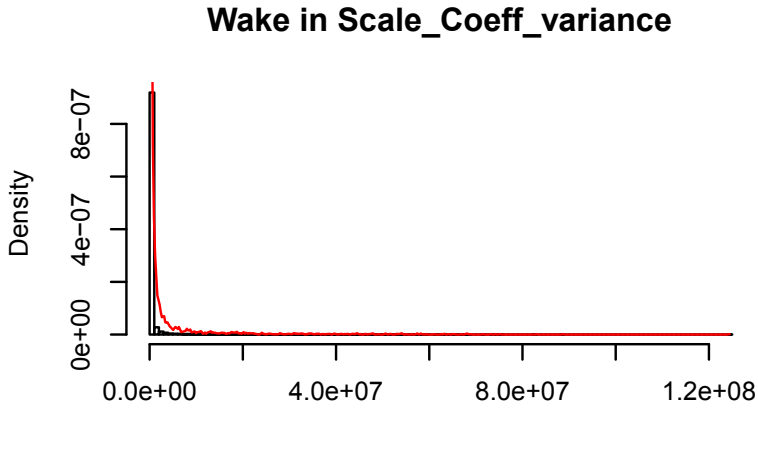

Scale_Coeff_variance
Density
0e+00
4e−07
8e−07
0.0e+00
4.0e+07
8.0e+07
1.2e+08
WC_Moments234[, kk]

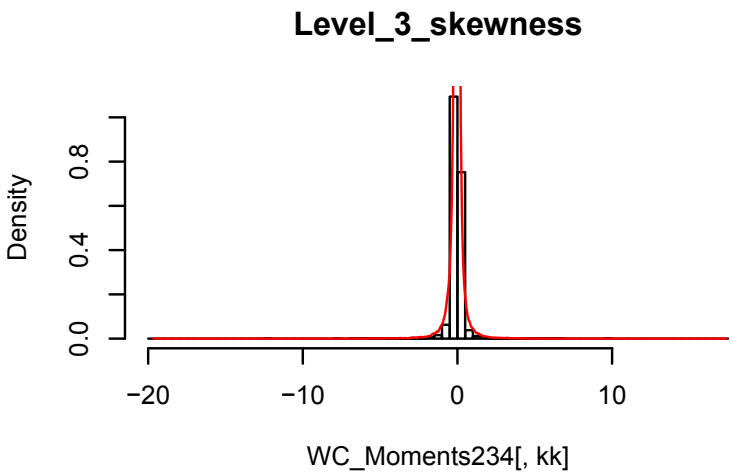

NREM 1 in Scale_Coeff_variance
Density
0e+00
2e−06
4e−06
0.0e+00
4.0e+07
8.0e+07
1.2e+08

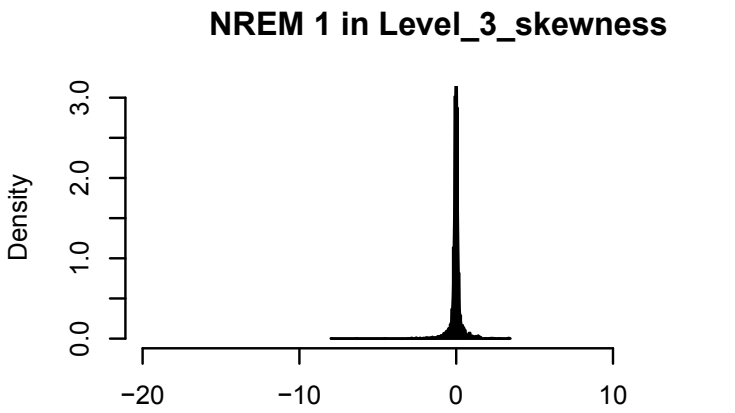

NREM 2 in Scale_Coeff_variance
Density
0.0e+00
1.0e−06
2.0e−06
0.0e+00
4.0e+07
8.0e+07
1.2e+08

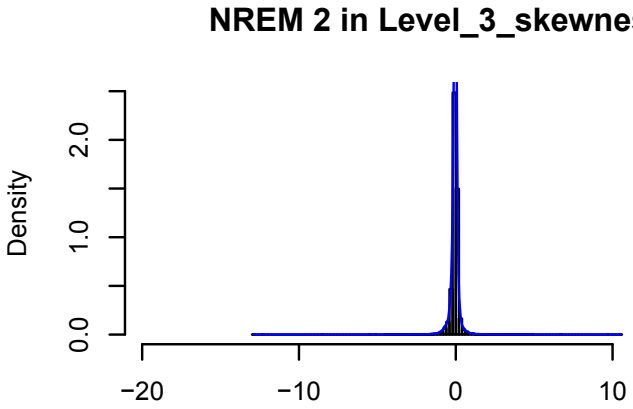

NREM 3 in Scale_Coeff_variance
Density
0.0e+00
1.0e−06
0.0e+00
4.0e+07
8.0e+07
1.2e+08

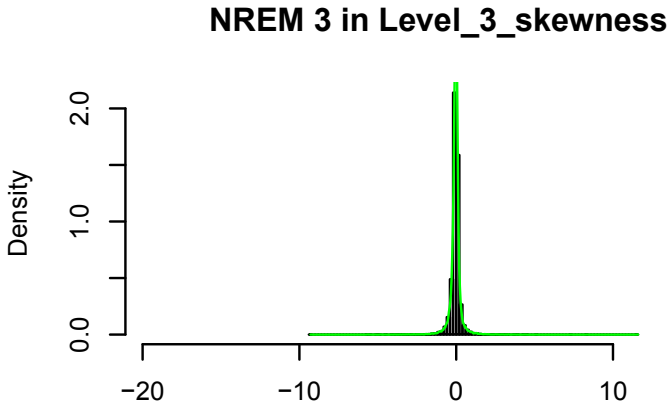

REM in Scale_Coeff_variance
Density
0e+00
2e−06
4e−06
0.0e+00
4.0e+07
8.0e+07
1.2e+08

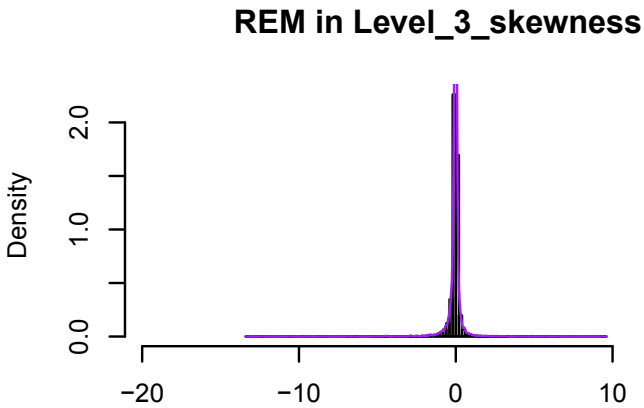

Wake in Scale_Coeff_variance
Density
0e+00
4e−07
8e−07
0.0e+00
4.0e+07
8.0e+07
1.2e+08

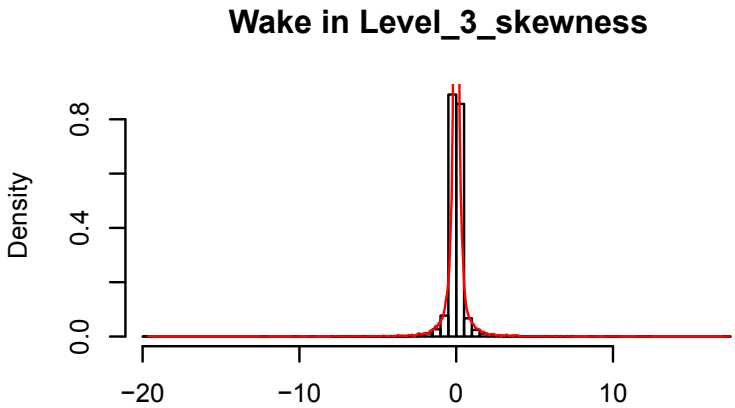

Level_3_skewness
Density
0.0
0.4
0.8
−20
−10
0
10
WC_Moments234[, kk]

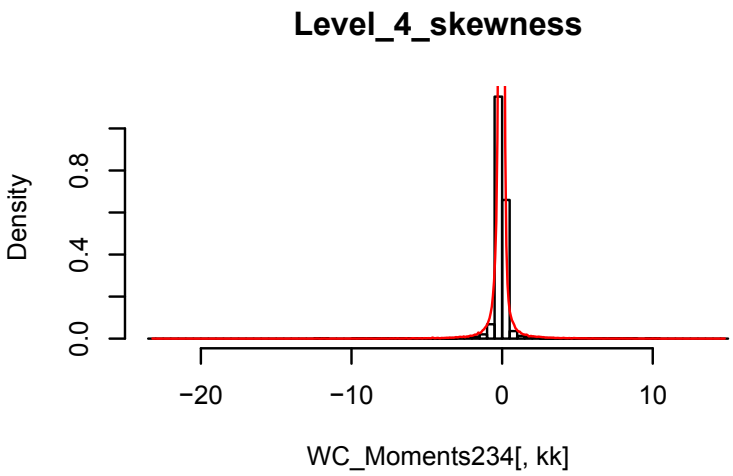

NREM 1 in Level_3_skewness
Density
0.0
1.0
2.0
3.0
−20
−10
0
10

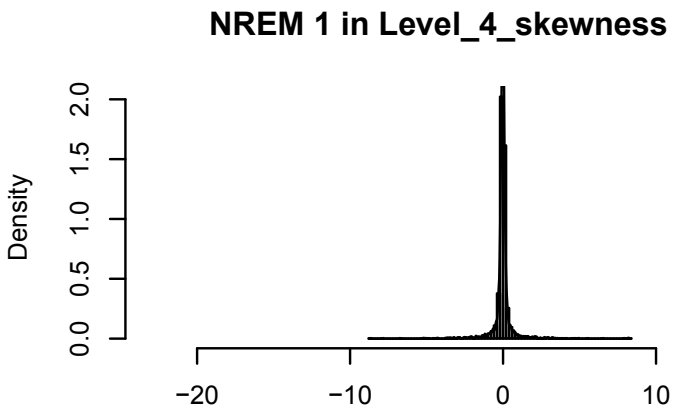

NREM 2 in Level_3_skewness
Density
0.0
1.0
2.0
−20
−10
0
10

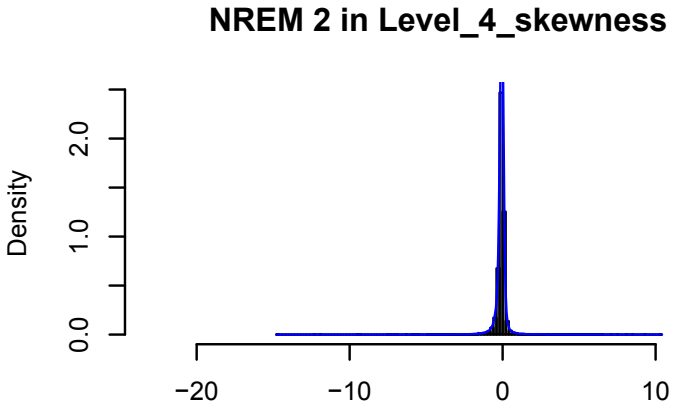

NREM 3 in Level_3_skewness
Density
0.0
1.0
2.0
−20
−10
0
10

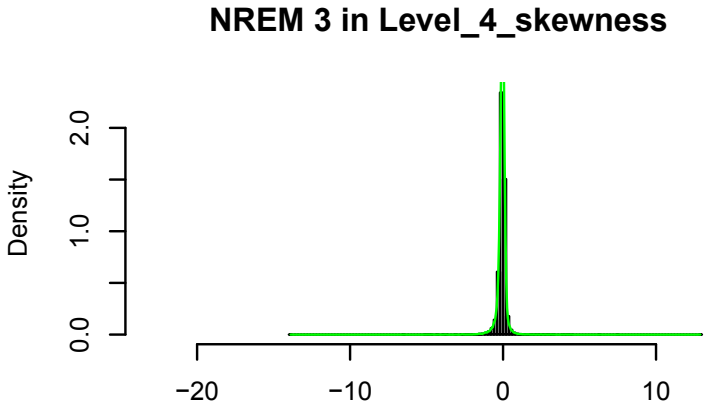

REM in Level_3_skewness
Density
0.0
1.0
2.0
−20
−10
0
10

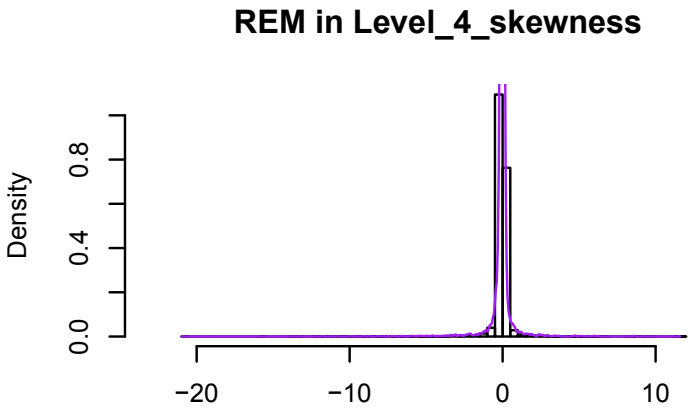

Wake in Level_3_skewness
Density
0.0
0.4
0.8
−20
−10
0
10

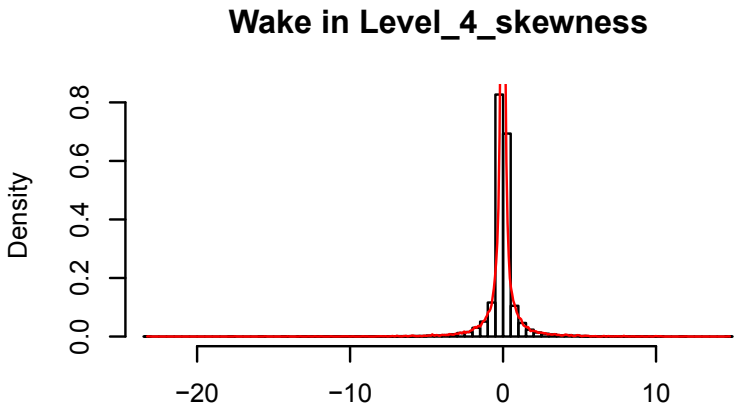

Level_4_skewness
Density
0.0
0.4
0.8
−20
−10
0
10
WC_Moments234[, kk]

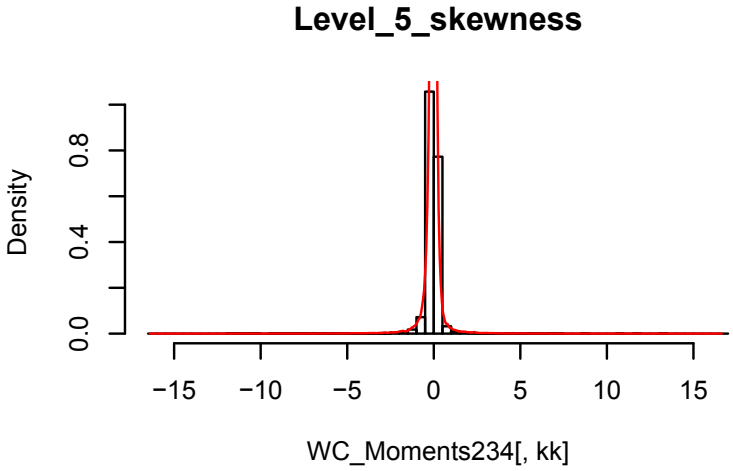

NREM 1 in Level_4_skewness
Density
0.0
0.5
1.0
1.5
2.0
−20
−10
0
10

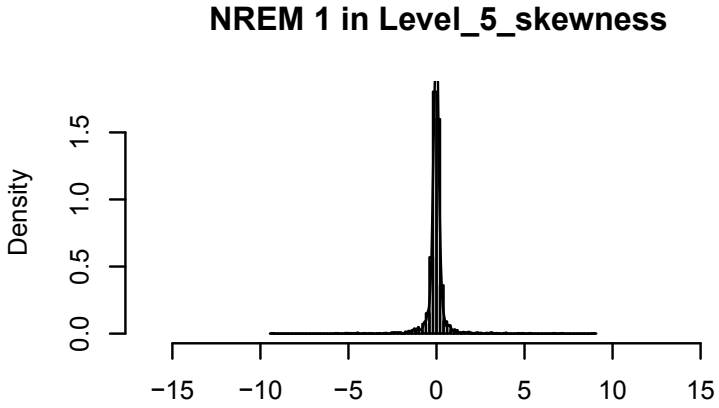

NREM 2 in Level_4_skewness
Density
0.0
1.0
2.0
−20
−10
0
10

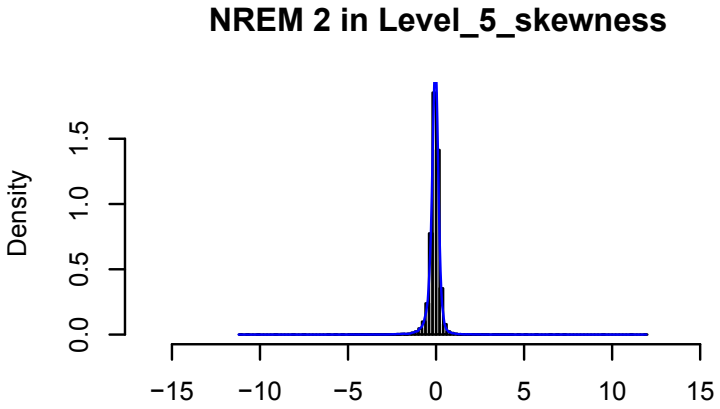

NREM 3 in Level_4_skewness
Density
0.0
1.0
2.0
−20
−10
0
10

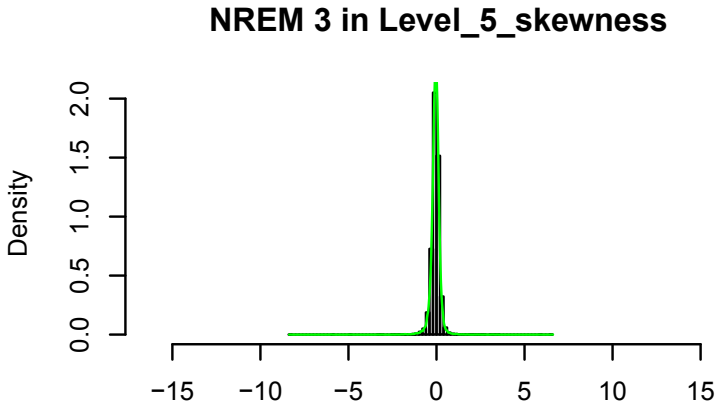

REM in Level_4_skewness
Density
0.0
0.4
0.8
−20
−10
0
10

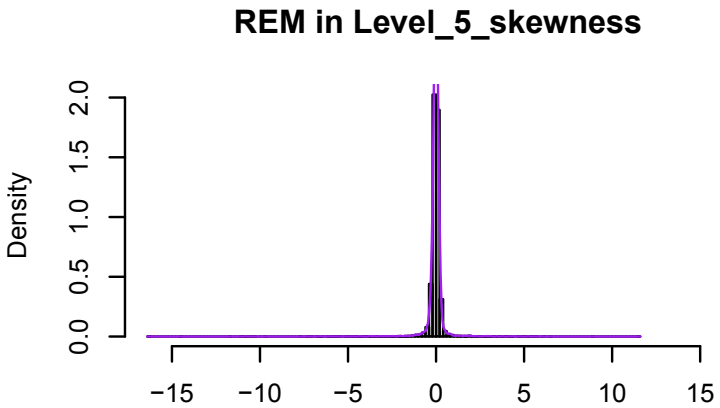

Wake in Level_4_skewness
Density
0.0
0.2
0.4
0.6
0.8
−20
−10
0
10

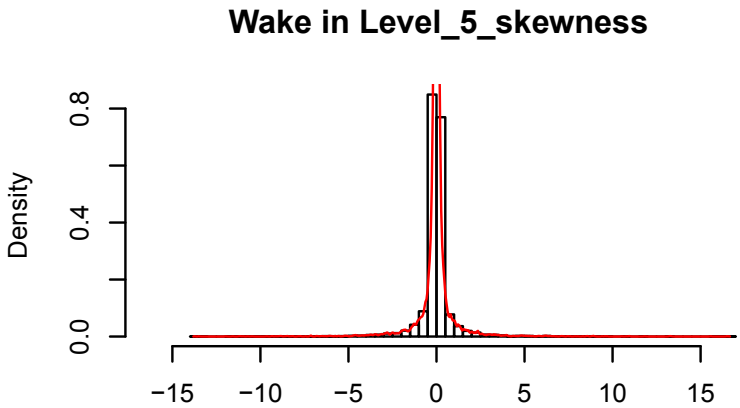

Level_5_skewness
Density
0.0
0.4
0.8
−15
−10
−5
0
5
10
15
WC_Moments234[, kk]

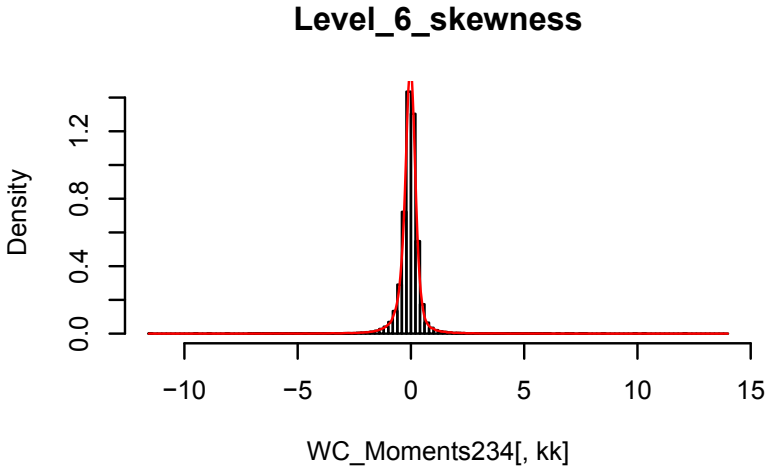

NREM 1 in Level_5_skewness
Density
0.0
0.5
1.0
1.5
−15
−10
−5
0
5
10
15

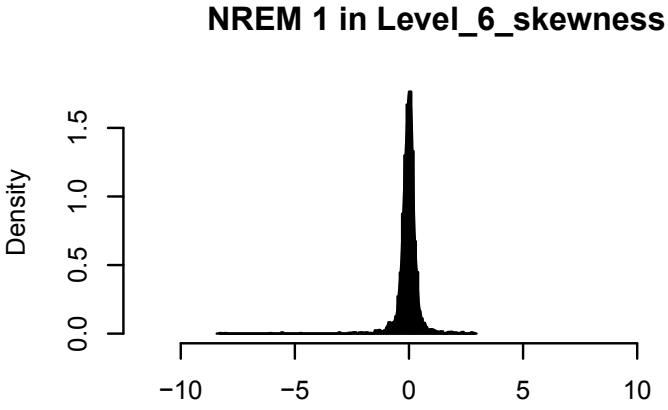

NREM 2 in Level_5_skewness
Density
0.0
0.5
1.0
1.5
−15
−10
−5
0
5
10
15

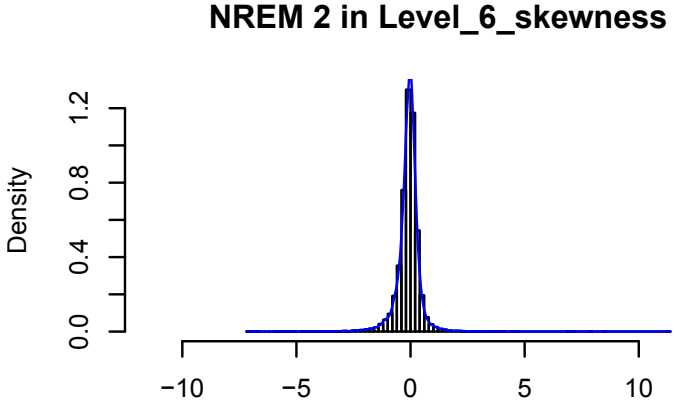

NREM 3 in Level_5_skewness
Density
0.0
0.5
1.0
1.5
2.0
−15
−10
−5
0
5
10
15

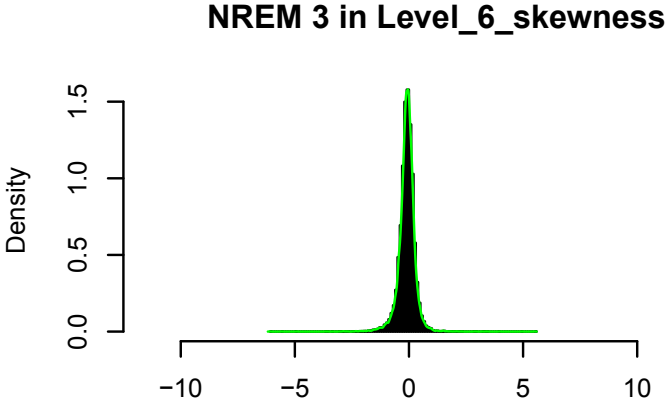

REM in Level_5_skewness
Density
0.0
0.5
1.0
1.5
2.0
−15
−10
−5
0
5
10
15

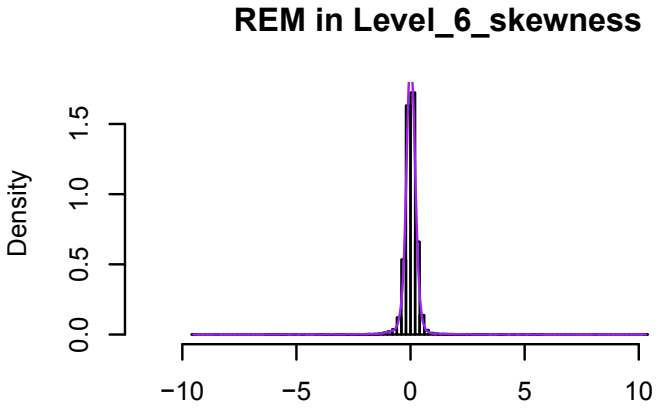

Wake in Level_5_skewness
Density
0.0
0.4
0.8
−15
−10
−5
0
5
10
15

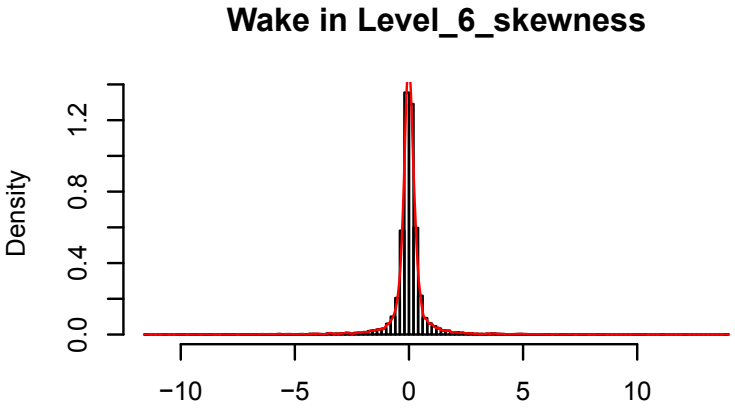

Level_6_skewness
Density
0.0
0.4
0.8
1.2
−10
−5
0
5
10
15
WC_Moments234[, kk]

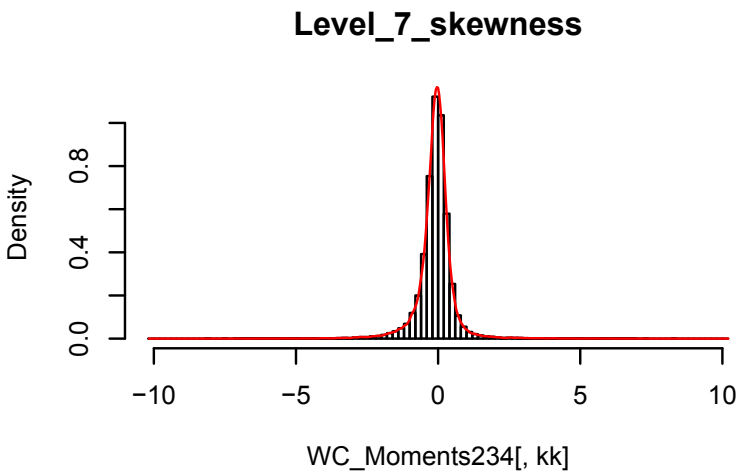

NREM 1 in Level_6_skewness
Density
0.0
0.5
1.0
1.5
−10
−5
0
5
10

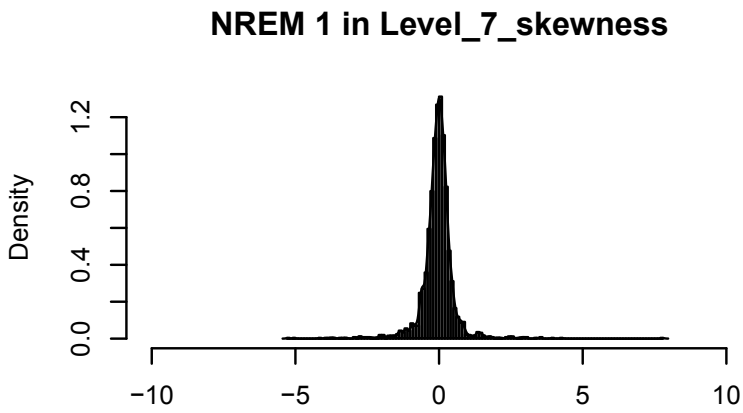

NREM 2 in Level_6_skewness
Density
0.0
0.4
0.8
1.2
−10
−5
0
5
10

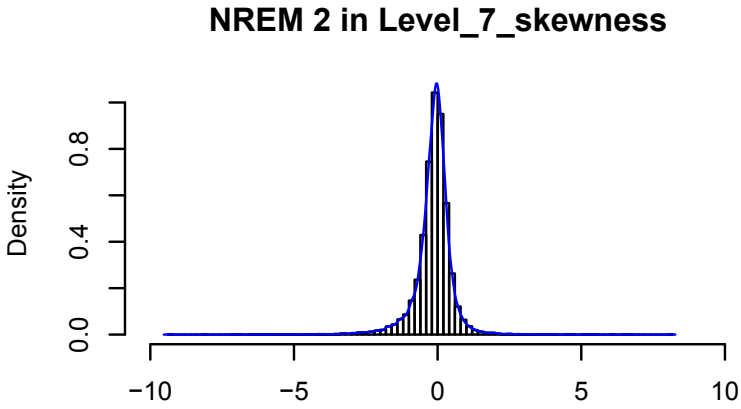

NREM 3 in Level_6_skewness
Density
0.0
0.5
1.0
1.5
−10
−5
0
5
10

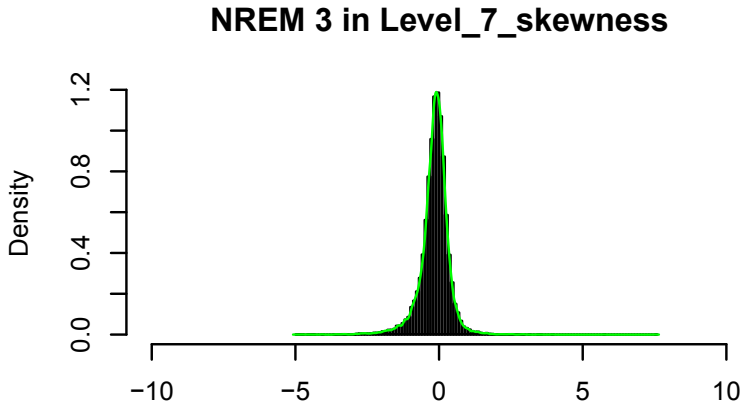

REM in Level_6_skewness
Density
0.0
0.5
1.0
1.5
−10
−5
0
5
10

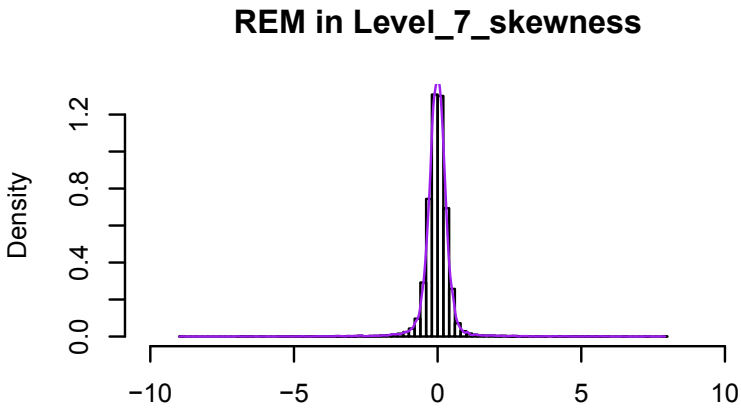

Wake in Level_6_skewness
Density
0.0
0.4
0.8
1.2
−10
−5
0
5
10

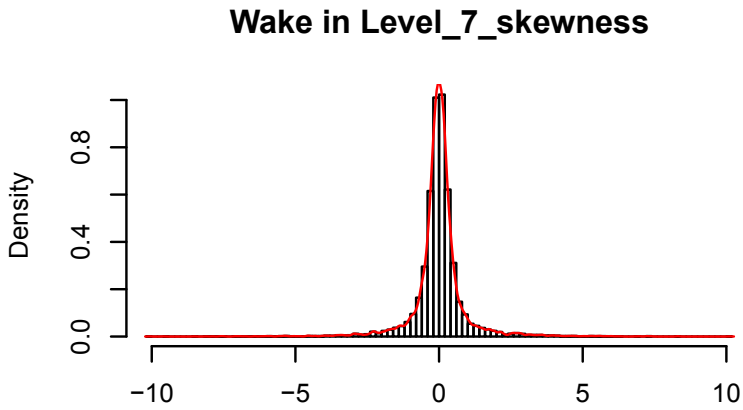

Level_7_skewness
Density
0.0
0.4
0.8
−10
−5
0
5
10
WC_Moments234[, kk]

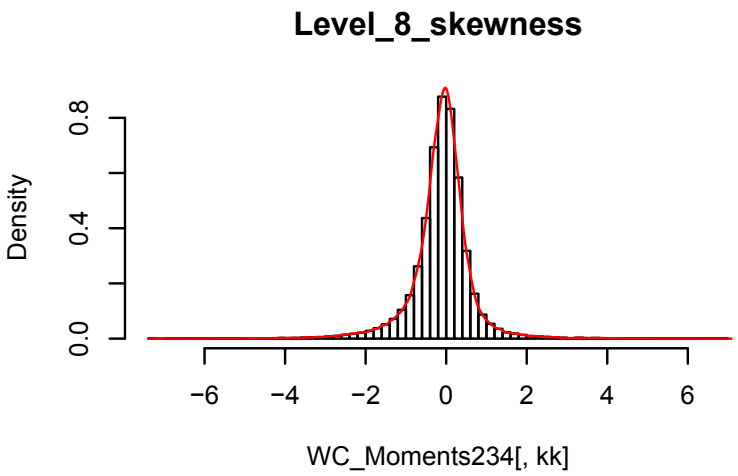

NREM 1 in Level_7_skewness
Density
0.0
0.4
0.8
1.2
−10
−5
0
5
10

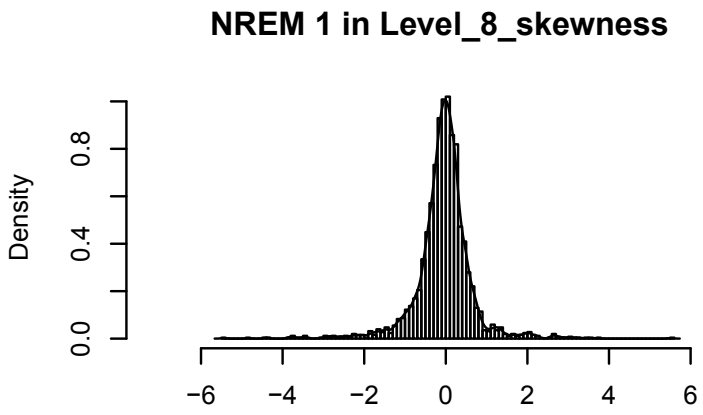

NREM 2 in Level_7_skewness
Density
0.0
0.4
0.8
−10
−5
0
5
10

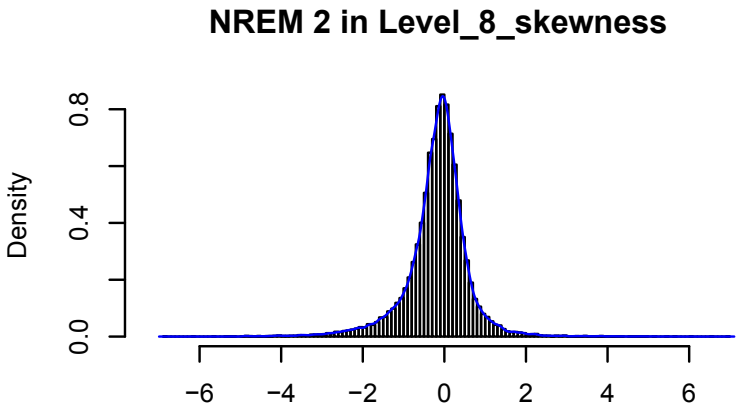

NREM 3 in Level_7_skewness
Density
0.0
0.4
0.8
1.2
−10
−5
0
5
10

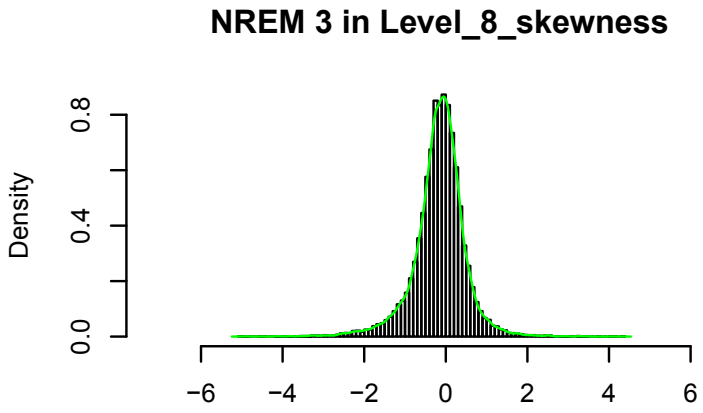

REM in Level_7_skewness
Density
0.0
0.4
0.8
1.2
−10
−5
0
5
10

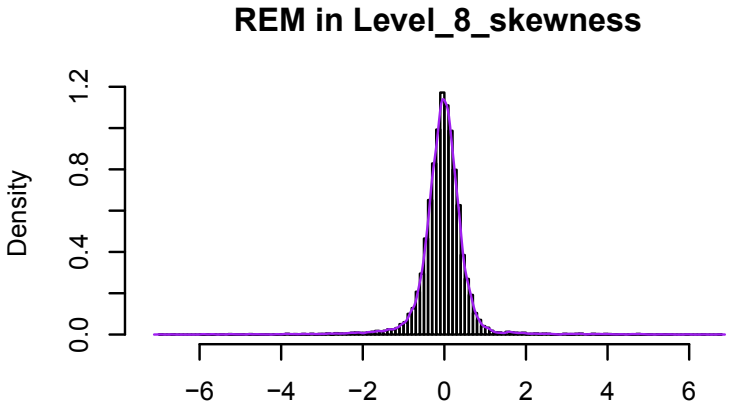

Wake in Level_7_skewness
Density
0.0
0.4
0.8
−10
−5
0
5
10

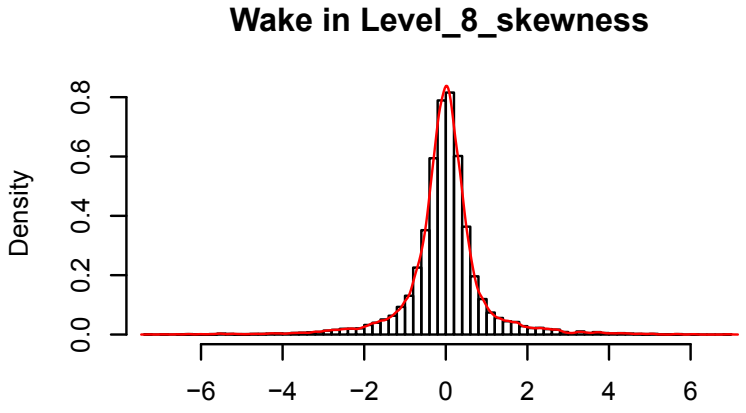

Level_8_skewness
Density
0.0
0.4
0.8
−6
−4
−2
0
2
4
6
WC_Moments234[, kk]

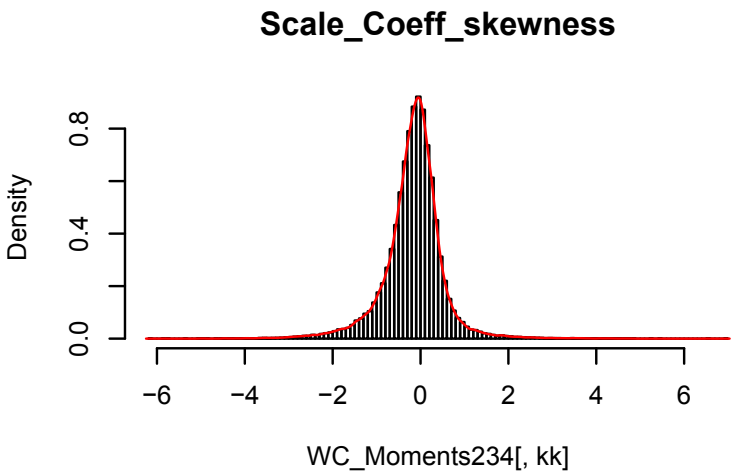

NREM 1 in Level_8_skewness
Density
0.0
0.4
0.8
−6
−4
−2
0
2
4
6

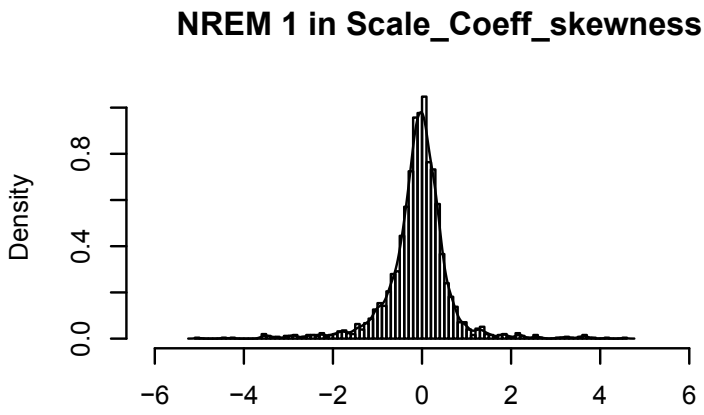

NREM 2 in Level_8_skewness
Density
0.0
0.4
0.8
−6
−4
−2
0
2
4
6

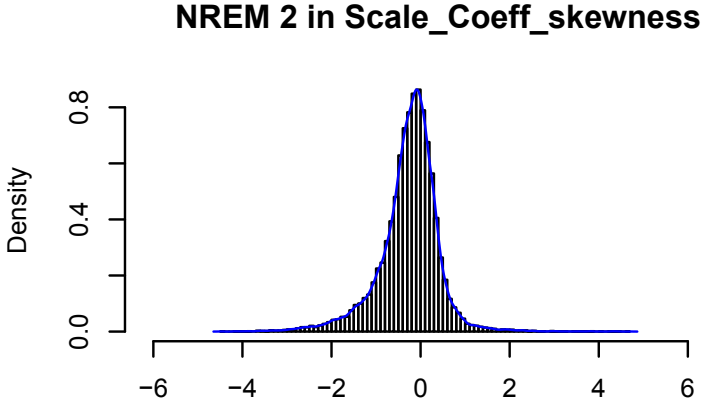

NREM 3 in Level_8_skewness
Density
0.0
0.4
0.8
−6
−4
−2
0
2
4
6

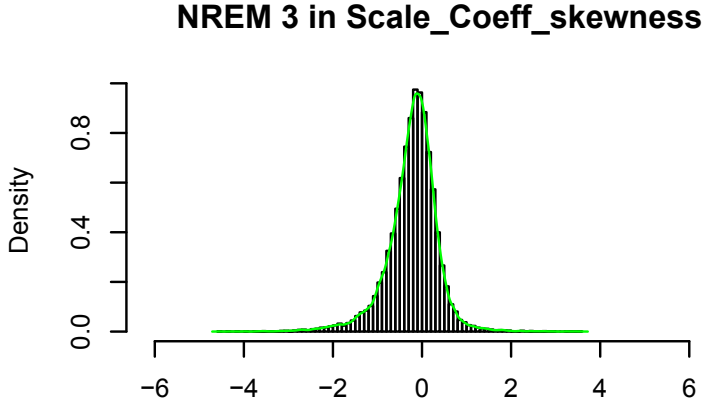

REM in Level_8_skewness
Density
0.0
0.4
0.8
1.2
−6
−4
−2
0
2
4
6

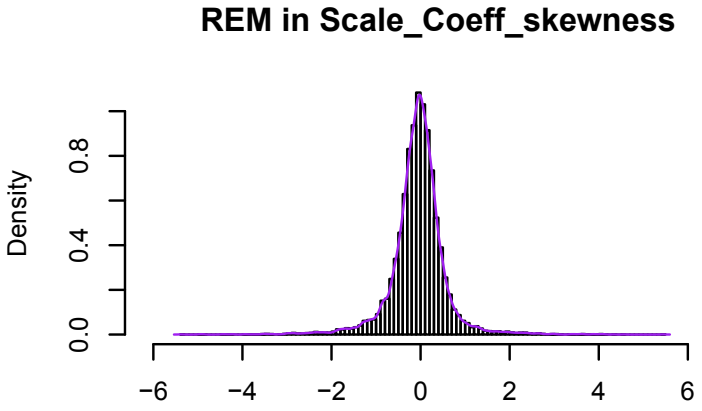

Wake in Level_8_skewness
Density
0.0
0.2
0.4
0.6
0.8
−6
−4
−2
0
2
4
6

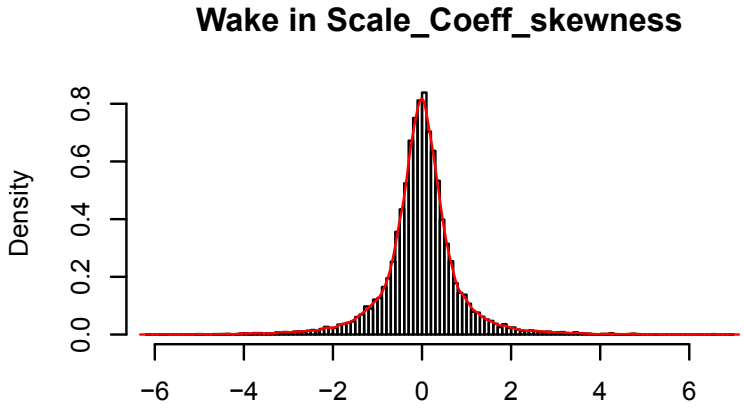

Scale_Coeff_skewness
Density
0.0
0.4
0.8
−6
−4
−2
0
2
4
6
WC_Moments234[, kk]

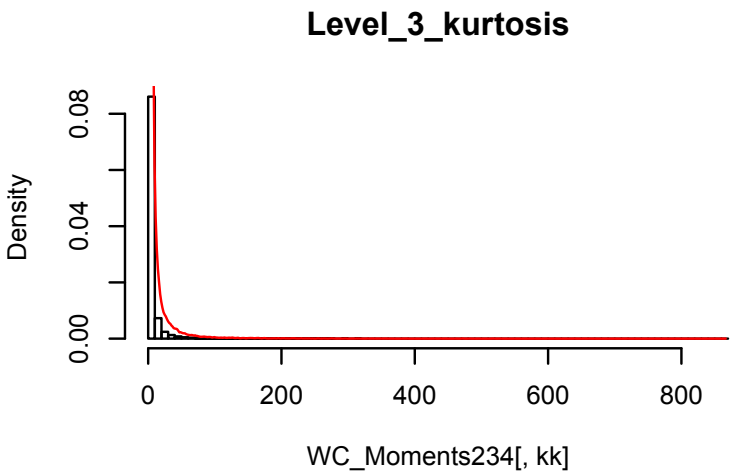

NREM 1 in Scale_Coeff_skewness
Density
0.0
0.4
0.8
−6
−4
−2
0
2
4
6

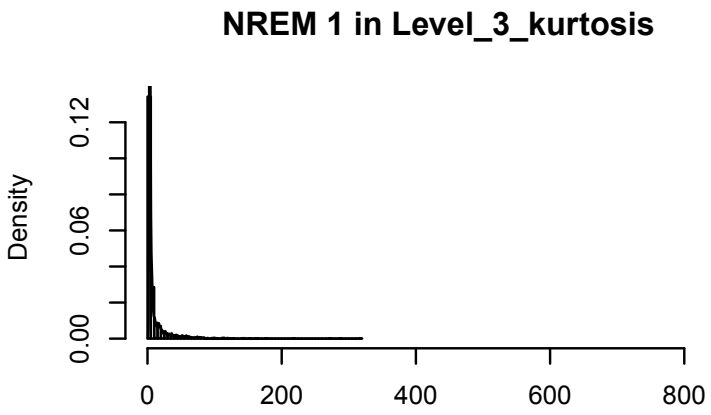

NREM 2 in Scale_Coeff_skewness
Density
0.0
0.4
0.8
−6
−4
−2
0
2
4
6

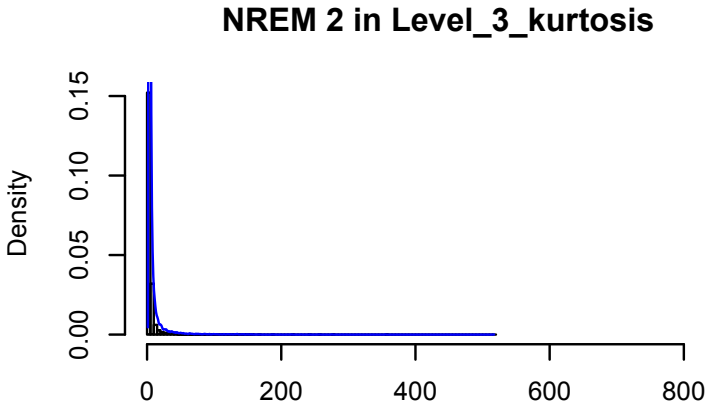

NREM 3 in Scale_Coeff_skewness
Density
0.0
0.4
0.8
−6
−4
−2
0
2
4
6

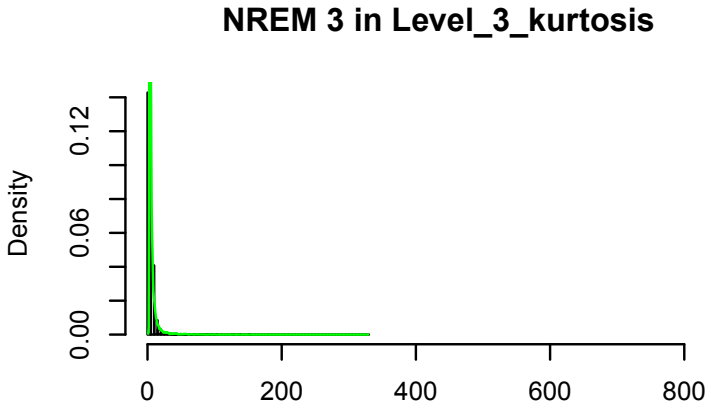

REM in Scale_Coeff_skewness
Density
0.0
0.4
0.8
−6
−4
−2
0
2
4
6

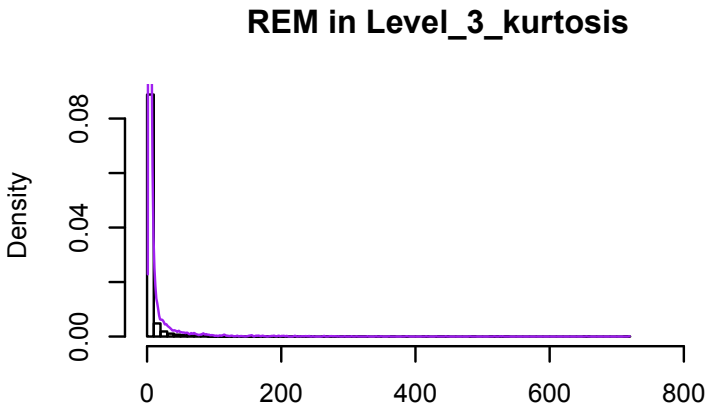

Wake in Scale_Coeff_skewness
Density
0.0
0.2
0.4
0.6
0.8
−6
−4
−2
0
2
4
6

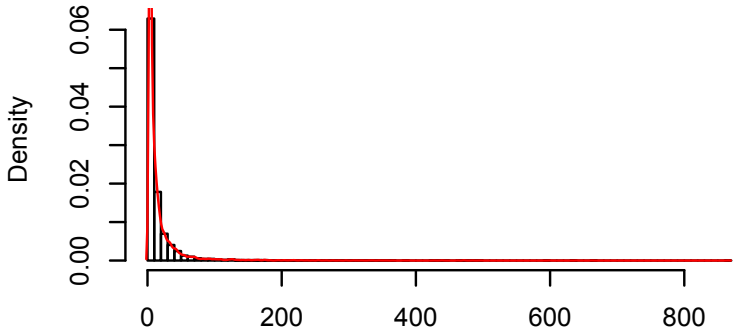

Level_3_kurtosis
Density
0.00
0.04
0.08
0
200
400
600
800
WC_Moments234[, kk]

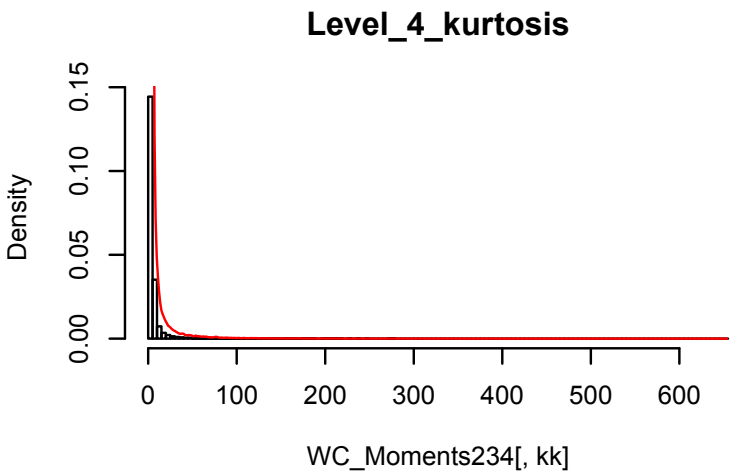

NREM 1 in Level_3_kurtosis
Density
0.00
0.06
0.12
0
200
400
600
800

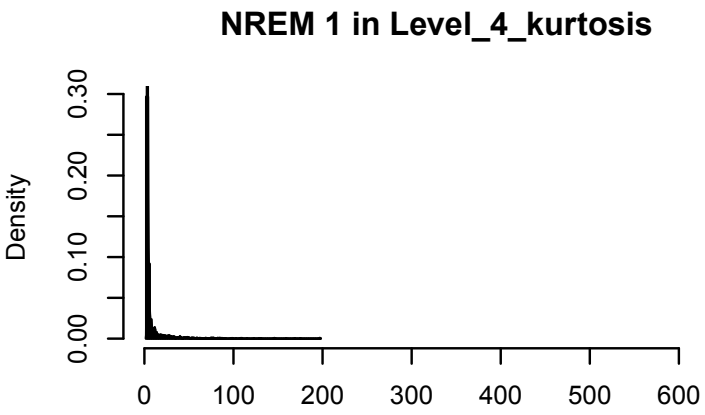

NREM 2 in Level_3_kurtosis
Density
0.00
0.05
0.10
0.15
0
200
400
600
800

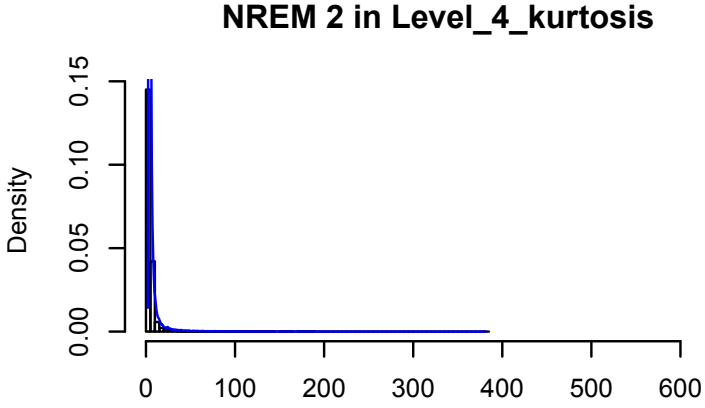

NREM 3 in Level_3_kurtosis
Density
0.00
0.06
0.12
0
200
400
600
800

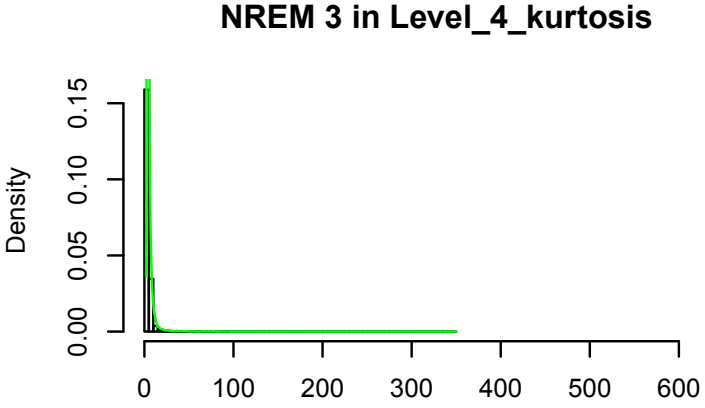

REM in Level_3_kurtosis
Density
0.00
0.04
0.08
0
200
400
600
800

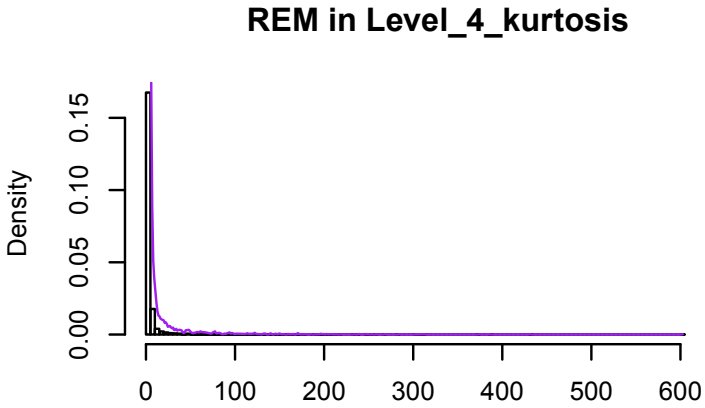

Wake in Level_3_kurtosis
Density
0.00
0.02
0.04
0.06
0
200
400
600
800

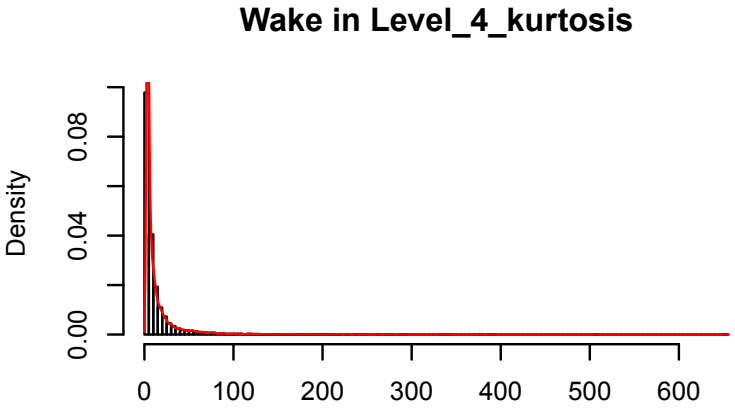

Level_4_kurtosis
Density
0.00
0.05
0.10
0.15
0
100
200
300
400
500
600
WC_Moments234[, kk]

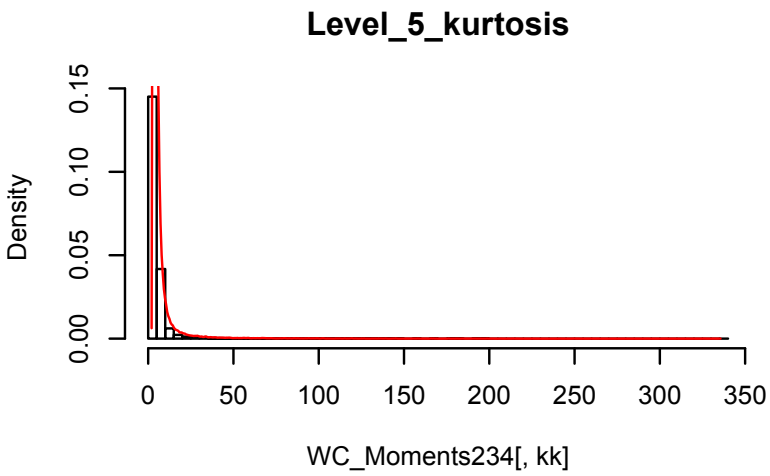

NREM 1 in Level_4_kurtosis
Density
0.00
0.10
0.20
0.30
0
100
200
300
400
500
600

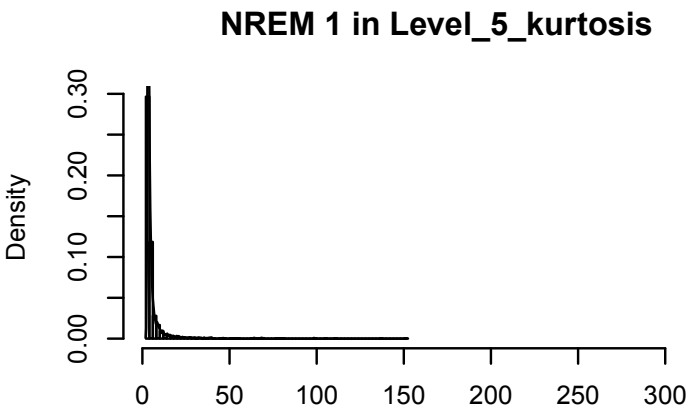

NREM 2 in Level_4_kurtosis
Density
0.00
0.05
0.10
0.15
0
100
200
300
400
500
600

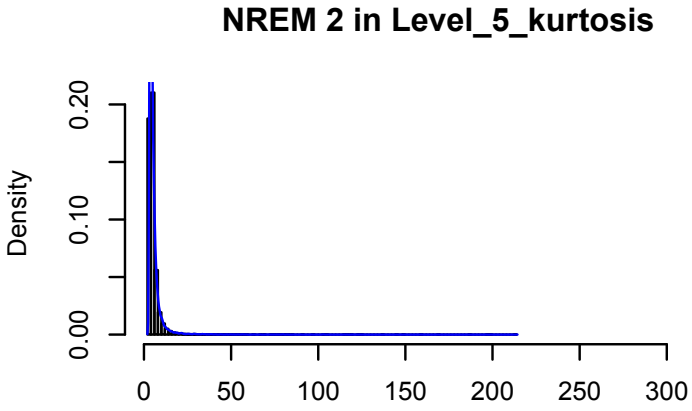

NREM 3 in Level_4_kurtosis
Density
0.00
0.05
0.10
0.15
0
100
200
300
400
500
600

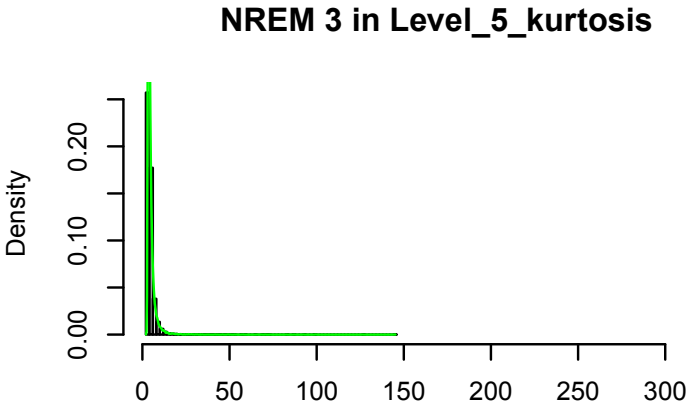

REM in Level_4_kurtosis
Density
0.00
0.05
0.10
0.15
0
100
200
300
400
500
600

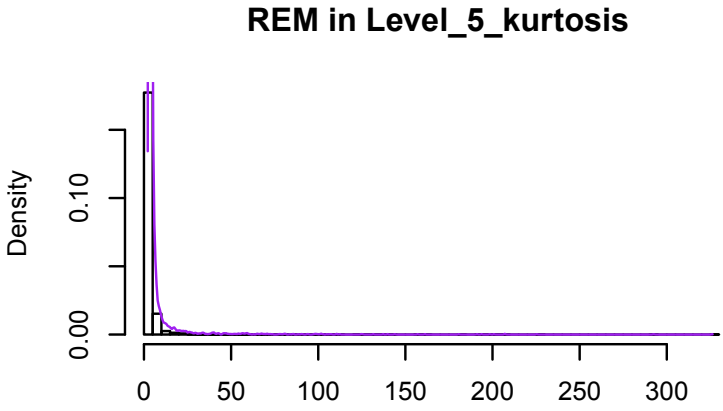

Wake in Level_4_kurtosis
Density
0.00
0.04
0.08
0
100
200
300
400
500
600

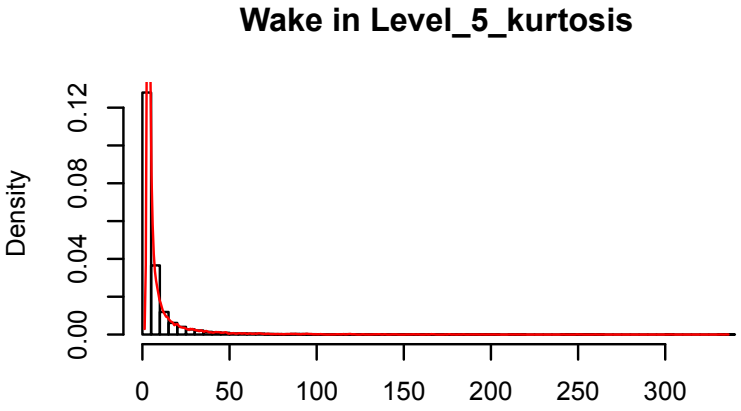

Level_5_kurtosis
Density
0.00
0.05
0.10
0.15
0
50
100
150
200
250
300
350
WC_Moments234[, kk]

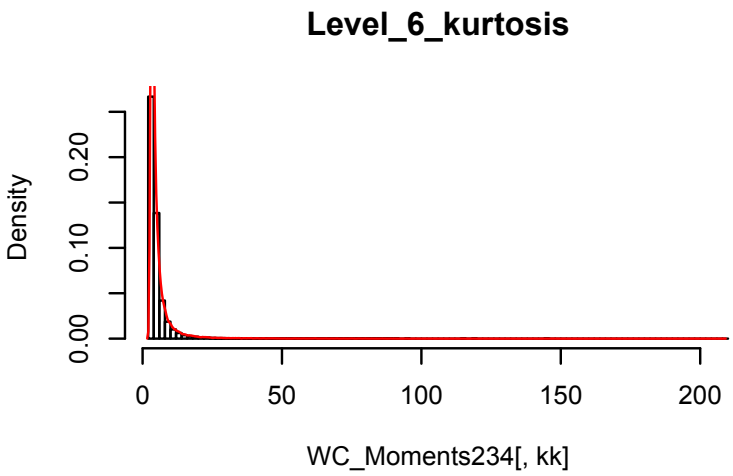

NREM 1 in Level_5_kurtosis
Density
0.00
0.10
0.20
0.30
0
50
100
150
200
250
300

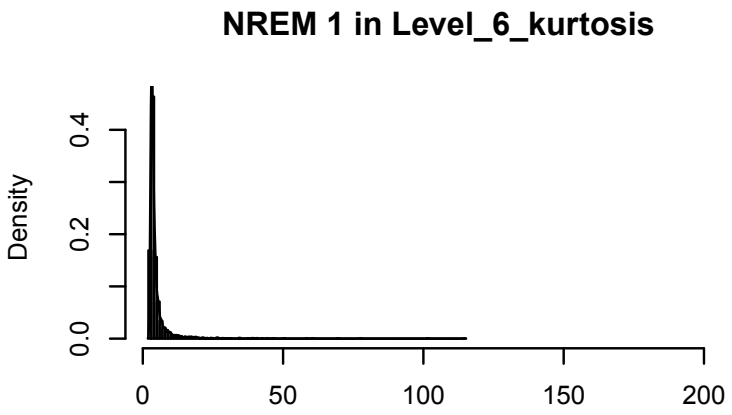

NREM 2 in Level_5_kurtosis
Density
0.00
0.10
0.20
0
50
100
150
200
250
300

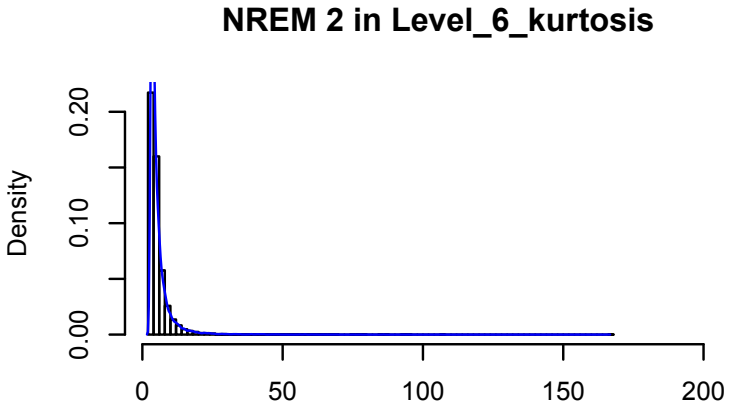

NREM 3 in Level_5_kurtosis
Density
0.00
0.10
0.20
0
50
100
150
200
250
300

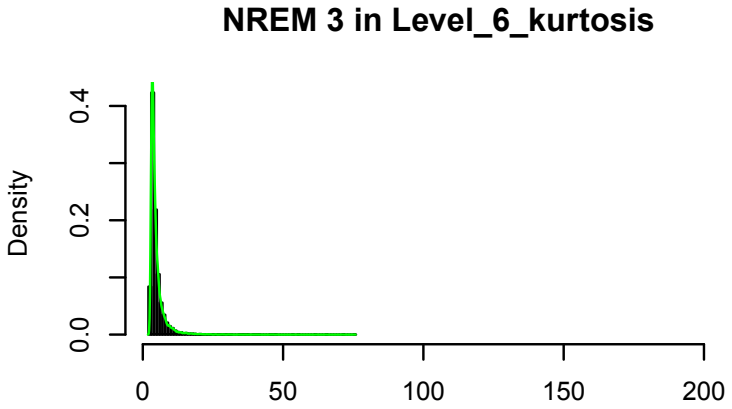

REM in Level_5_kurtosis
Density
0.00
0.10
0
50
100
150
200
250
300

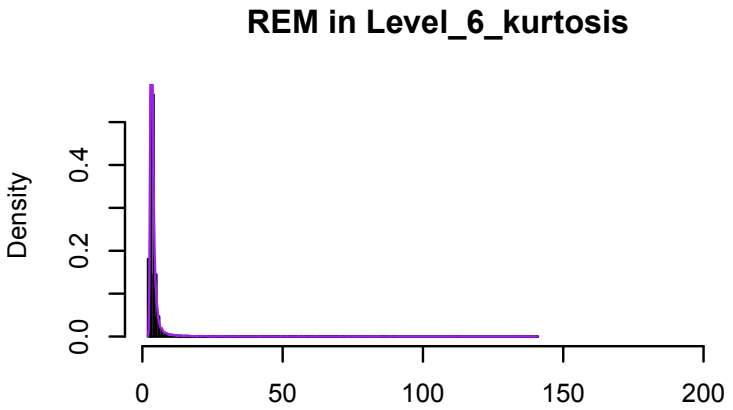

Wake in Level_5_kurtosis
Density
0.00
0.04
0.08
0.12
0
50
100
150
200
250
300

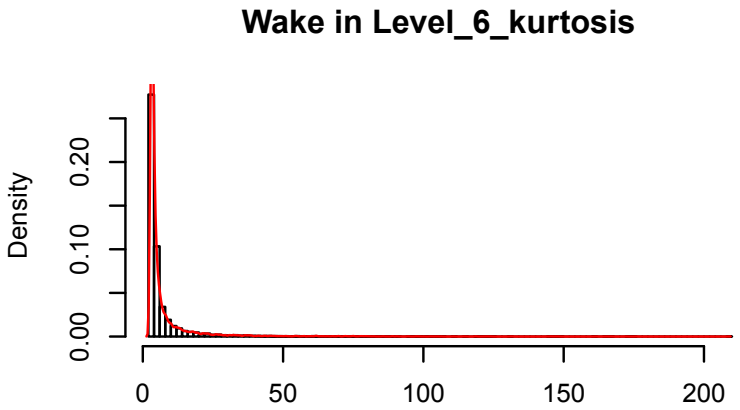

Level_6_kurtosis
Density
0.00
0.10
0.20
0
50
100
150
200
WC_Moments234[, kk]

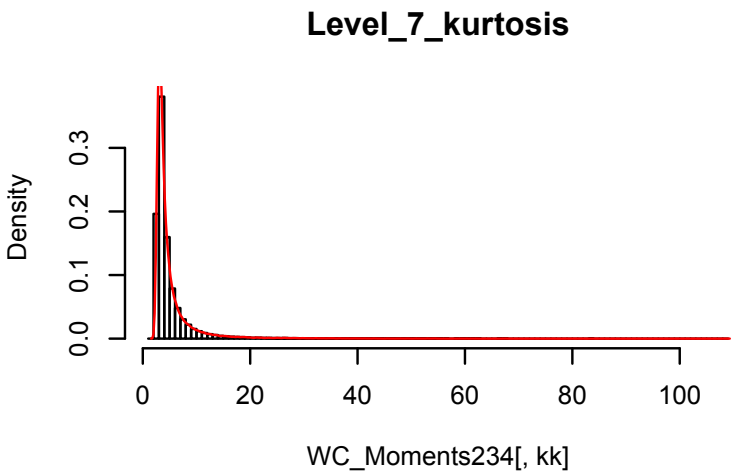

NREM 1 in Level_6_kurtosis
Density
0.0
0.2
0.4
0
50
100
150
200

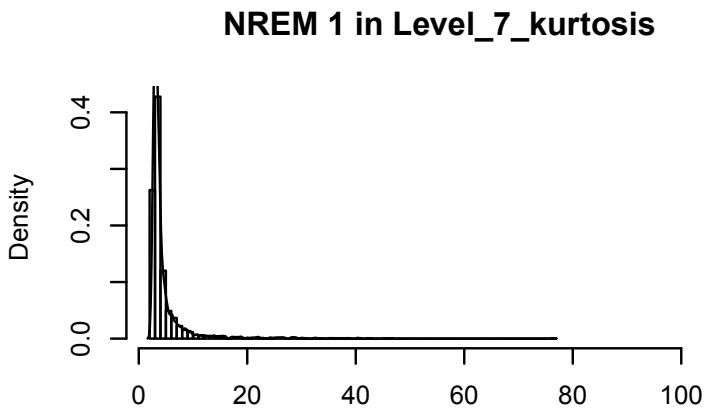

NREM 2 in Level_6_kurtosis
Density
0.00
0.10
0.20
0
50
100
150
200

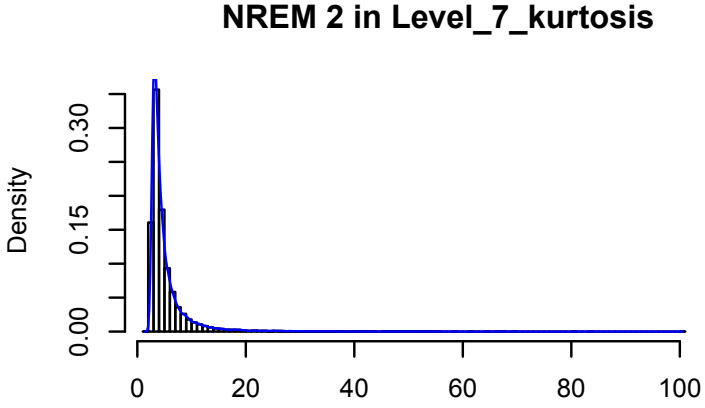

NREM 3 in Level_6_kurtosis
Density
0.0
0.2
0.4
0
50
100
150
200

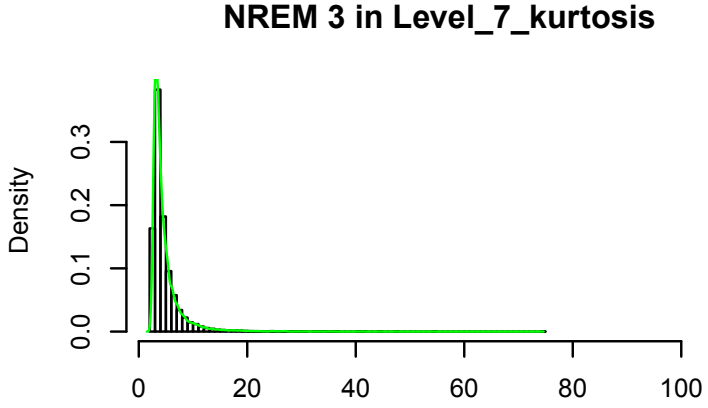

REM in Level_6_kurtosis
Density
0.0
0.2
0.4
0
50
100
150
200

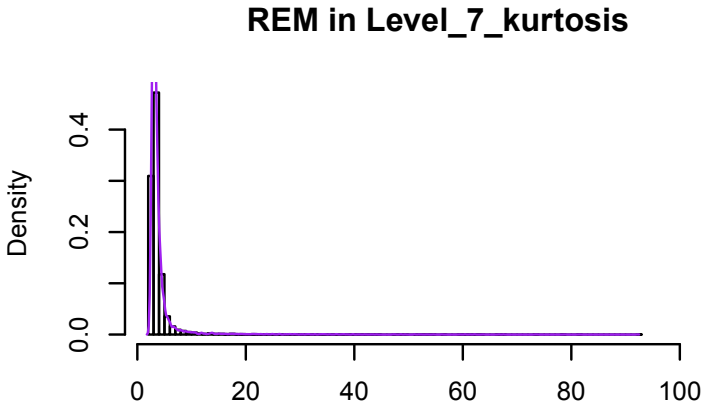

Wake in Level_6_kurtosis
Density
0.00
0.10
0.20
0
50
100
150
200

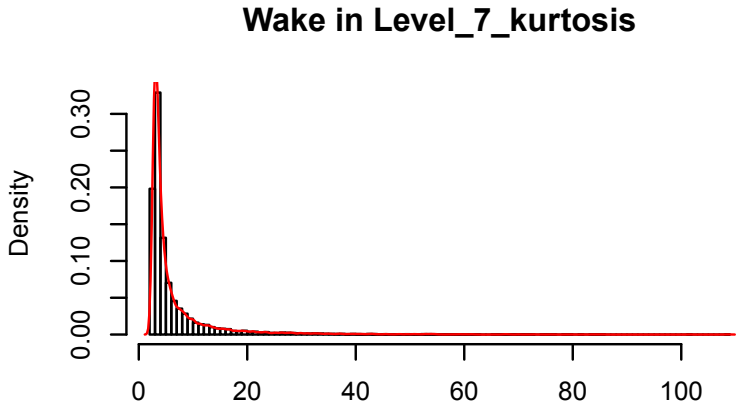

Level_7_kurtosis
Density
0.0
0.1
0.2
0.3
0
20
40
60
80
100
WC_Moments234[, kk]

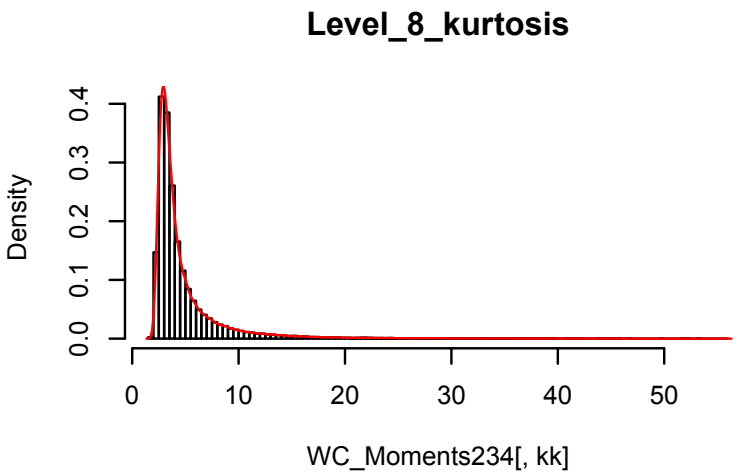

NREM 1 in Level_7_kurtosis
Density
0.0
0.2
0.4
0
20
40
60
80
100

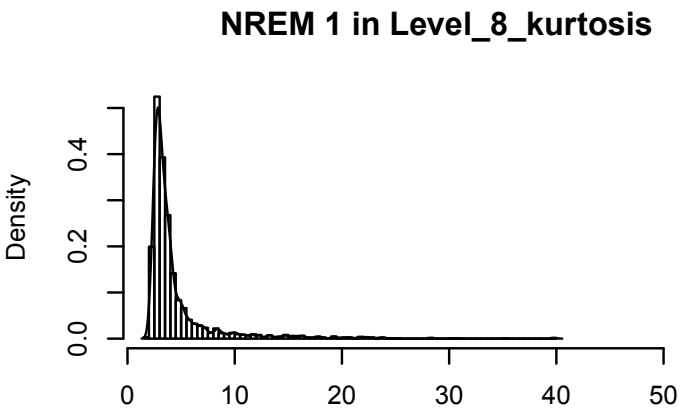

NREM 2 in Level_7_kurtosis
Density
0.00
0.15
0.30
0
20
40
60
80
100

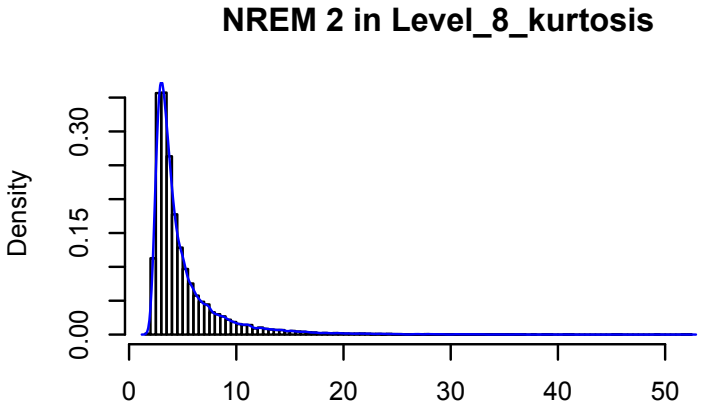

NREM 3 in Level_7_kurtosis
Density
0.0
0.1
0.2
0.3
0
20
40
60
80
100

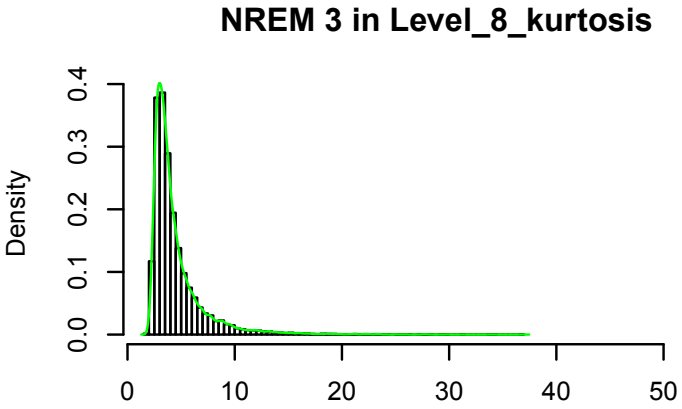

REM in Level_7_kurtosis
Density
0.0
0.2
0.4
0
20
40
60
80
100

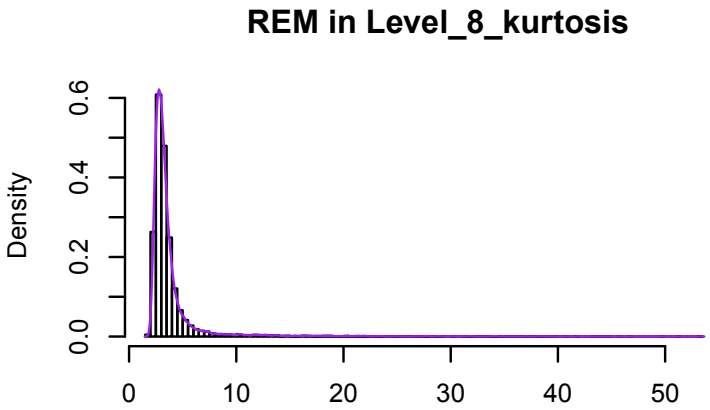

Wake in Level_7_kurtosis
Density
0.00
0.10
0.20
0.30
0
20
40
60
80
100

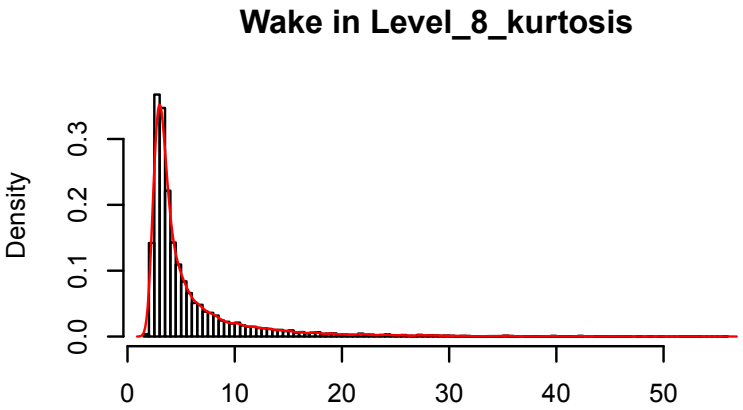

Level_8_kurtosis
Density
0.0
0.1
0.2
0.3
0.4
0
10
20
30
40
50
WC_Moments234[, kk]

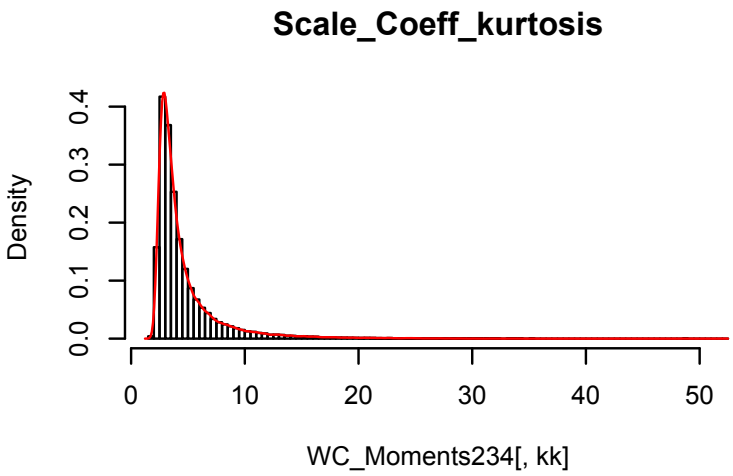

NREM 1 in Level_8_kurtosis
Density
0.0
0.2
0.4
0
10
20
30
40
50

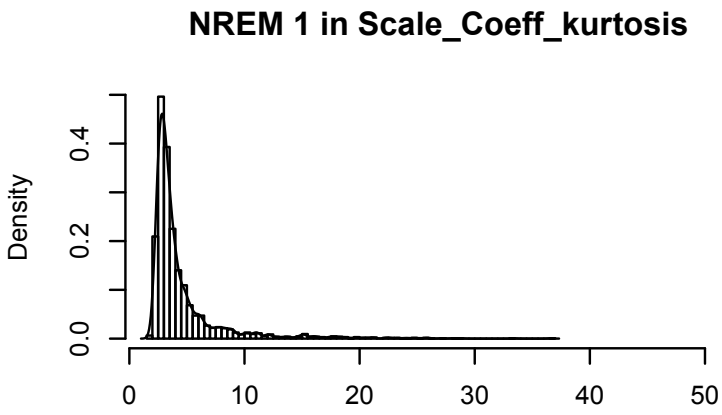

NREM 2 in Level_8_kurtosis
Density
0.00
0.15
0.30
0
10
20
30
40
50

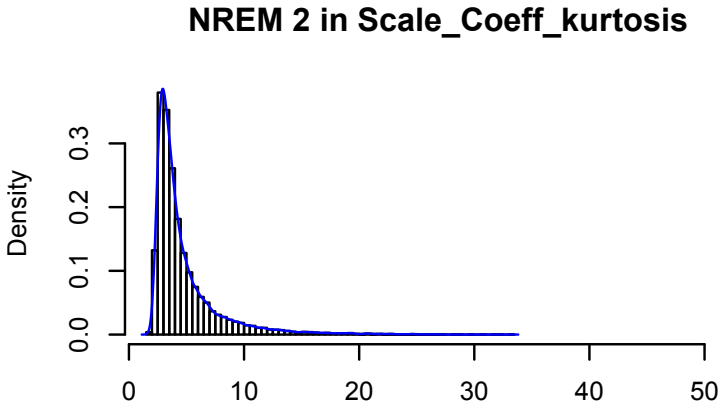

NREM 3 in Level_8_kurtosis
Density
0.0
0.1
0.2
0.3
0.4
0
10
20
30
40
50

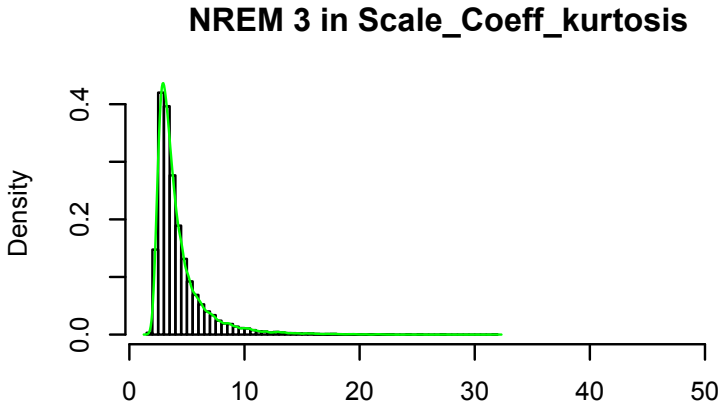

REM in Level_8_kurtosis
Density
0.0
0.2
0.4
0.6
0
10
20
30
40
50

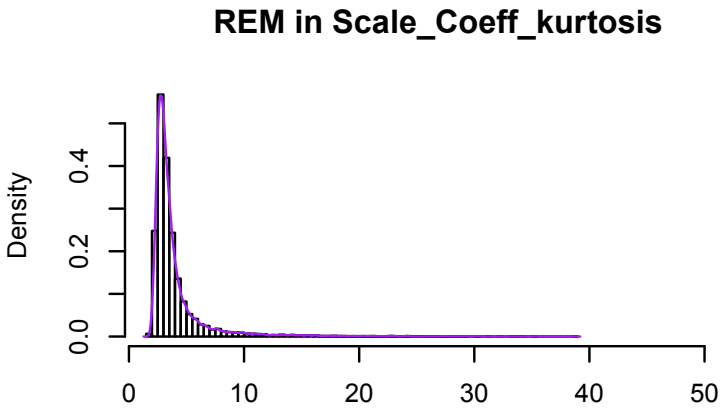

Wake in Level_8_kurtosis
Density
0.0
0.1
0.2
0.3
0
10
20
30
40
50

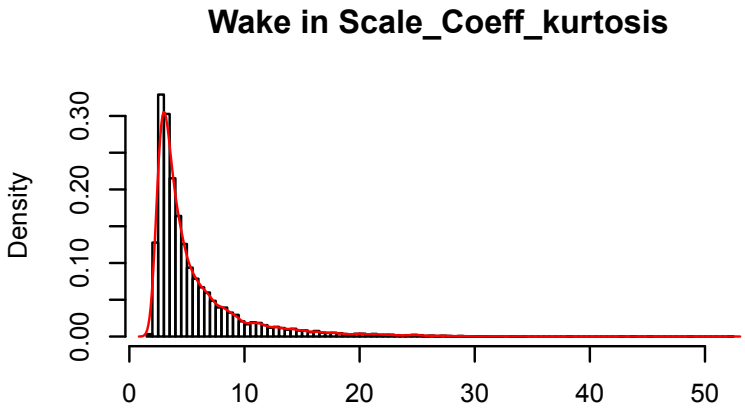

Scale_Coeff_kurtosis
Density
0.0
0.1
0.2
0.3
0.4
0
10
20
30
40
50
WC_Moments234[, kk]

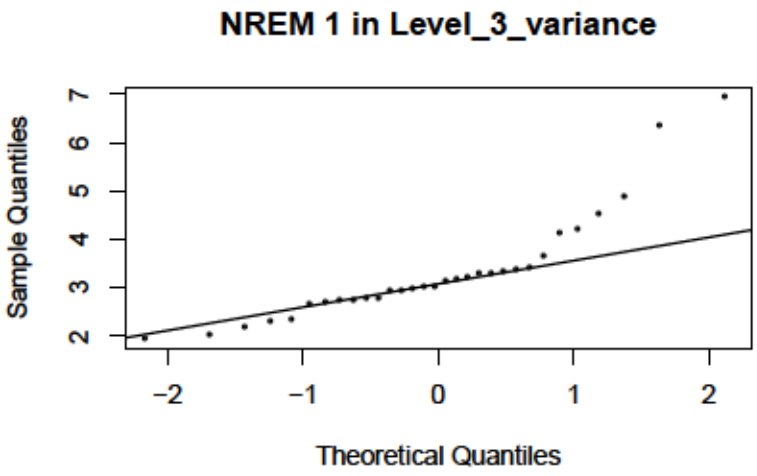

NREM 1 in Scale_Coeff_kurtosis
Density
0.0
0.2
0.4
0
10
20
30
40
50

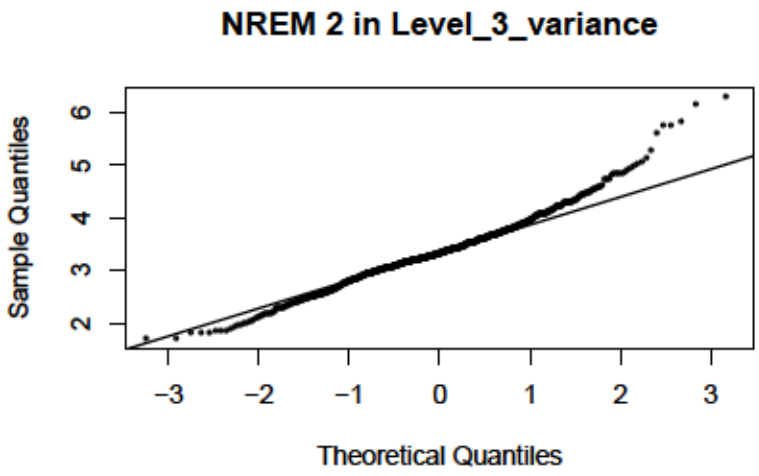

NREM 2 in Scale_Coeff_kurtosis
Density
0.0
0.1
0.2
0.3
0
10
20
30
40
50

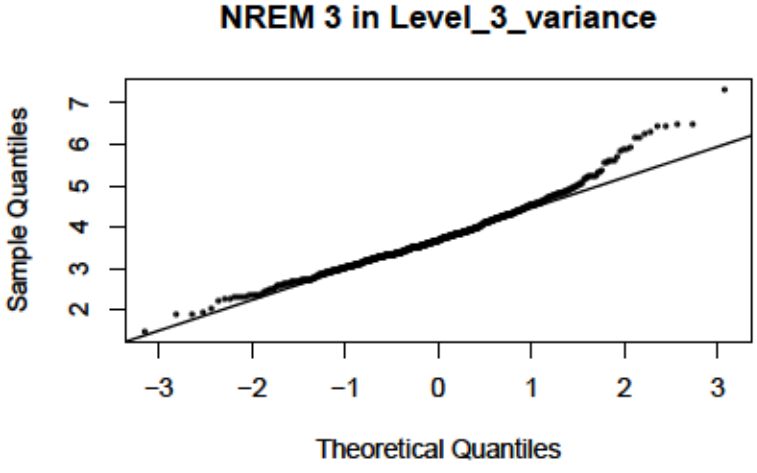

NREM 3 in Scale_Coeff_kurtosis
Density
0.0
0.2
0.4
0
10
20
30
40
50

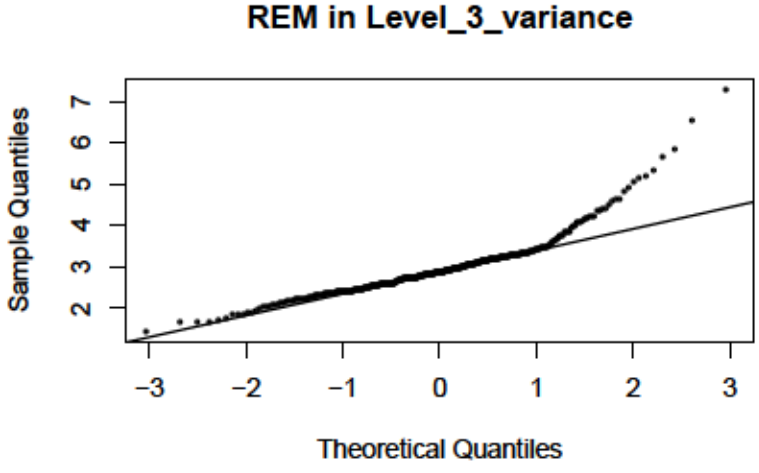

REM in Scale_Coeff_kurtosis
Density
0.0
0.2
0.4
0
10
20
30
40
50

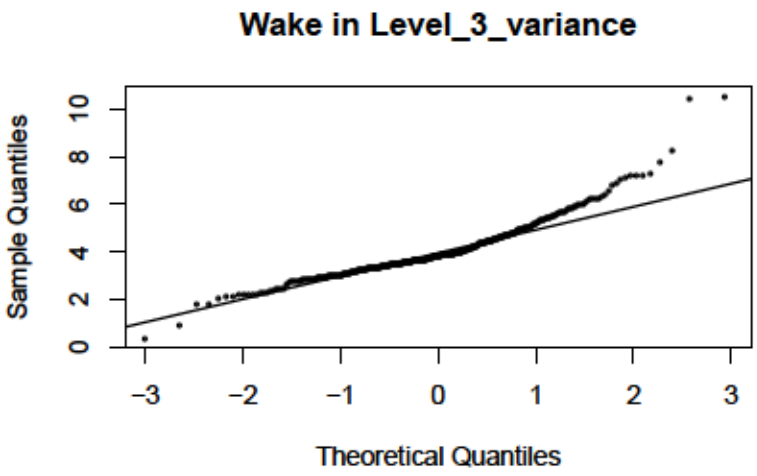

Wake in Scale_Coeff_kurtosis
Density
0.00
0.10
0.20
0.30
0
10
20
30
40
50

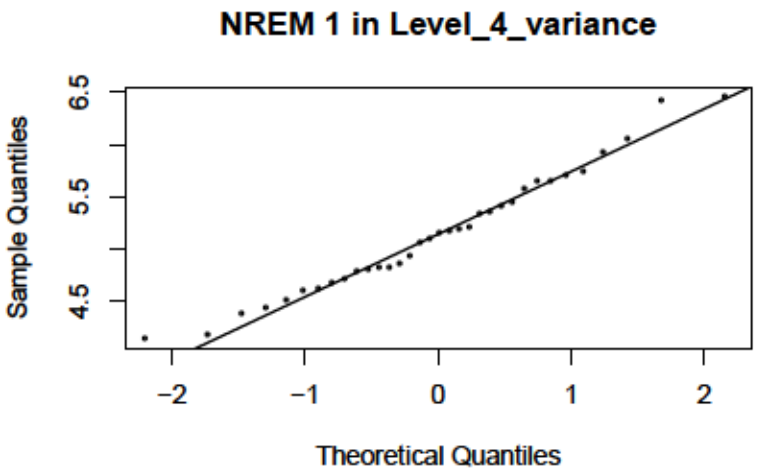

NREM 1 in Level_3_variance
Sample Quantiles
Theoretical Quantiles

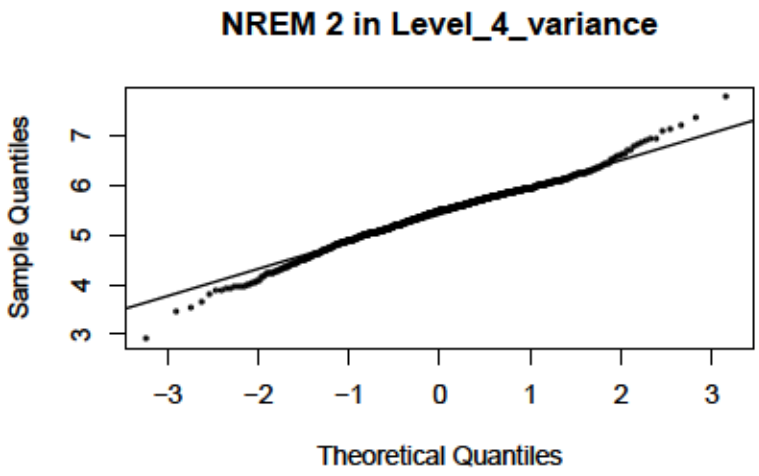

NREM 2 in Level_3_variance
Sample Quantiles
Theoretical Quantiles

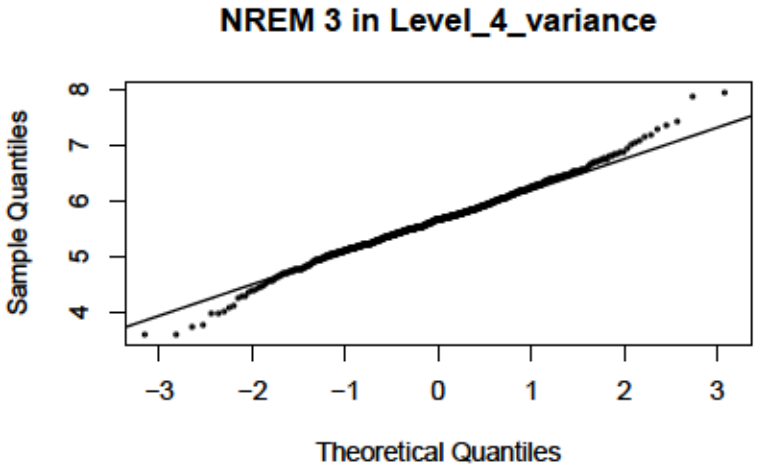

NREM 3 in Level_3_variance
Sample Quantiles
Theoretical Quantiles

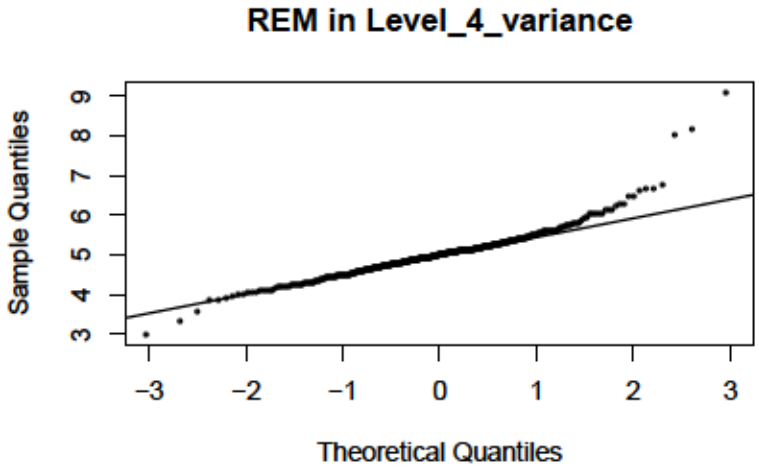

REM in Level_3_variance
Sample Quantiles
Theoretical Quantiles

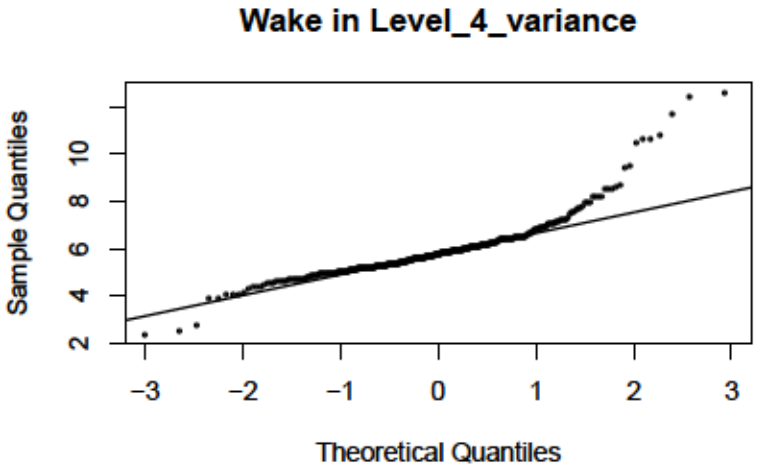

Wake in Level_3_variance
Sample Quantiles
Theoretical Quantiles

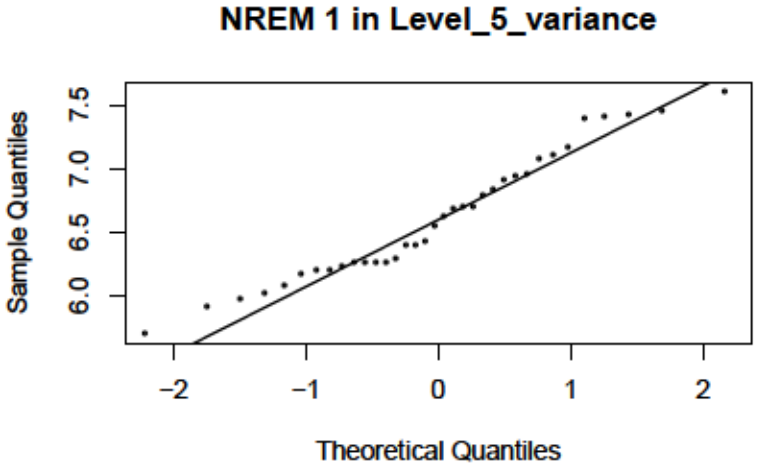

NREM 1 in Level_4_variance
Sample Quantiles
Theoretical Quantiles

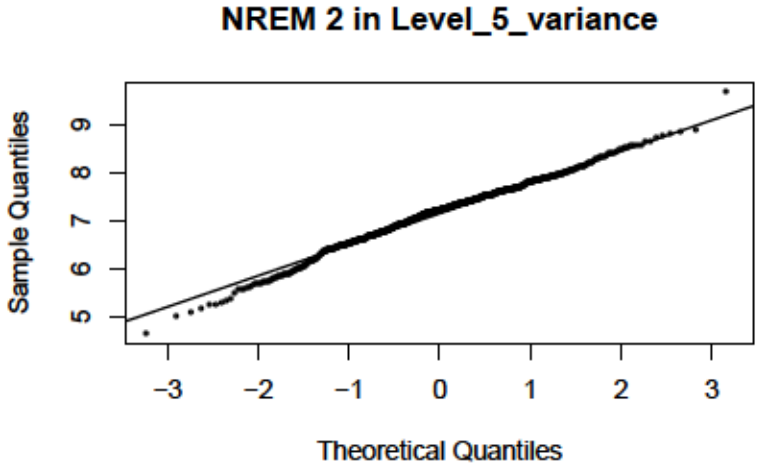

NREM 2 in Level_4_variance
Sample Quantiles
Theoretical Quantiles

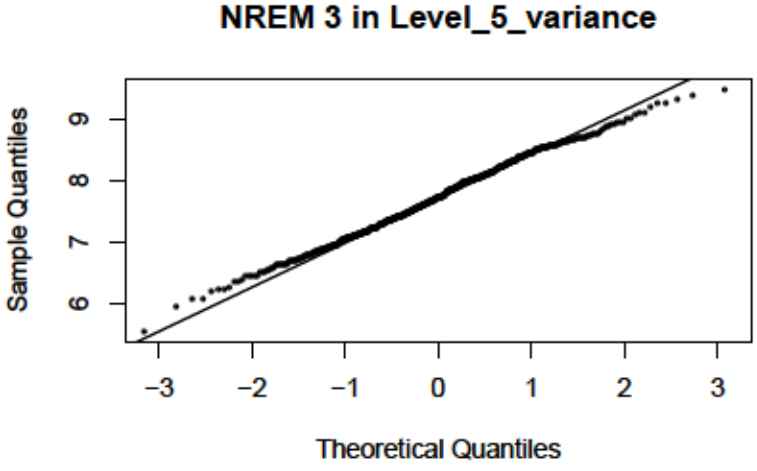

NREM 3 in Level_4_variance
Sample Quantiles
Theoretical Quantiles

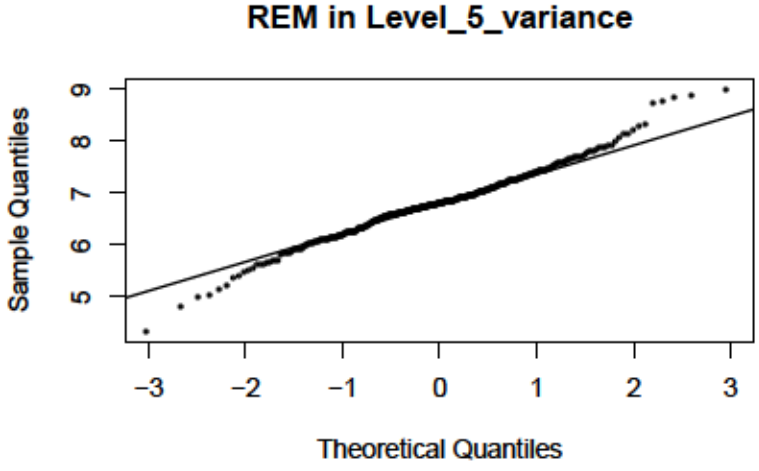

REM in Level_4_variance
Sample Quantiles
Theoretical Quantiles

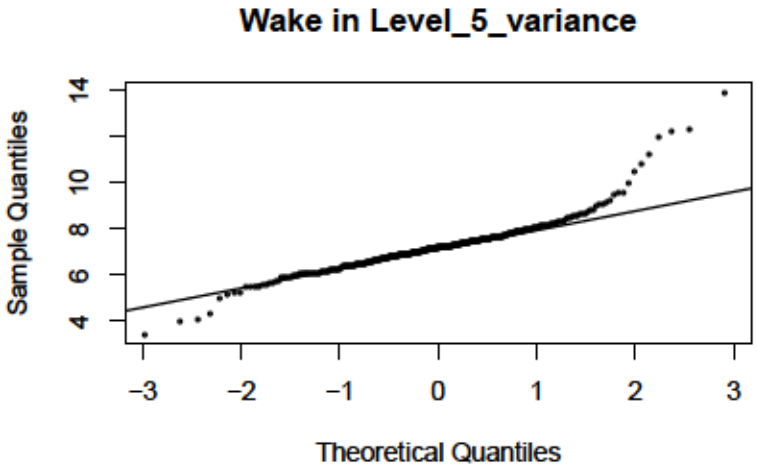

Wake in Level_4_variance
Sample Quantiles
Theoretical Quantiles

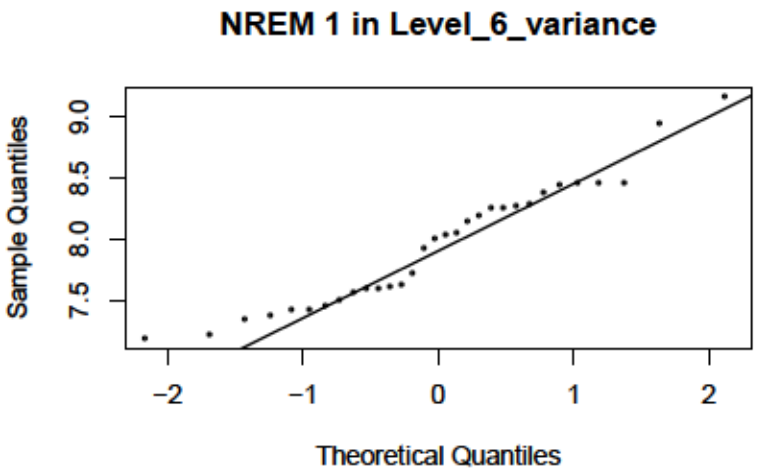

NREM 1 in Level_5_variance
Sample Quantiles
Theoretical Quantiles

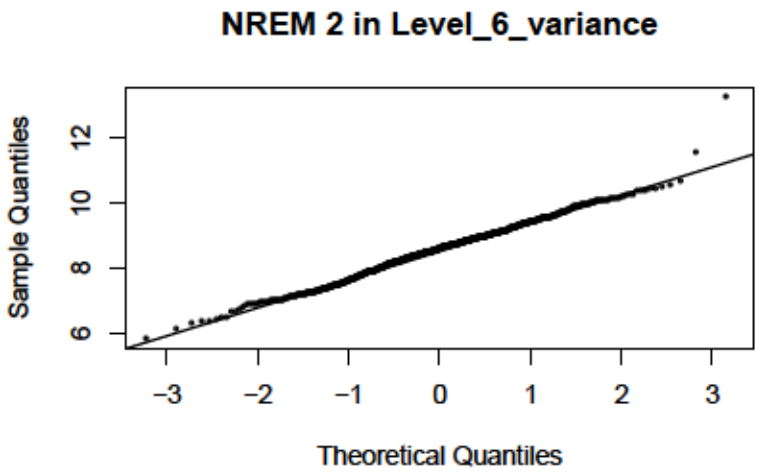

NREM 2 in Level_5_variance
Sample Quantiles
Theoretical Quantiles

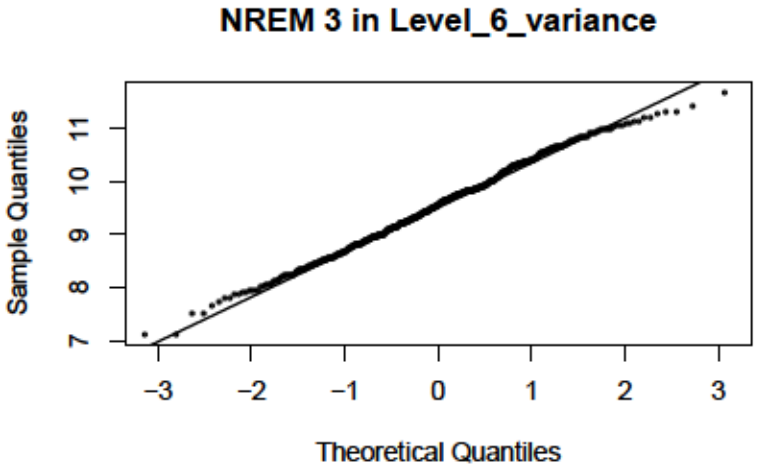

NREM 3 in Level_5_variance
Sample Quantiles
Theoretical Quantiles

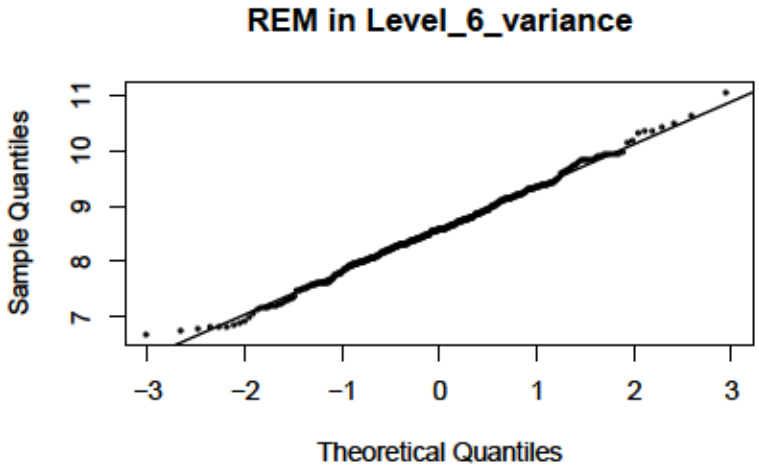

REM in Level_5_variance
Sample Quantiles
Theoretical Quantiles

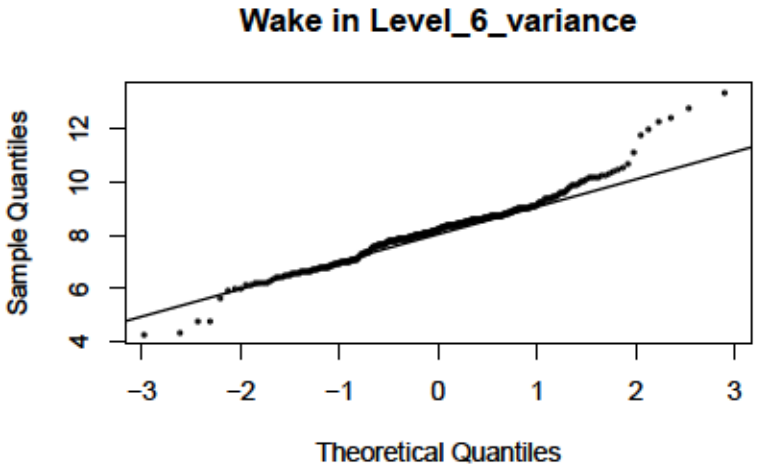

Wake in Level_5_variance
Sample Quantiles
Theoretical Quantiles

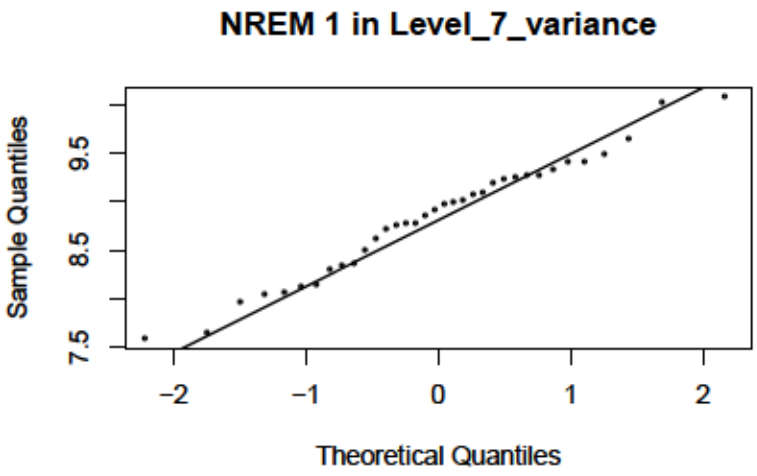

NREM 1 in Level_6_variance
Sample Quantiles
Theoretical Quantiles

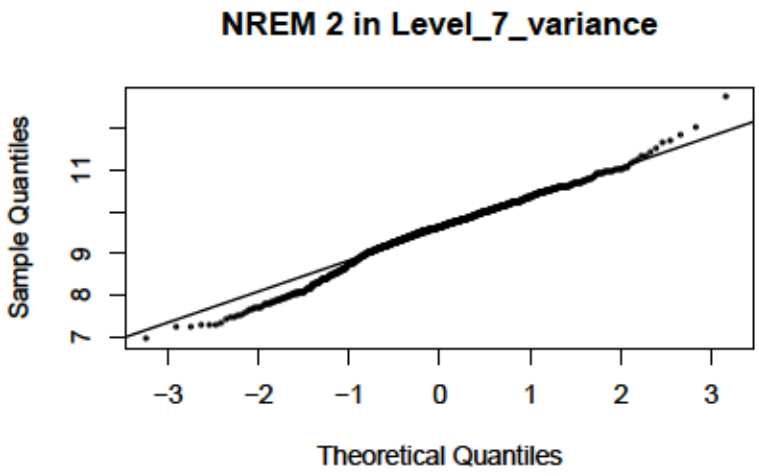

NREM 2 in Level_6_variance
Sample Quantiles
Theoretical Quantiles

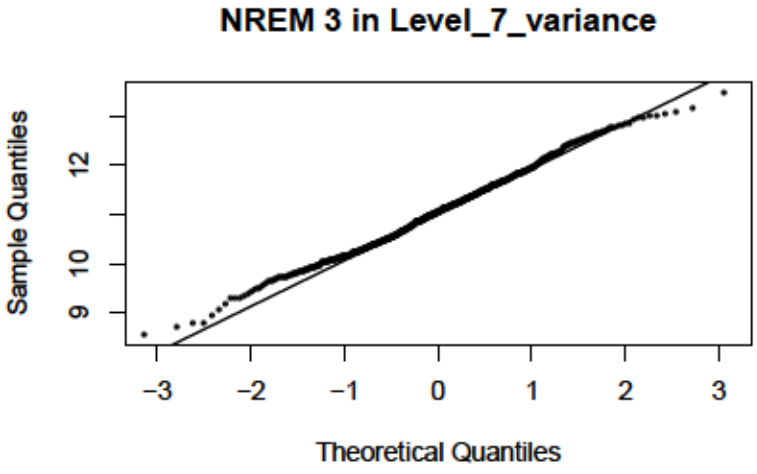

NREM 3 in Level_6_variance
Sample Quantiles
Theoretical Quantiles

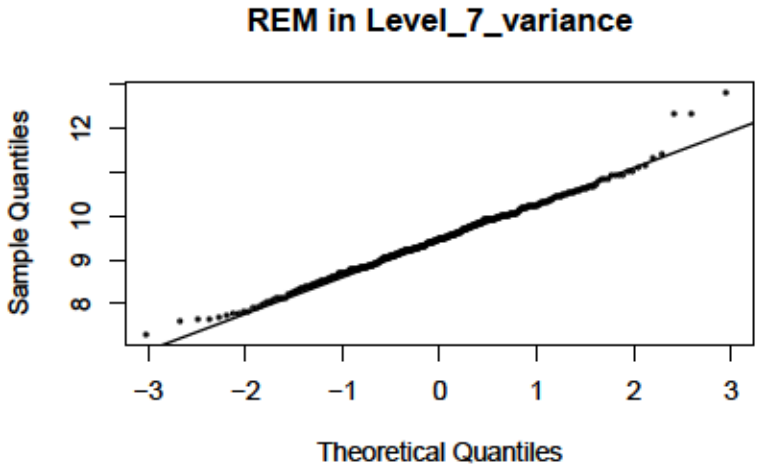

REM in Level_6_variance
Sample Quantiles
Theoretical Quantiles

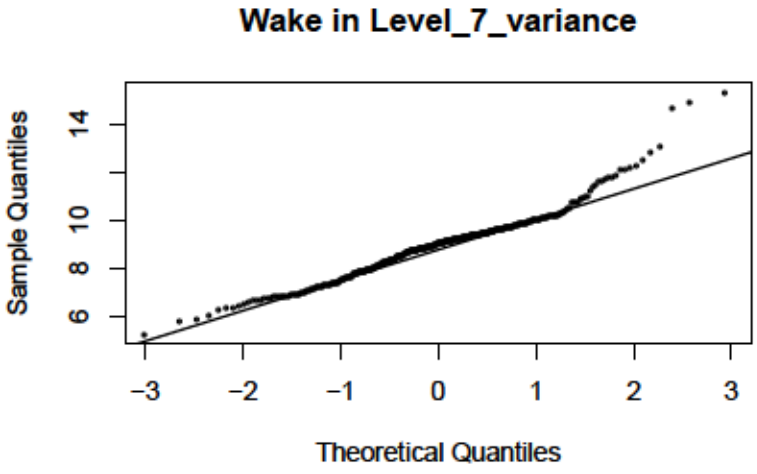

Wake in Level_6_variance
Sample Quantiles
Theoretical Quantiles

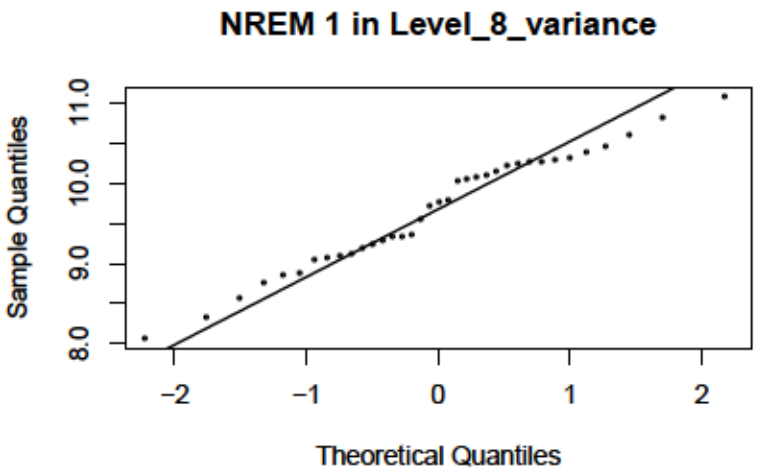

NREM 1 in Level_7_variance
Sample Quantiles
Theoretical Quantiles

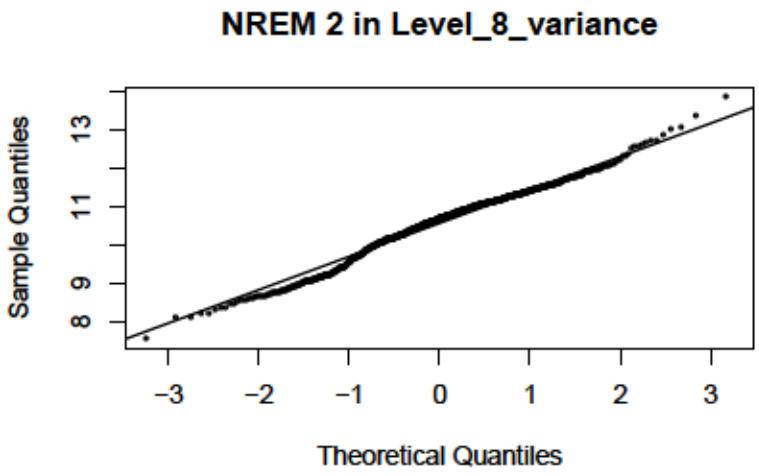

NREM 2 in Level_7_variance
Sample Quantiles
Theoretical Quantiles

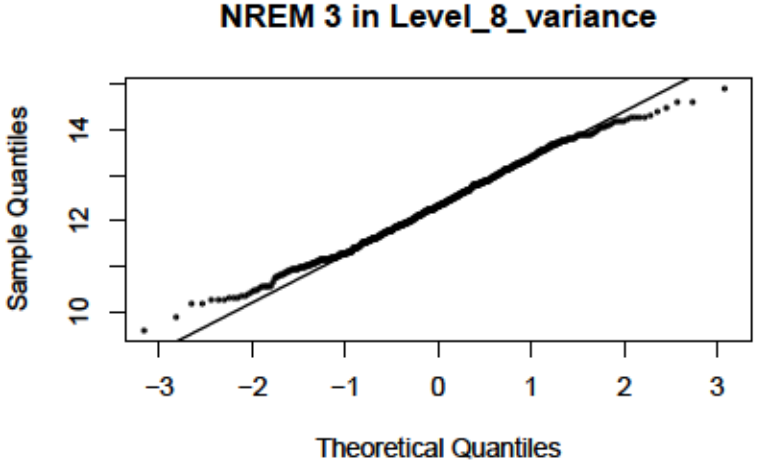

NREM 3 in Level_7_variance
Sample Quantiles
Theoretical Quantiles

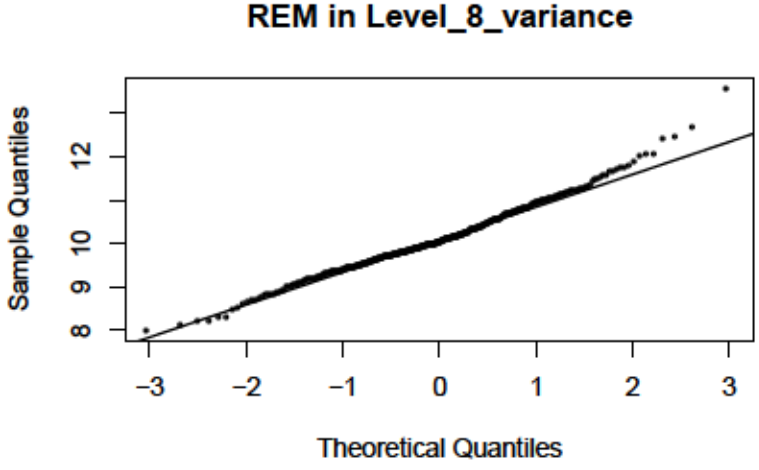

REM in Level_7_variance
Sample Quantiles
Theoretical Quantiles

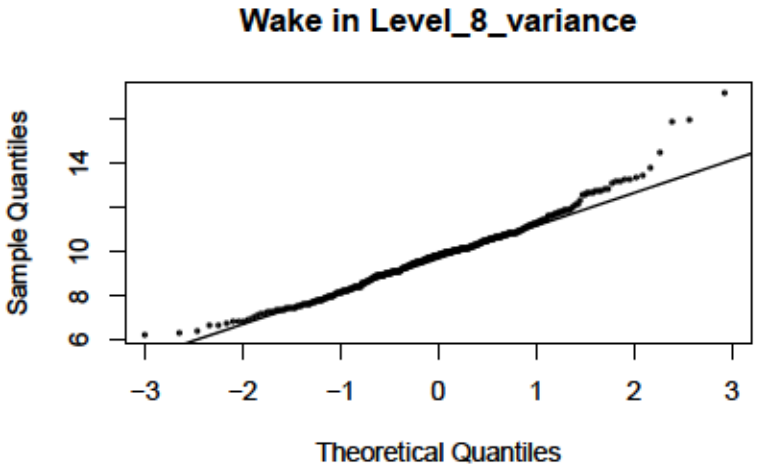

Wake in Level_7_variance
Sample Quantiles
Theoretical Quantiles

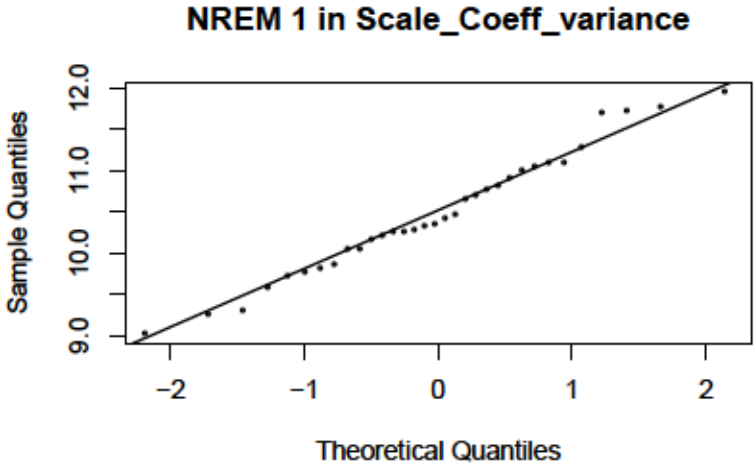

NREM 1 in Level_8_variance
Sample Quantiles
Theoretical Quantiles

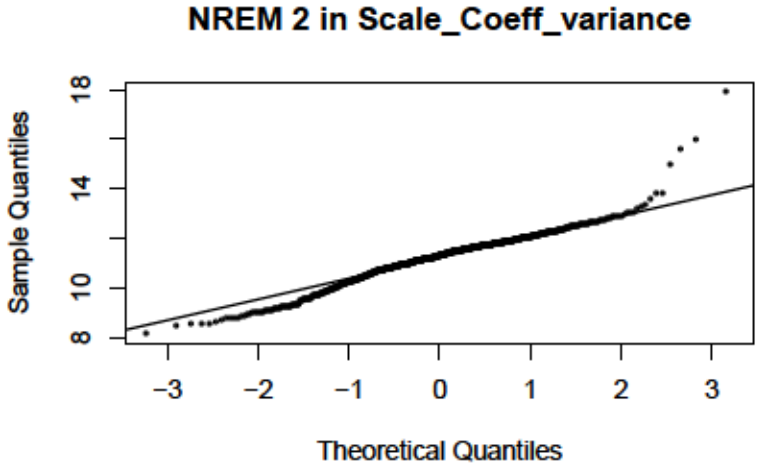

NREM 2 in Level_8_variance
Sample Quantiles
Theoretical Quantiles

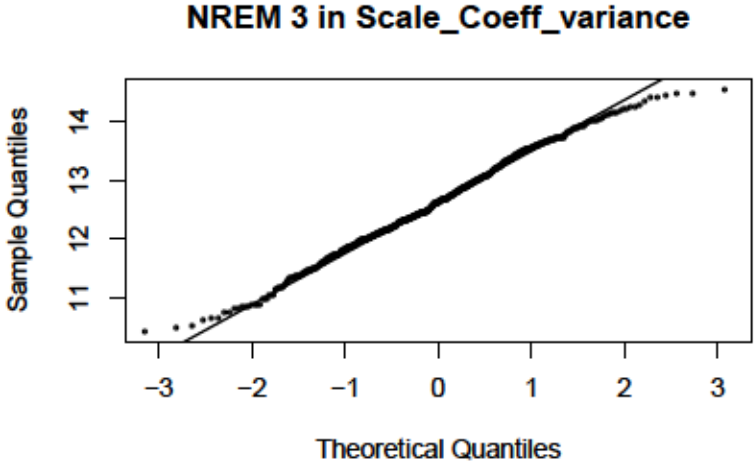

NREM 3 in Level_8_variance
Sample Quantiles
Theoretical Quantiles

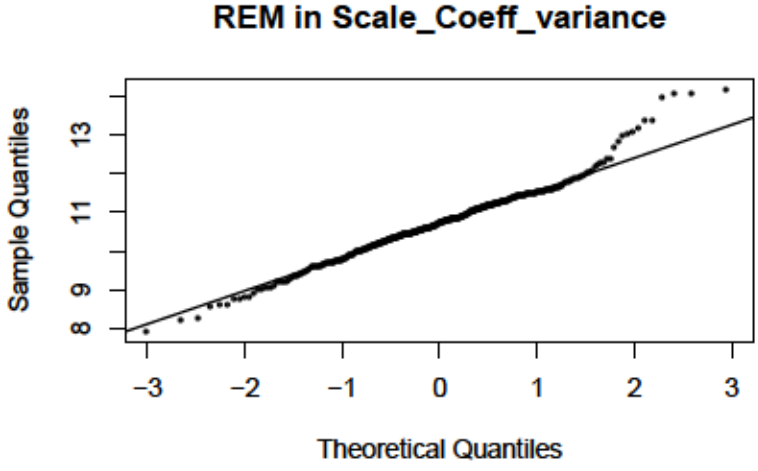

REM in Level_8_variance
Sample Quantiles
Theoretical Quantiles

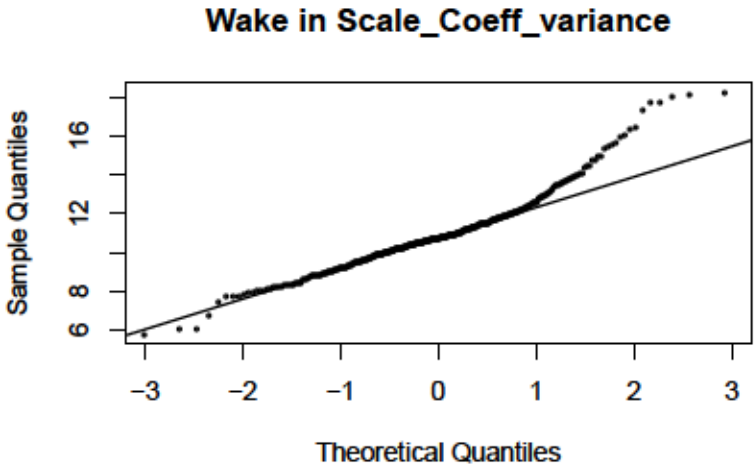

Wake in Level_8_variance
Sample Quantiles
Theoretical Quantiles

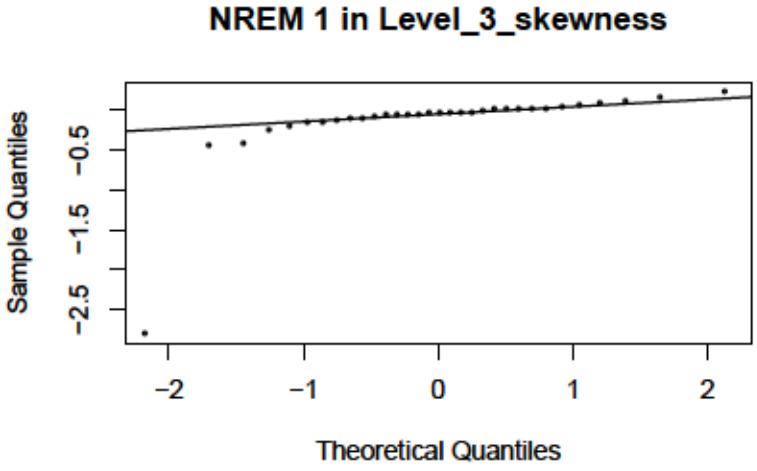

NREM 1 in Scale_Coeff_variance
Sample Quantiles
Theoretical Quantiles

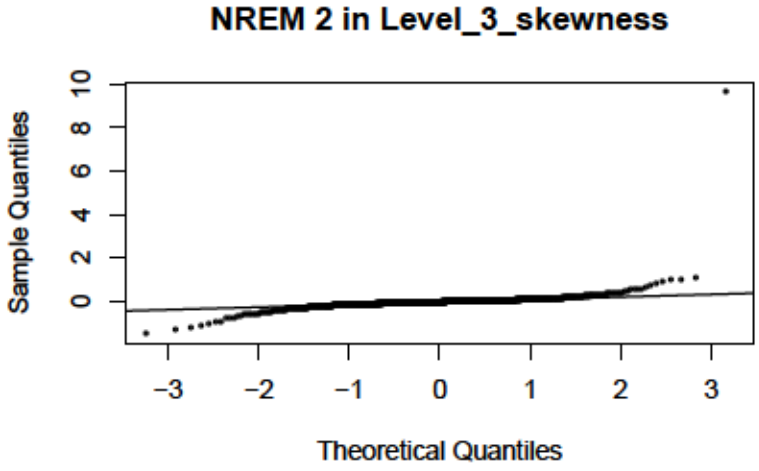

NREM 2 in Scale_Coeff_variance
Sample Quantiles
Theoretical Quantiles

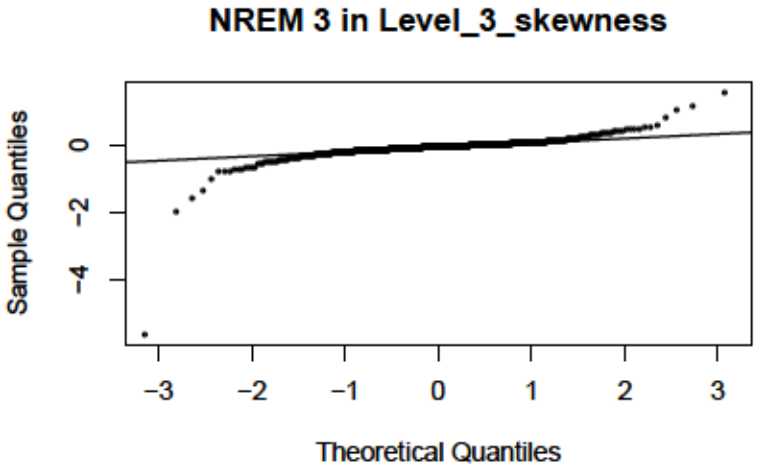

NREM 3 in Scale_Coeff_variance
Sample Quantiles
Theoretical Quantiles

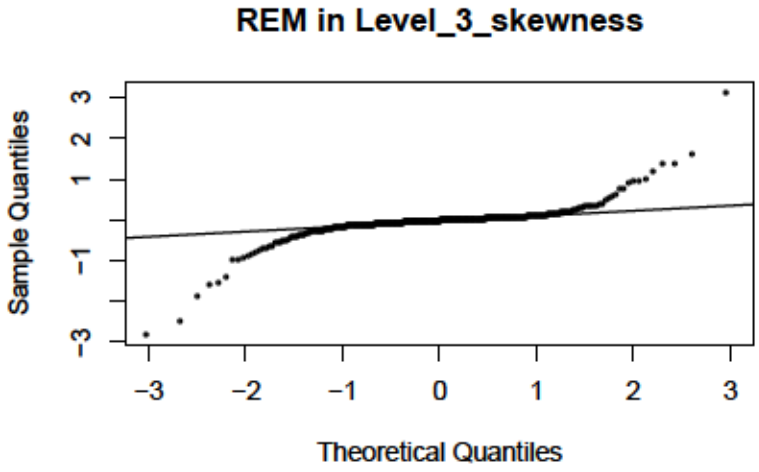

REM in Scale_Coeff_variance
Sample Quantiles
Theoretical Quantiles

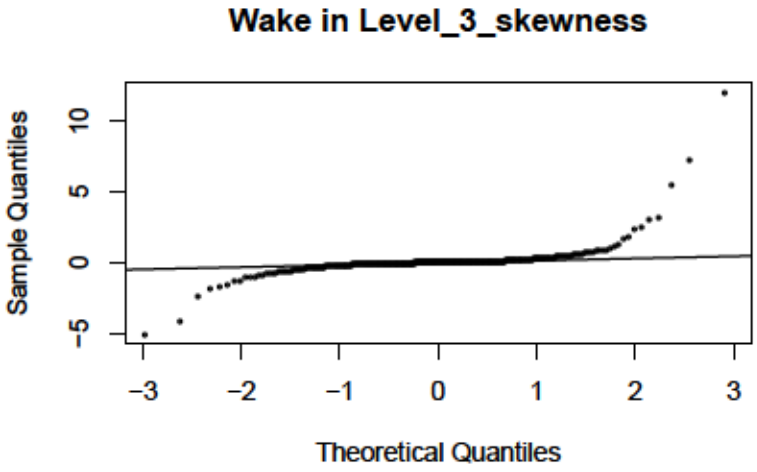

Wake in Scale_Coeff_variance
Sample Quantiles
Theoretical Quantiles

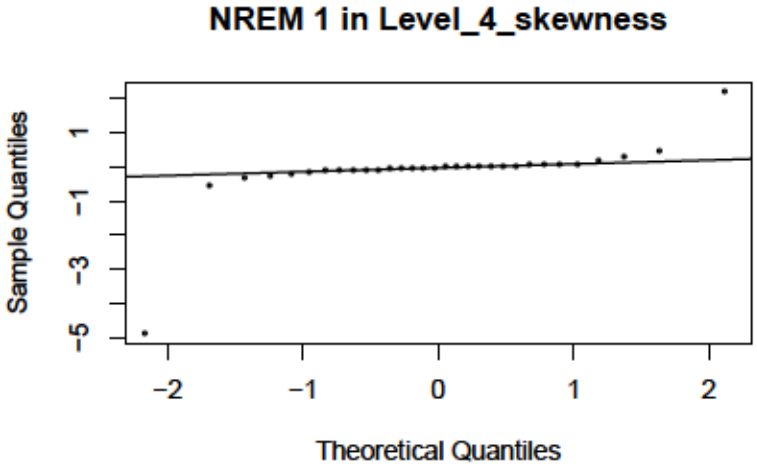

NREM 1 in Level_3_skewness
Sample Quantiles
Theoretical Quantiles

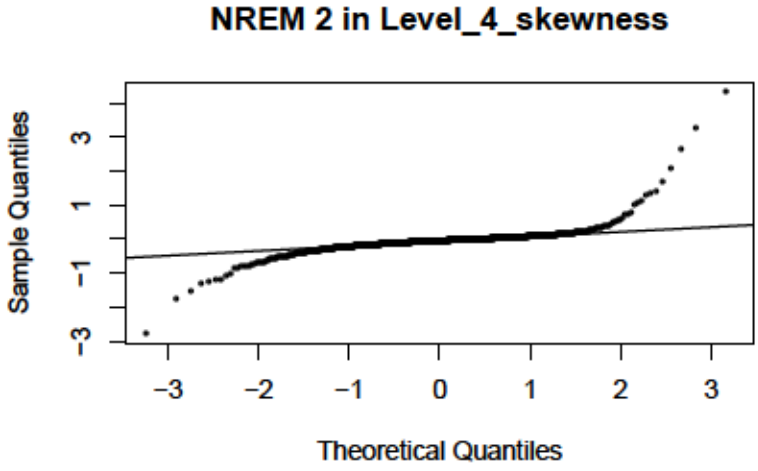

NREM 2 in Level_3_skewness
Sample Quantiles
Theoretical Quantiles

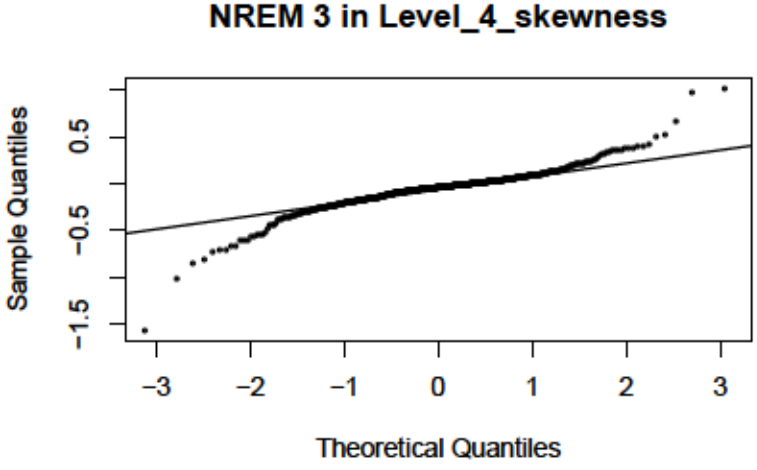

NREM 3 in Level_3_skewness
Sample Quantiles
Theoretical Quantiles

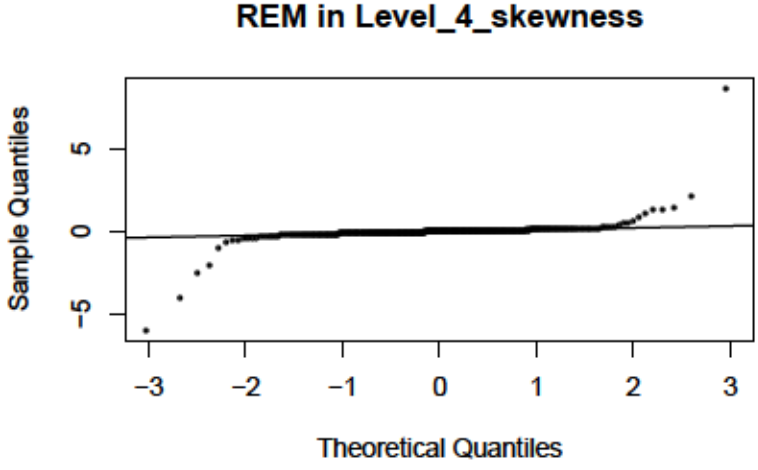

REM in Level_3_skewness
Sample Quantiles
Theoretical Quantiles

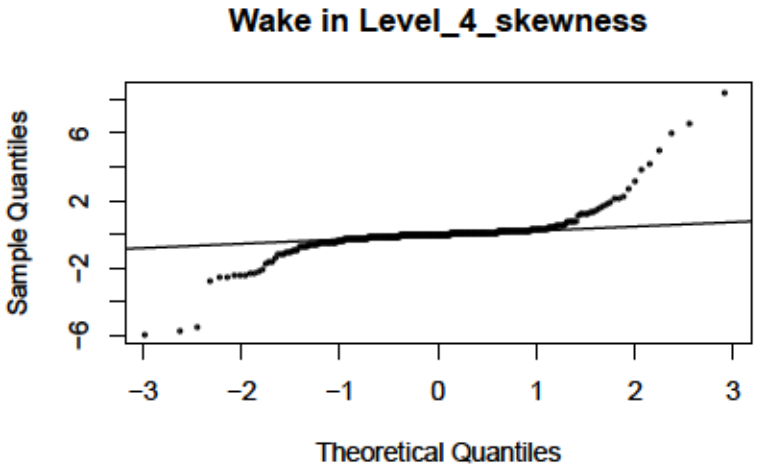

Wake in Level_3_skewness
Sample Quantiles
Theoretical Quantiles

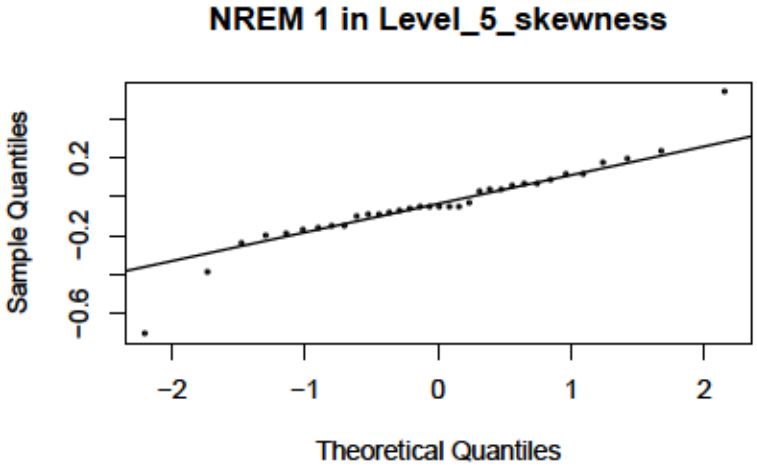

NREM 1 in Level_4_skewness
Sample Quantiles
Theoretical Quantiles

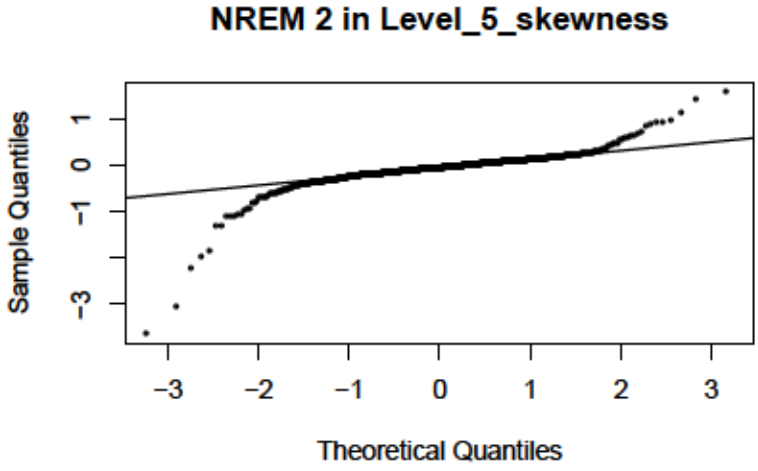

NREM 2 in Level_4_skewness
Sample Quantiles
Theoretical Quantiles

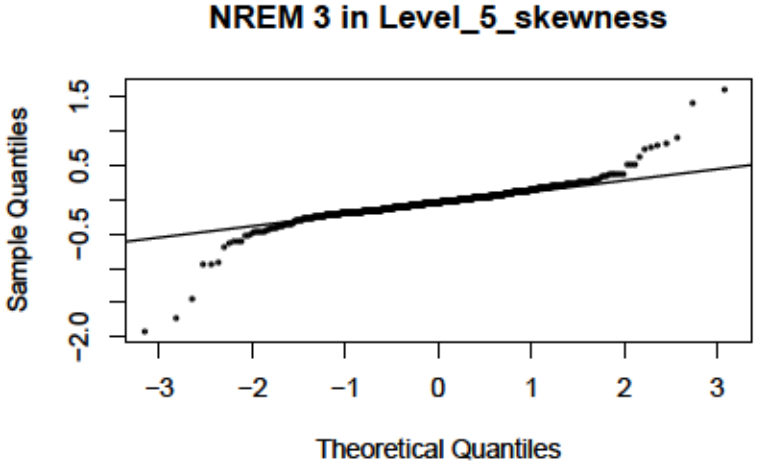

NREM 3 in Level_4_skewness
Sample Quantiles
Theoretical Quantiles

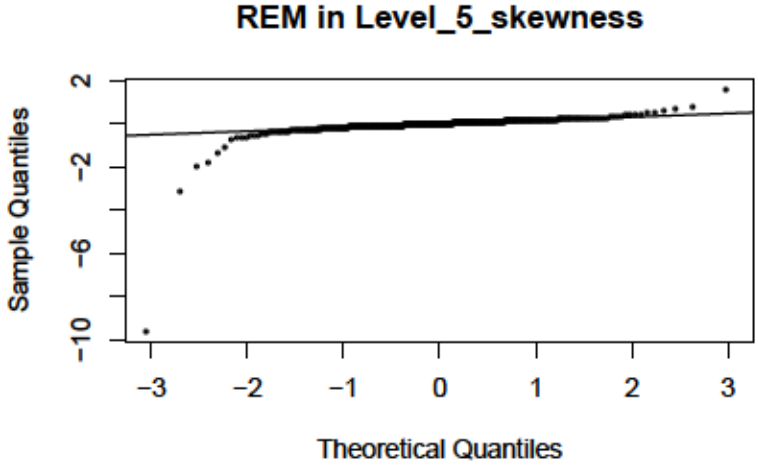

REM in Level_4_skewness
Sample Quantiles
Theoretical Quantiles

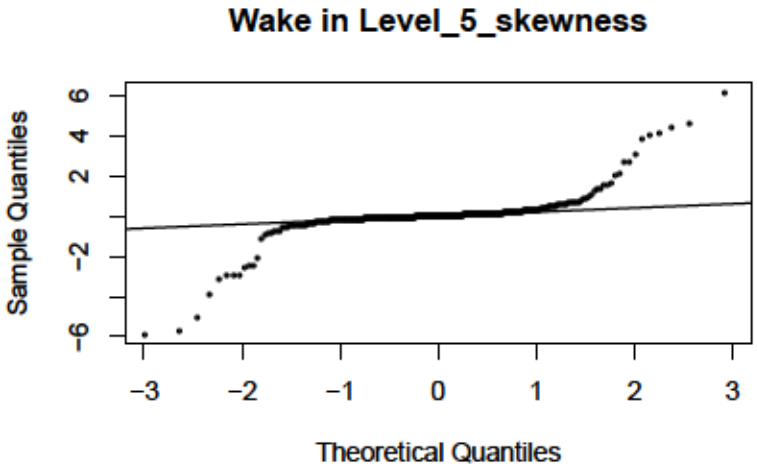

Wake in Level_4_skewness
Sample Quantiles
Theoretical Quantiles

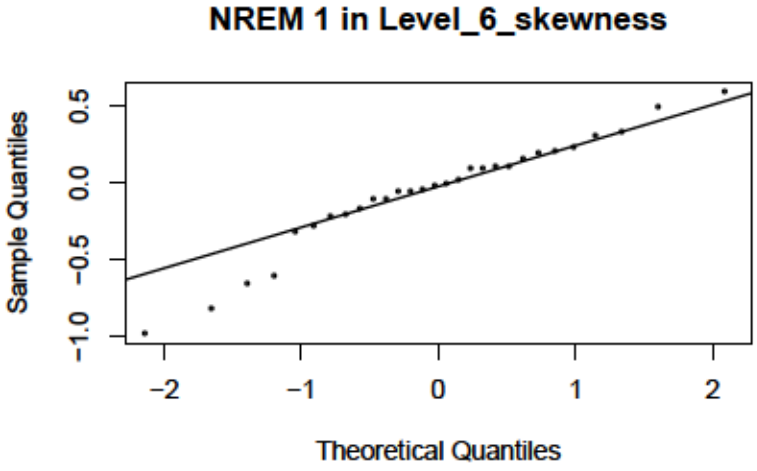

NREM 1 in Level_5_skewness
Sample Quantiles
Theoretical Quantiles

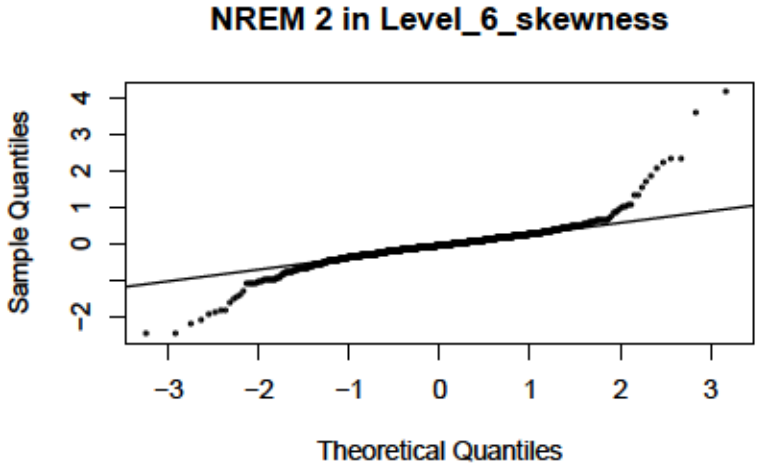

NREM 2 in Level_5_skewness
Sample Quantiles
Theoretical Quantiles

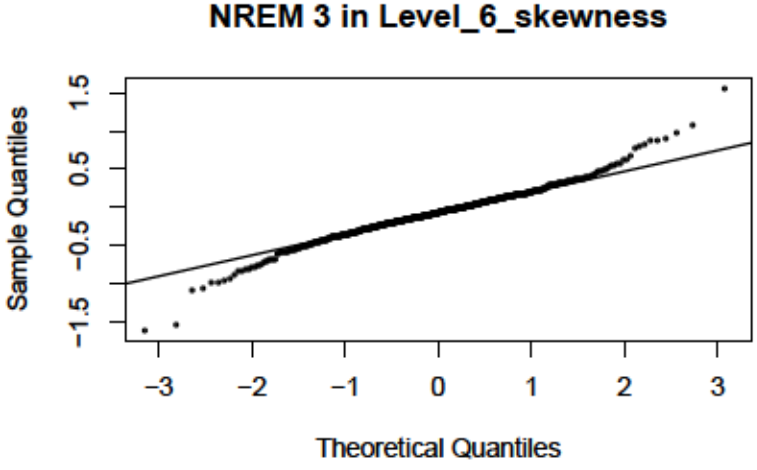

NREM 3 in Level_5_skewness
Sample Quantiles
Theoretical Quantiles

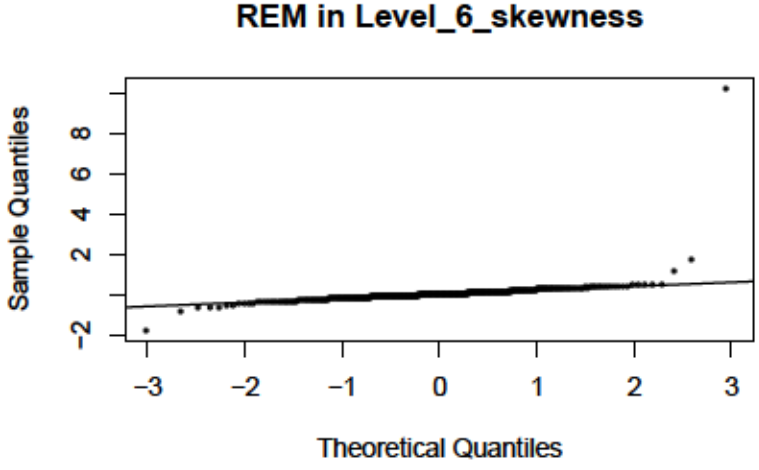

REM in Level_5_skewness
Sample Quantiles
Theoretical Quantiles

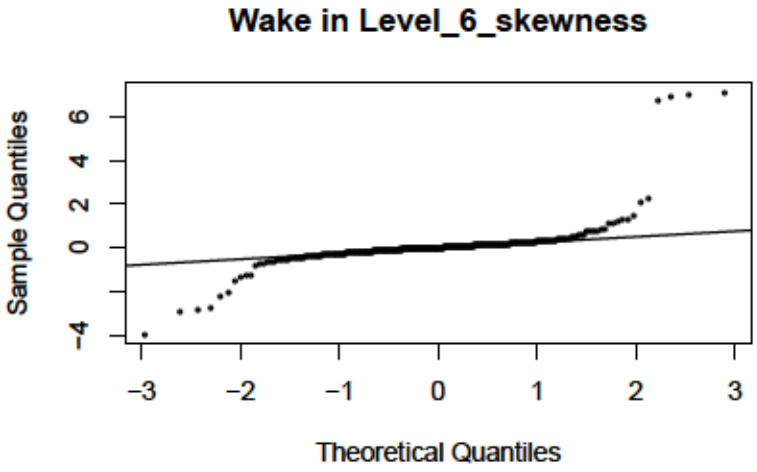

Wake in Level_5_skewness
Sample Quantiles
Theoretical Quantiles

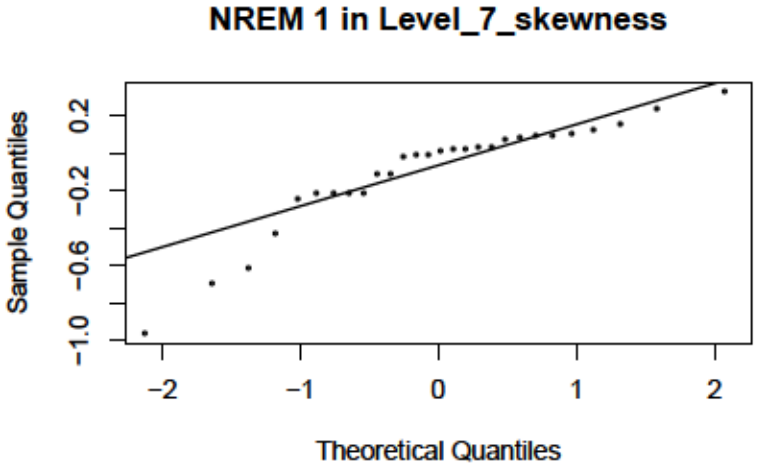

NREM 1 in Level_6_skewness
Sample Quantiles
Theoretical Quantiles

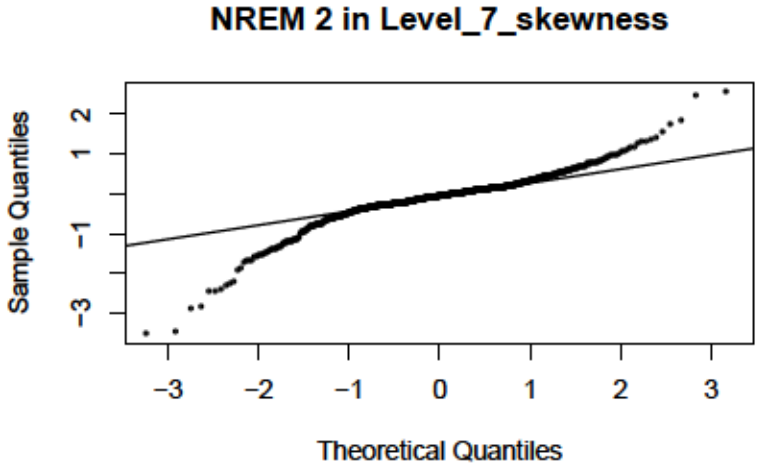

NREM 2 in Level_6_skewness
Sample Quantiles
Theoretical Quantiles

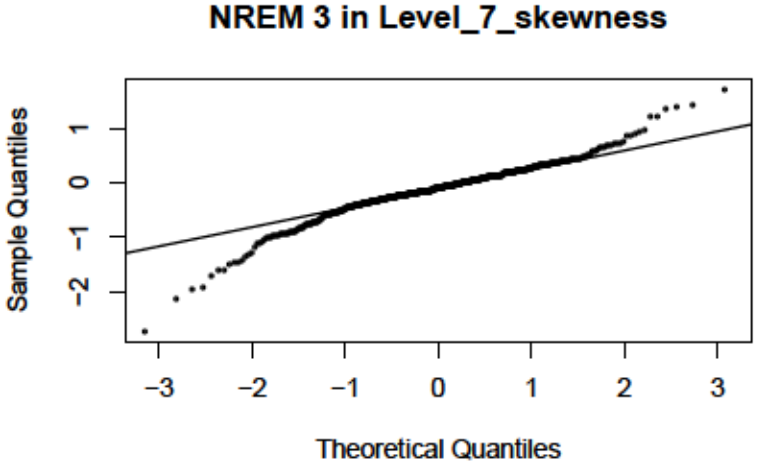

NREM 3 in Level_6_skewness
Sample Quantiles
Theoretical Quantiles

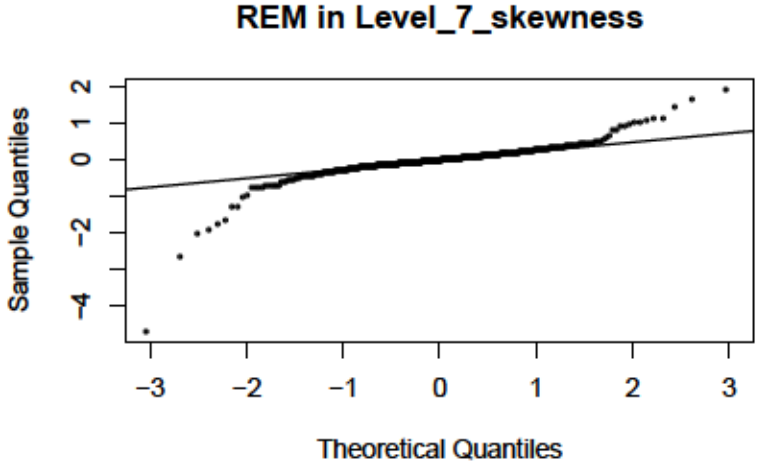

REM in Level_6_skewness
Sample Quantiles
Theoretical Quantiles

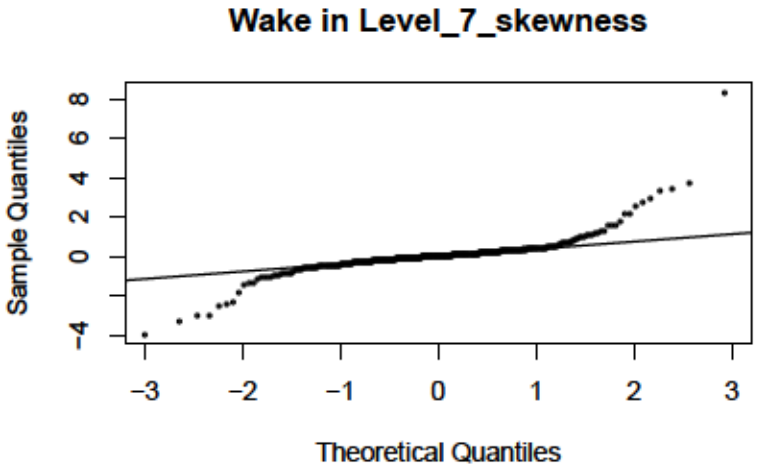

Wake in Level_6_skewness
Sample Quantiles
Theoretical Quantiles

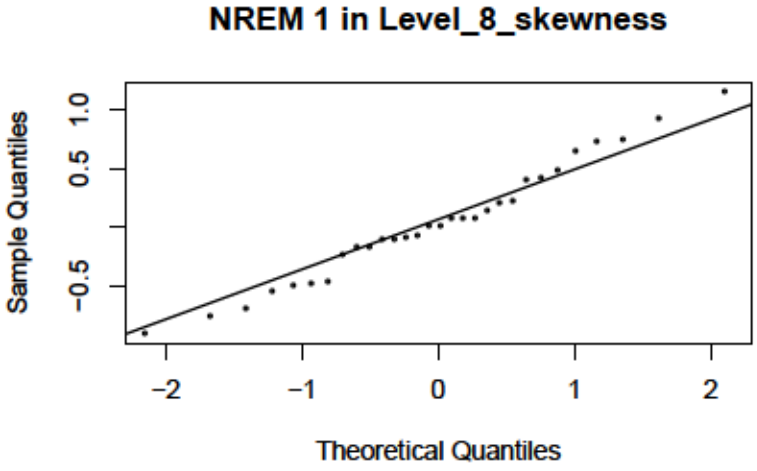

NREM 1 in Level_7_skewness
Sample Quantiles
Theoretical Quantiles

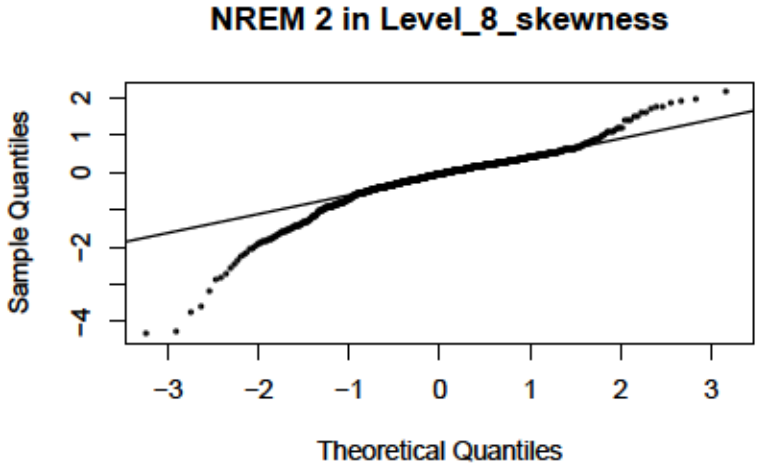

NREM 2 in Level_7_skewness
Sample Quantiles
Theoretical Quantiles

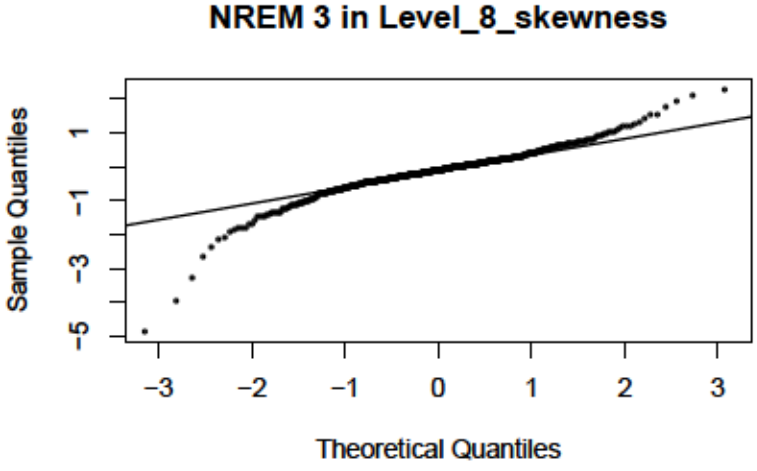

NREM 3 in Level_7_skewness
Sample Quantiles
Theoretical Quantiles

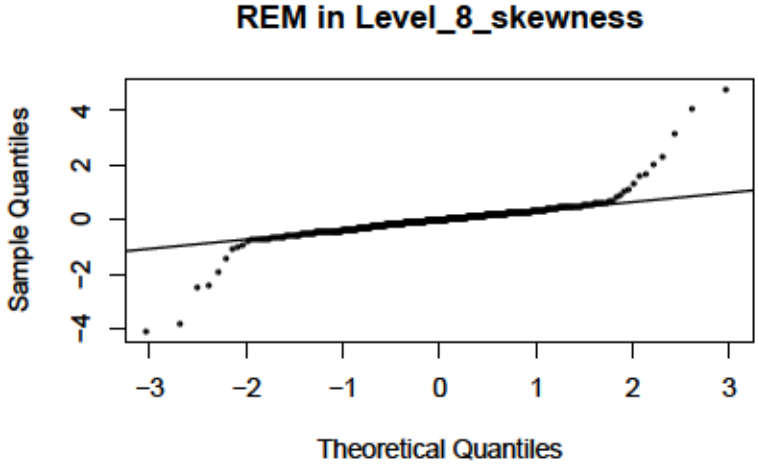

REM in Level_7_skewness
Sample Quantiles
Theoretical Quantiles

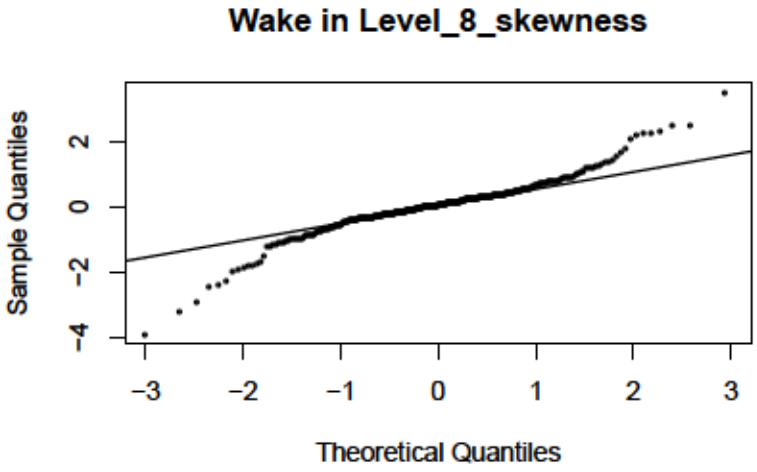

Wake in Level_7_skewness
Sample Quantiles
Theoretical Quantiles

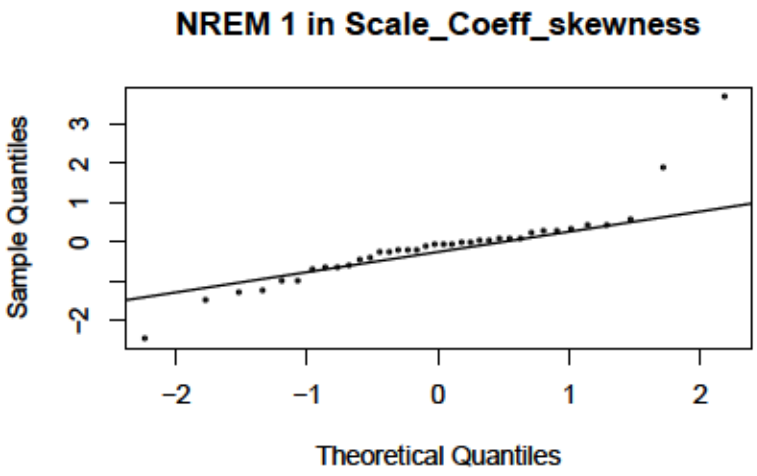

NREM 1 in Level_8_skewness
Sample Quantiles
Theoretical Quantiles

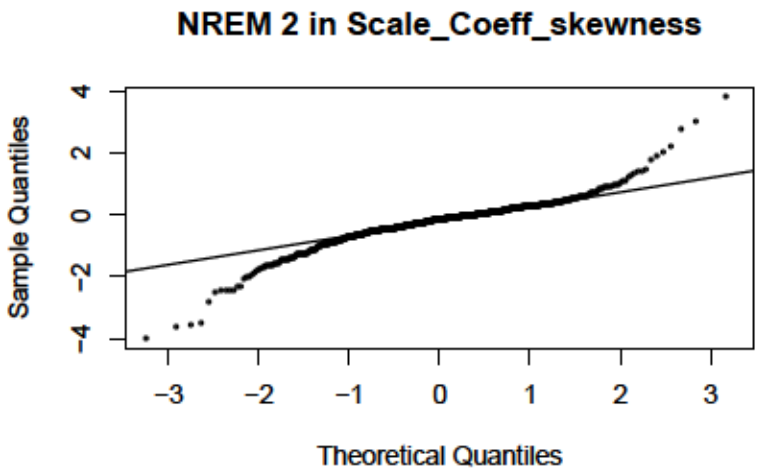

NREM 2 in Level_8_skewness
Sample Quantiles
Theoretical Quantiles

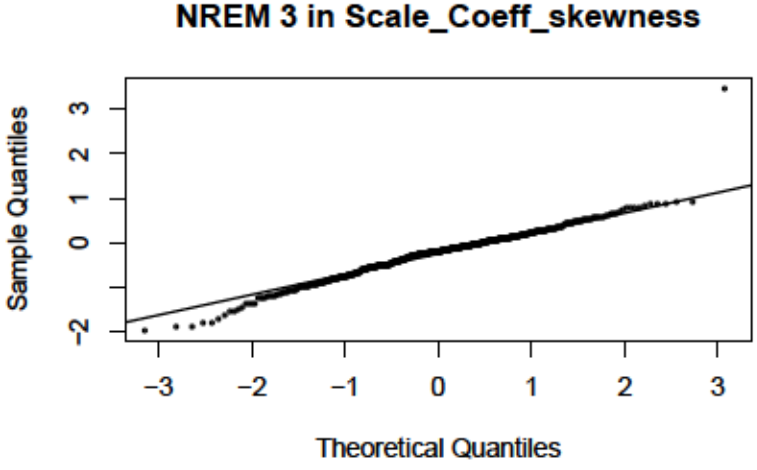

NREM 3 in Level_8_skewness
Sample Quantiles
Theoretical Quantiles

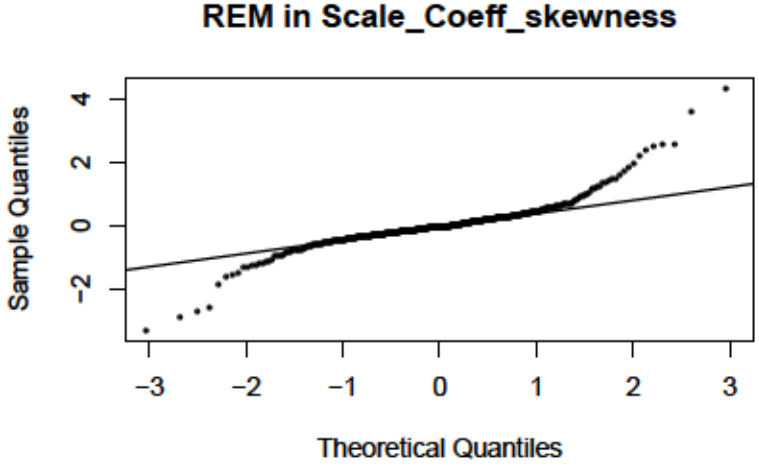

REM in Level_8_skewness
Sample Quantiles
Theoretical Quantiles

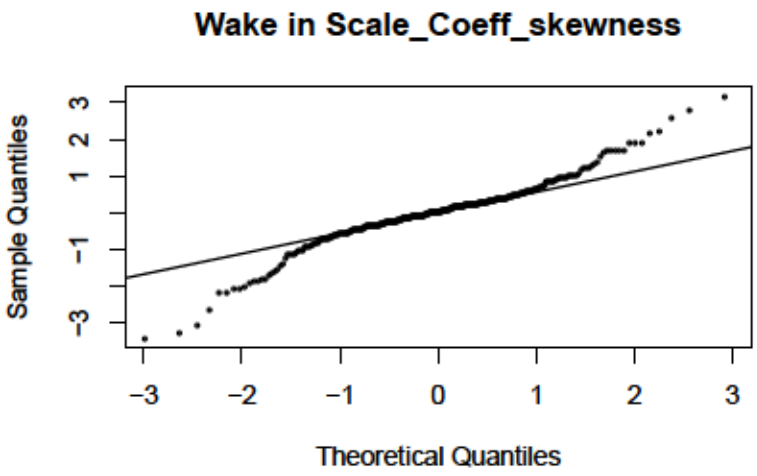

Wake in Level_8_skewness
Sample Quantiles
Theoretical Quantiles

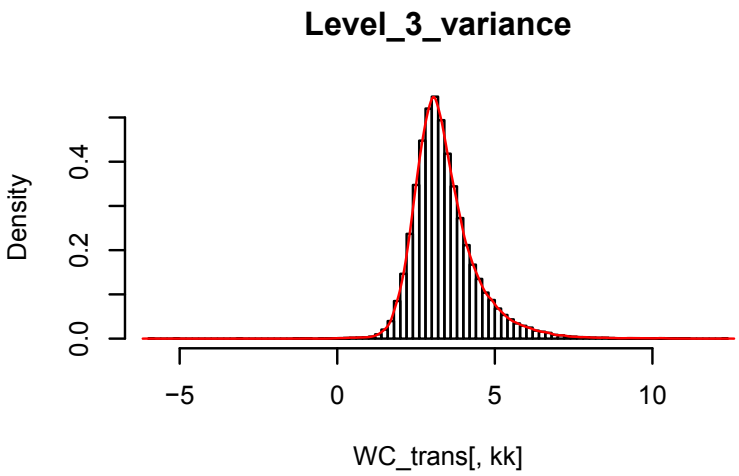

NREM 1 in Scale_Coeff_skewness
Sample Quantiles
Theoretical Quantiles

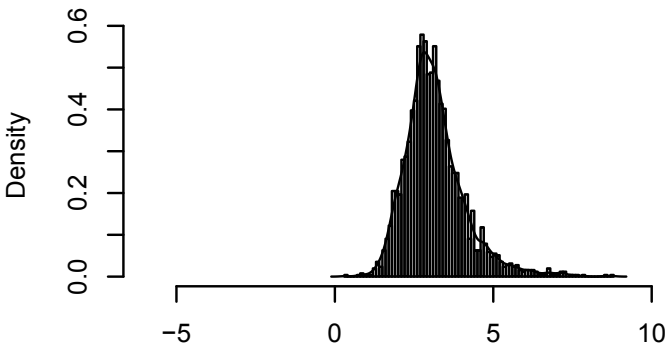

NREM 2 in Scale_Coeff_skewness
Sample Quantiles
Theoretical Quantiles

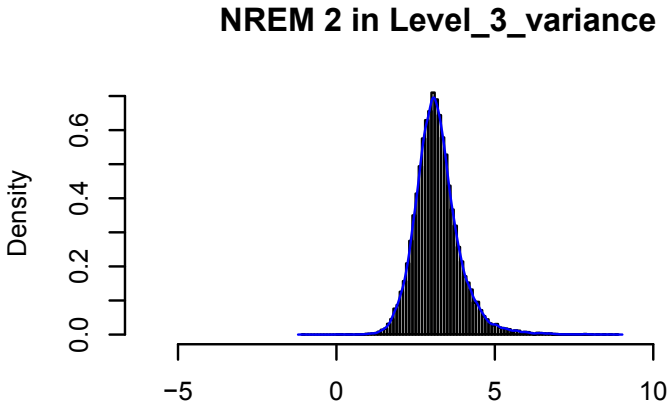

NREM 3 in Scale_Coeff_skewness
Sample Quantiles
Theoretical Quantiles

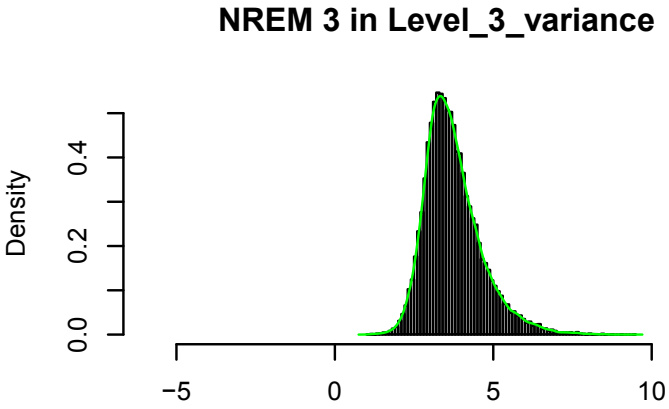

REM in Scale_Coeff_skewness
Sample Quantiles
Theoretical Quantiles

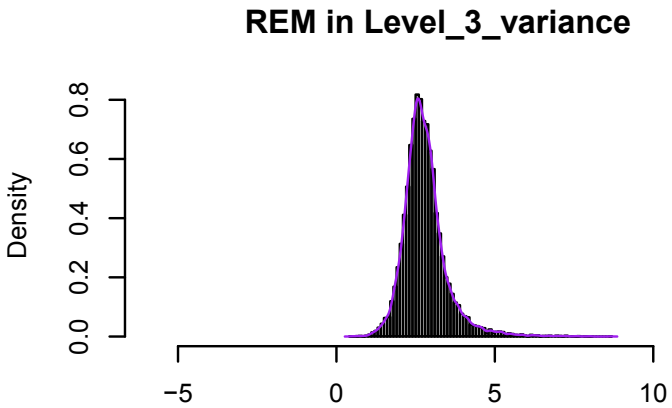

Wake in Scale_Coeff_skewness
Sample Quantiles
Theoretical Quantiles

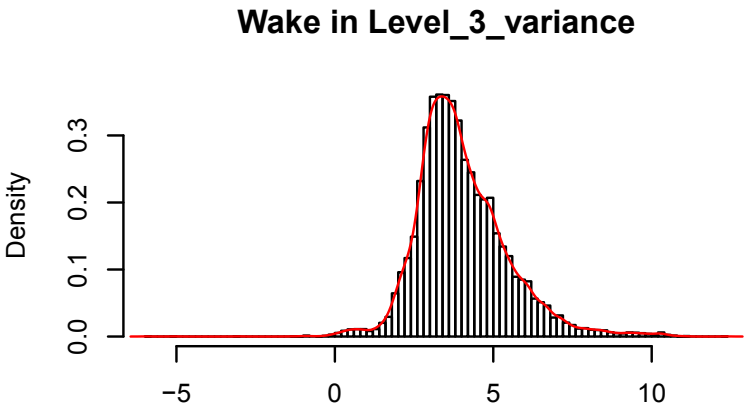

Level_3_variance
Density
0.0
0.2
0.4
−5
0
5
10
WC_trans[, kk]

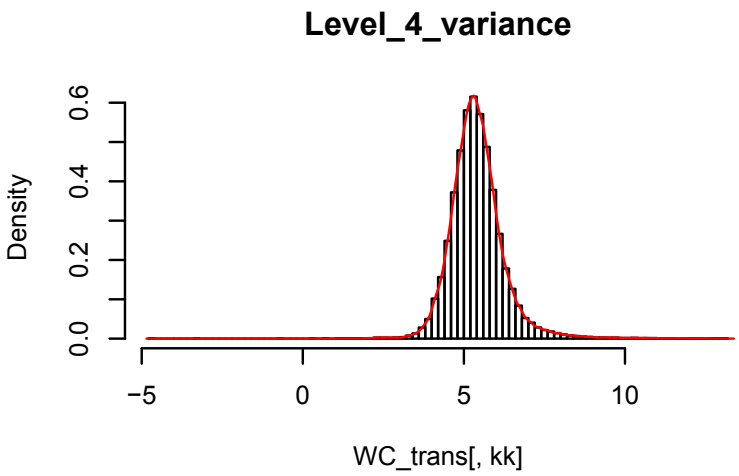

NREM 1 in Level_3_variance
Density
0.0
0.2
0.4
0.6
−5
0
5
10

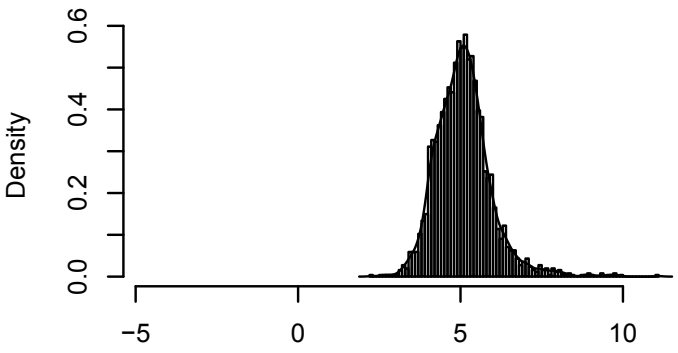

NREM 2 in Level_3_variance
Density
0.0
0.2
0.4
0.6
−5
0
5
10

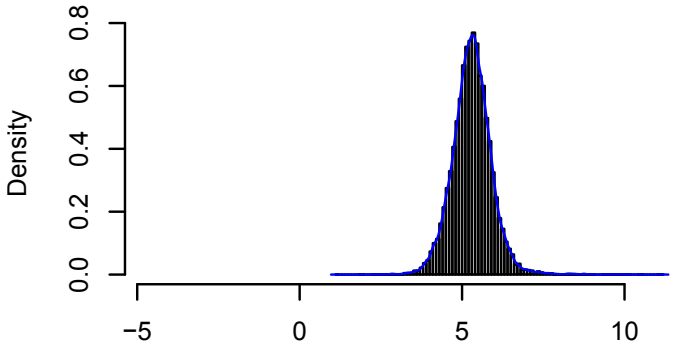

NREM 3 in Level_3_variance
Density
0.0
0.2
0.4
−5
0
5
10

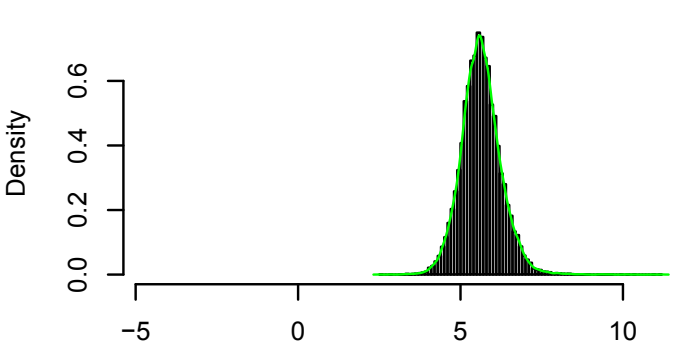

REM in Level_3_variance
Density
0.0
0.2
0.4
0.6
0.8
−5
0
5
10

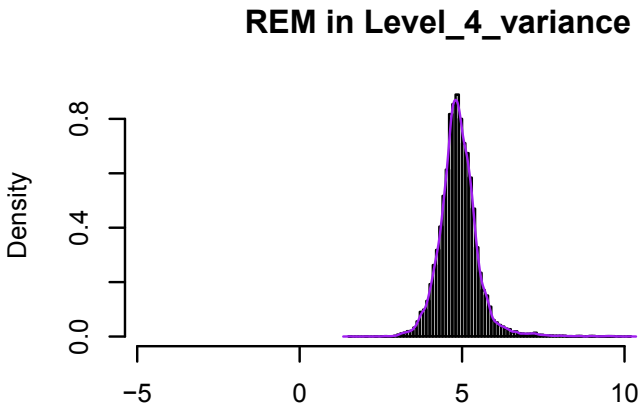

Wake in Level_3_variance
Density
0.0
0.1
0.2
0.3
−5
0
5
10

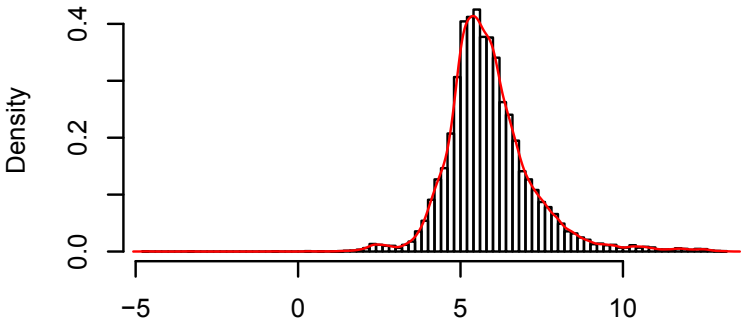

Level_4_variance
Density
0.0
0.2
0.4
0.6
−5
0
5
10
WC_trans[, kk]

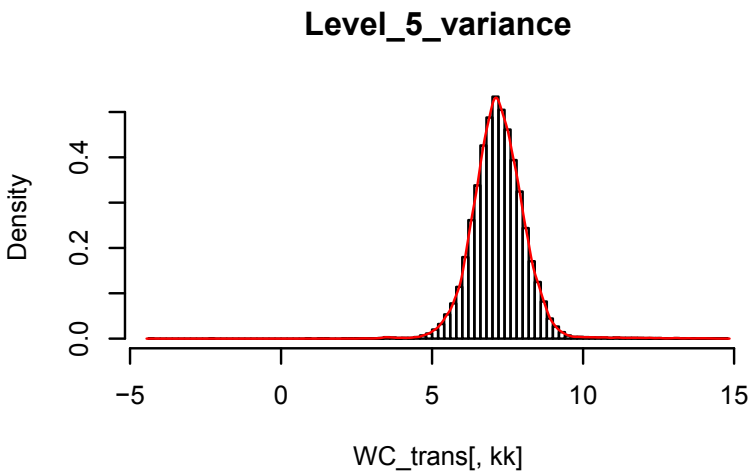

NREM 1 in Level_4_variance
Density
0.0
0.2
0.4
0.6
−5
0
5
10

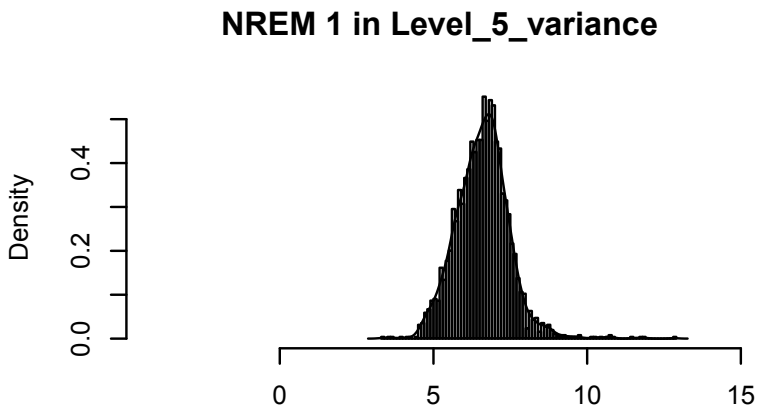

NREM 2 in Level_4_variance
Density
0.0
0.2
0.4
0.6
0.8
−5
0
5
10

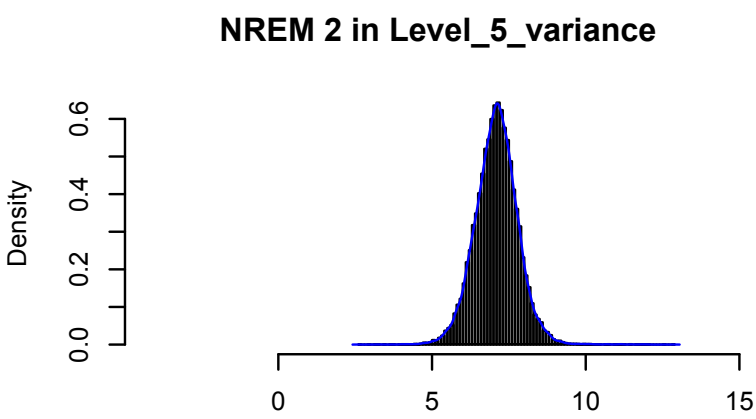

NREM 3 in Level_4_variance
Density
0.0
0.2
0.4
0.6
−5
0
5
10

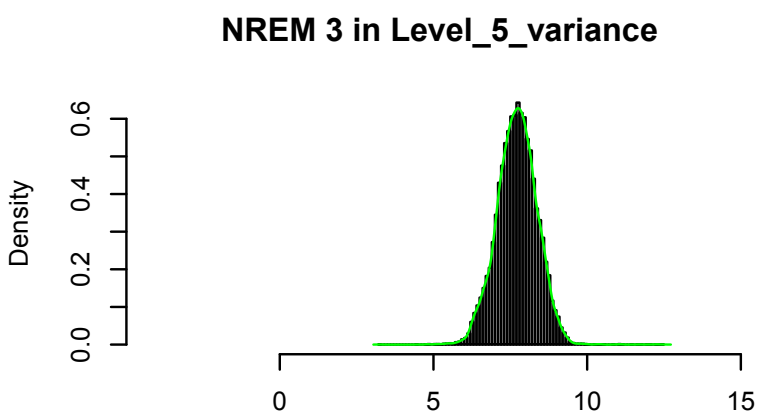

REM in Level_4_variance
Density
0.0
0.4
0.8
−5
0
5
10

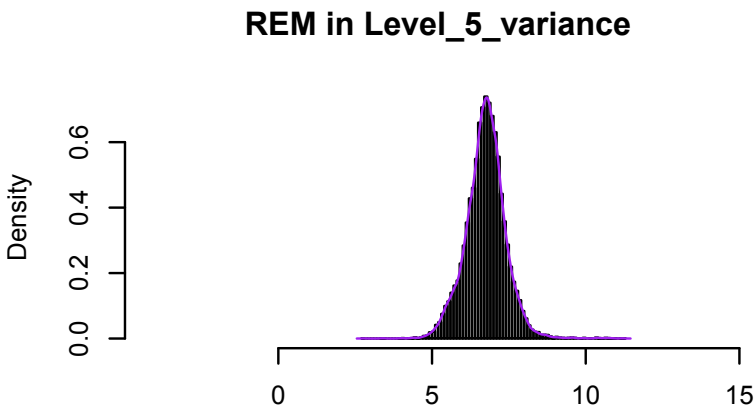

Wake in Level_4_variance
Density
0.0
0.2
0.4
−5
0
5
10

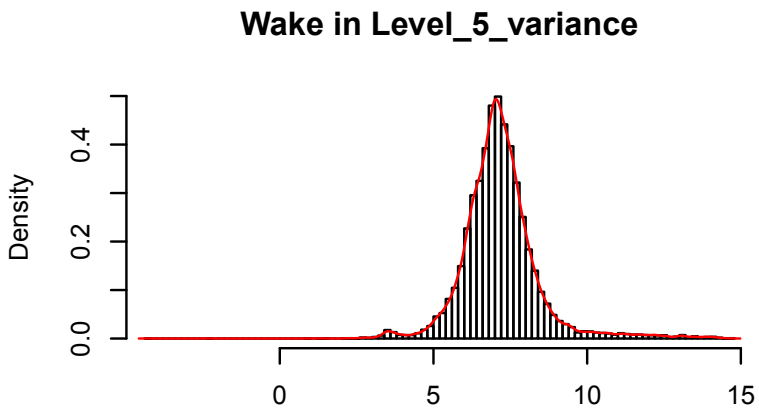

Level_5_variance
Density
0.0
0.2
0.4
−5
0
5
10
15
WC_trans[, kk]

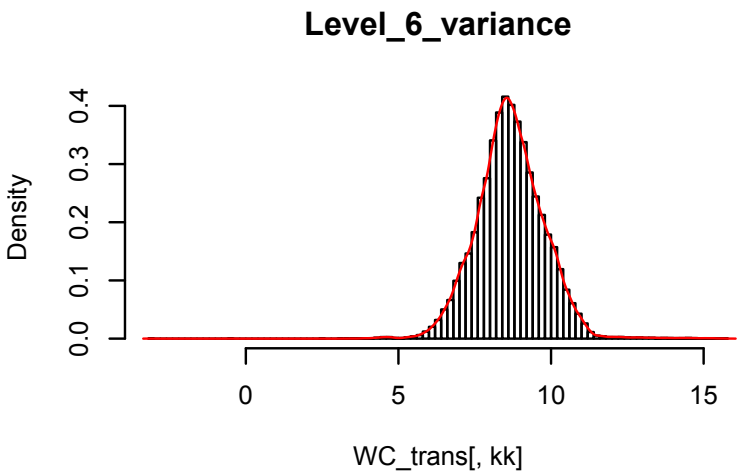

NREM 1 in Level_5_variance
Density
0.0
0.2
0.4
0
5
10
15

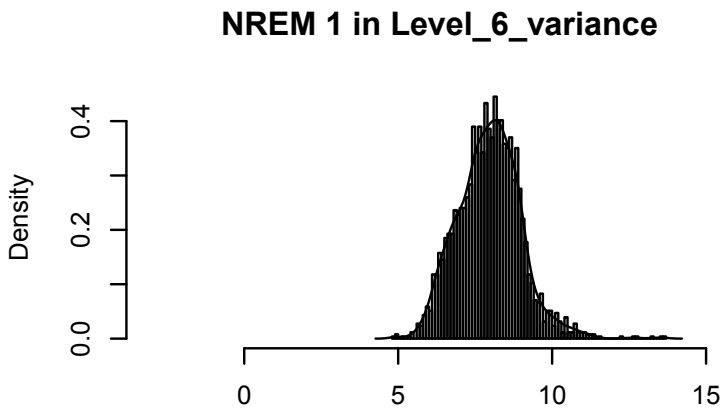

NREM 2 in Level_5_variance
Density
0.0
0.2
0.4
0.6
0
5
10
15

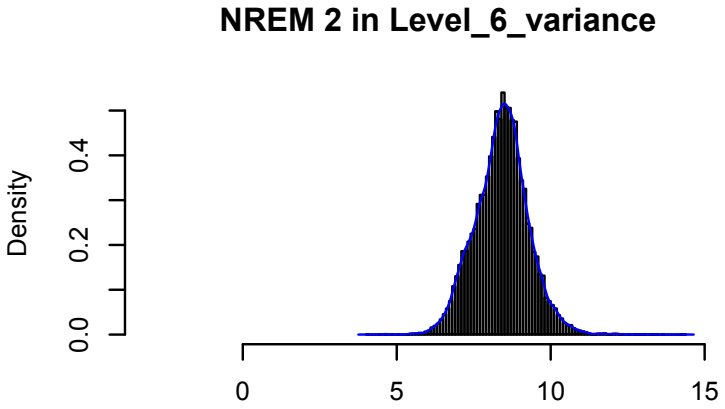

NREM 3 in Level_5_variance
Density
0.0
0.2
0.4
0.6
0
5
10
15

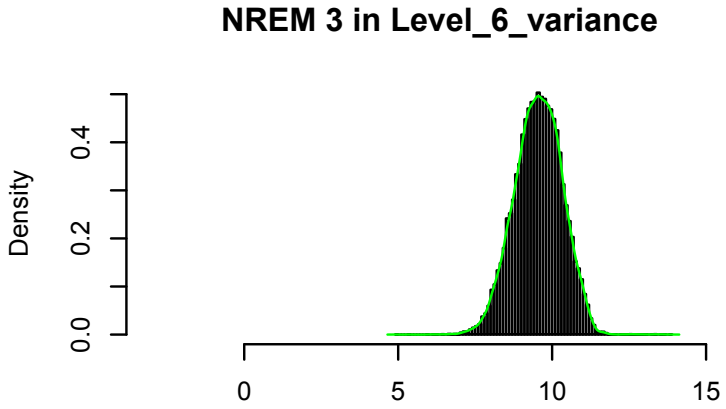

REM in Level_5_variance
Density
0.0
0.2
0.4
0.6
0
5
10
15

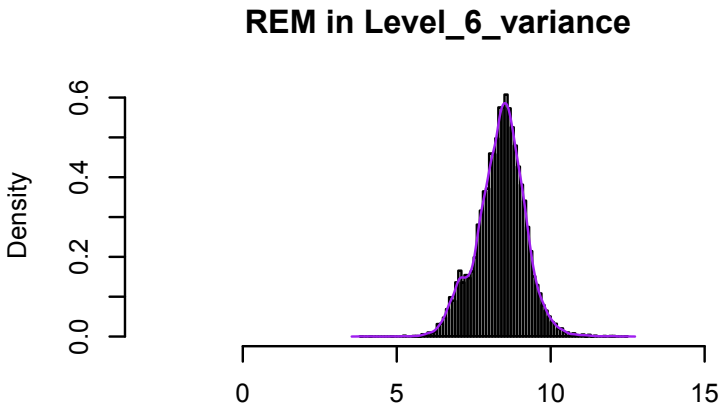

Wake in Level_5_variance
Density
0.0
0.2
0.4
0
5
10
15

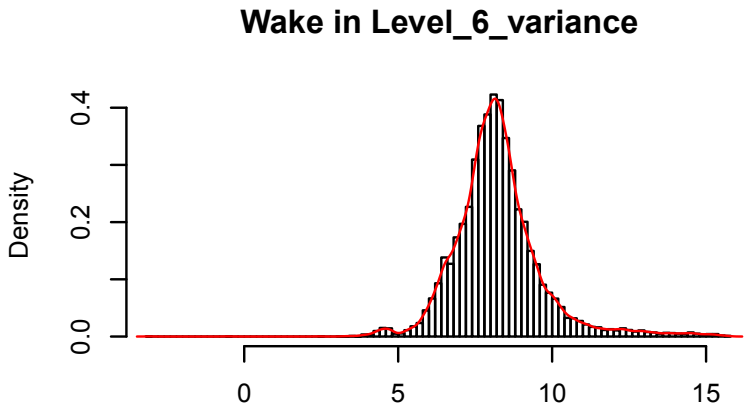

Level_6_variance
Density
0.0
0.1
0.2
0.3
0.4
0
5
10
15
WC_trans[, kk]

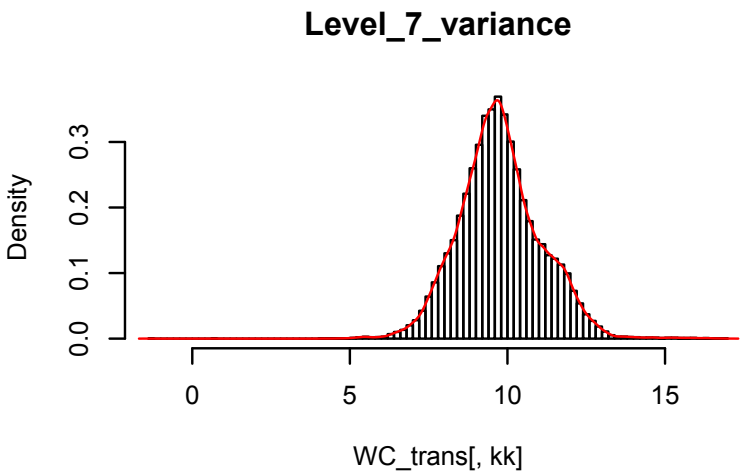

NREM 1 in Level_6_variance
Density
0.0
0.2
0.4
0
5
10
15

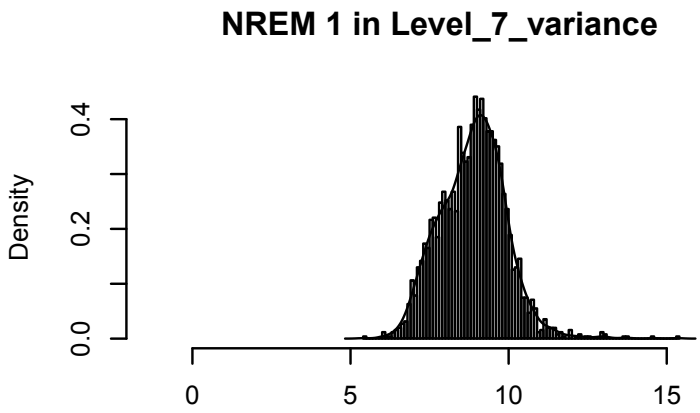

NREM 2 in Level_6_variance
Density
0.0
0.2
0.4
0
5
10
15

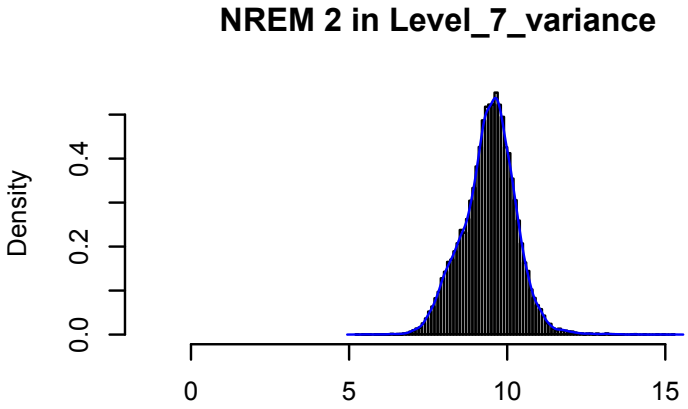

NREM 3 in Level_6_variance
Density
0.0
0.2
0.4
0
5
10
15

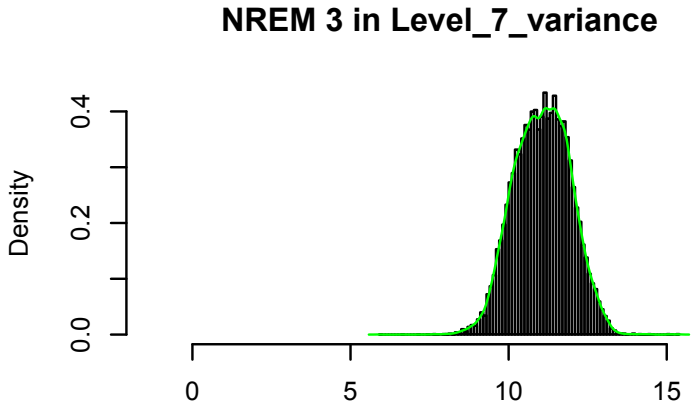

REM in Level_6_variance
Density
0.0
0.2
0.4
0.6
0
5
10
15

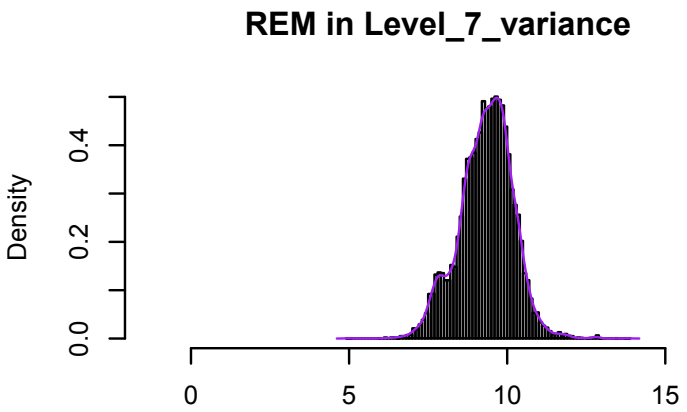

Wake in Level_6_variance
Density
0.0
0.2
0.4
0
5
10
15

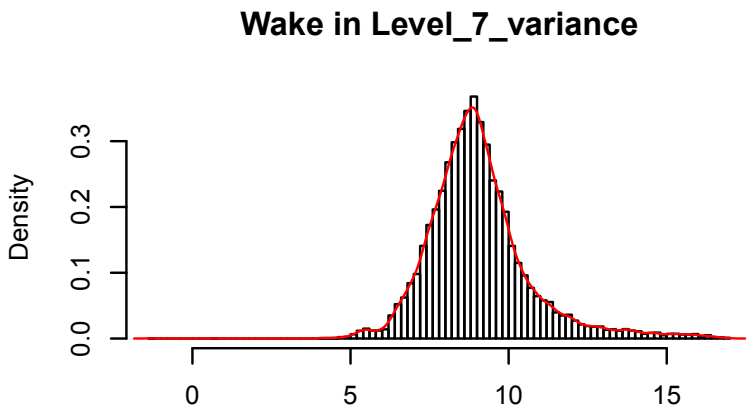

Level_7_variance
Density
0.0
0.1
0.2
0.3
0
5
10
15
WC_trans[, kk]

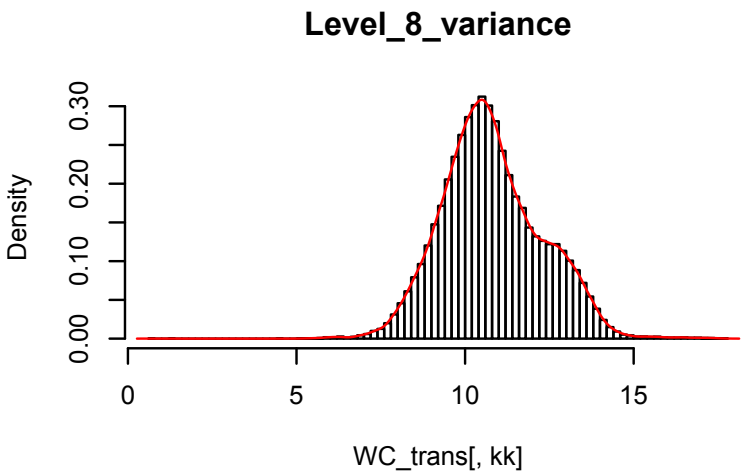

NREM 1 in Level_7_variance
Density
0.0
0.2
0.4
0
5
10
15

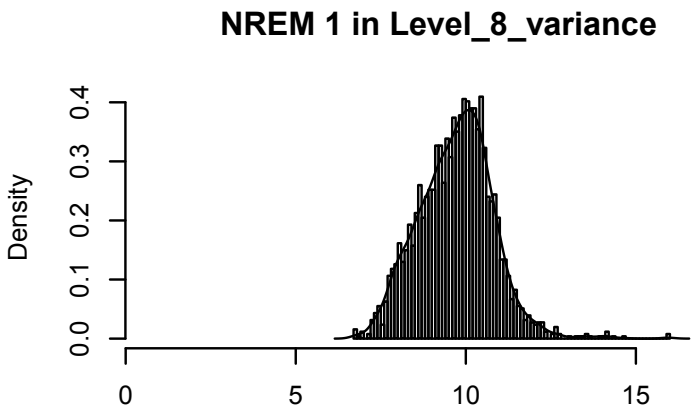

NREM 2 in Level_7_variance
Density
0.0
0.2
0.4
0
5
10
15

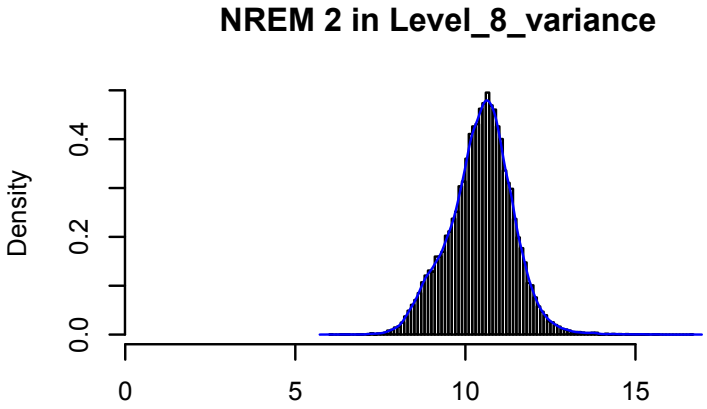

NREM 3 in Level_7_variance
Density
0.0
0.2
0.4
0
5
10
15

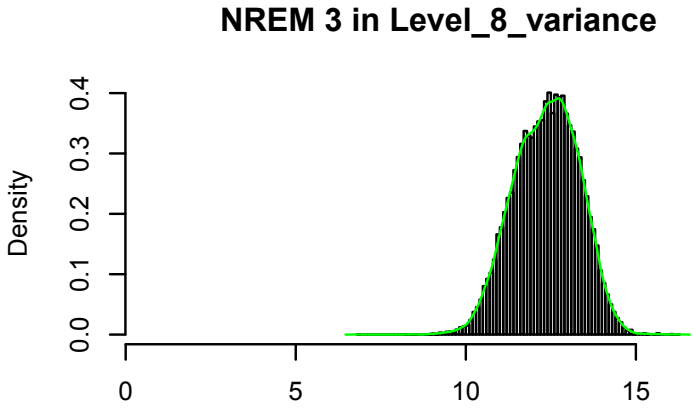

REM in Level_7_variance
Density
0.0
0.2
0.4
0
5
10
15

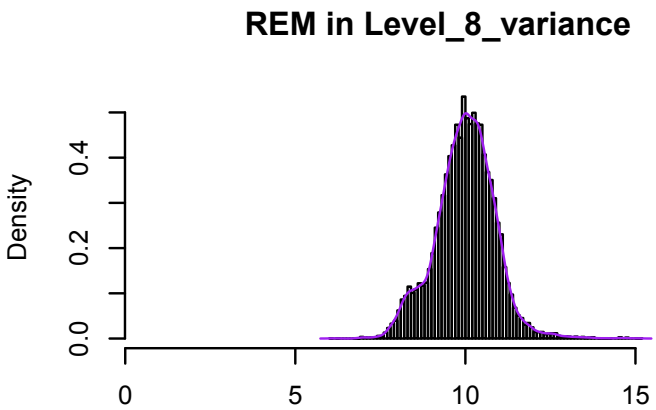

Wake in Level_7_variance
Density
0.0
0.1
0.2
0.3
0
5
10
15

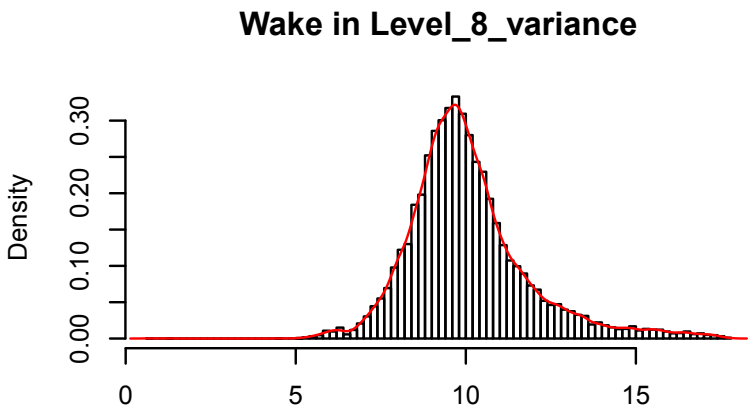

Level_8_variance
Density
0.00
0.10
0.20
0.30
0
5
10
15
WC_trans[, kk]

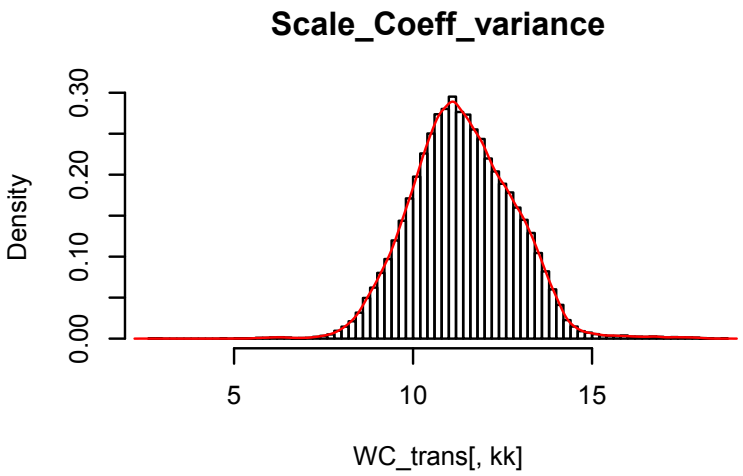

NREM 1 in Level_8_variance
Density
0.0
0.1
0.2
0.3
0.4
0
5
10
15

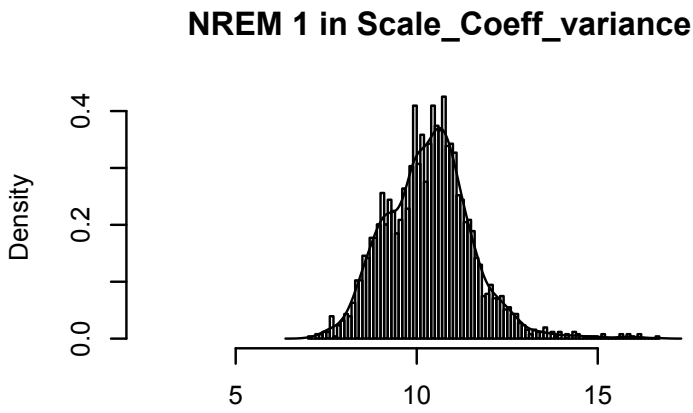

NREM 2 in Level_8_variance
Density
0.0
0.2
0.4
0
5
10
15

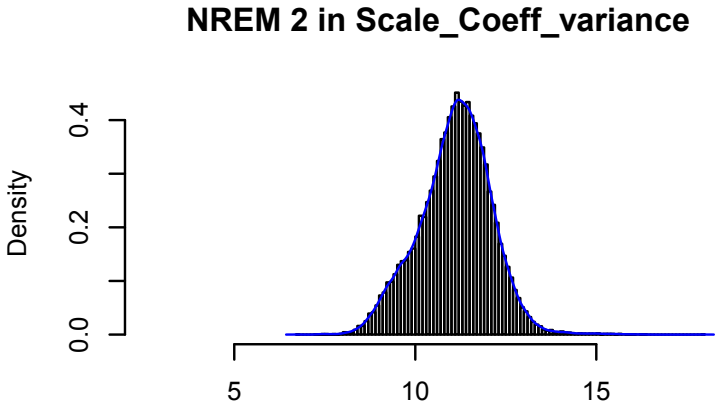

NREM 3 in Level_8_variance
Density
0.0
0.1
0.2
0.3
0.4
0
5
10
15

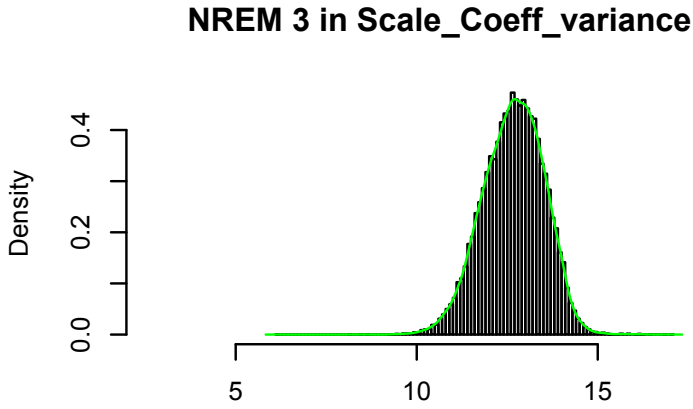

REM in Level_8_variance
Density
0.0
0.2
0.4
0
5
10
15

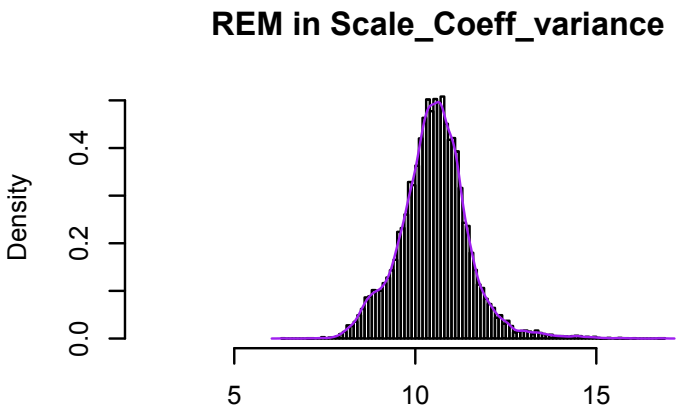

Wake in Level_8_variance
Density
0.00
0.10
0.20
0.30
0
5
10
15

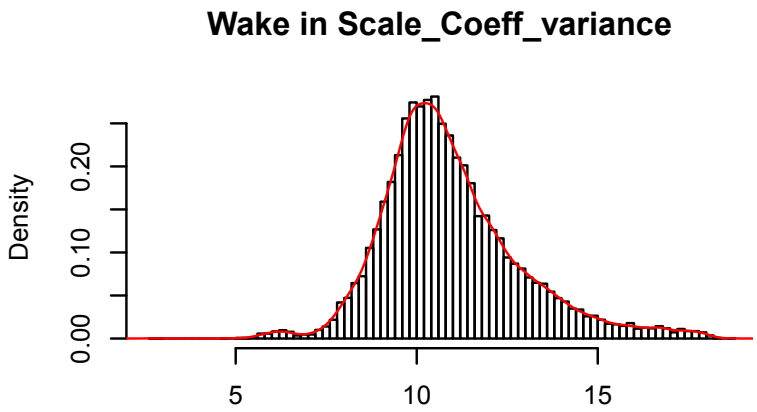

Scale_Coeff_variance
Density
0.00
0.10
0.20
0.30
5
10
15
WC_trans[, kk]

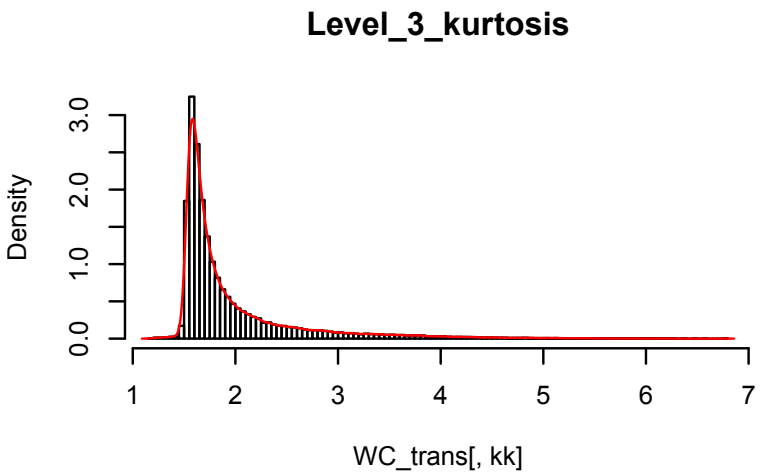

NREM 1 in Scale_Coeff_variance
Density
0.0
0.2
0.4
5
10
15

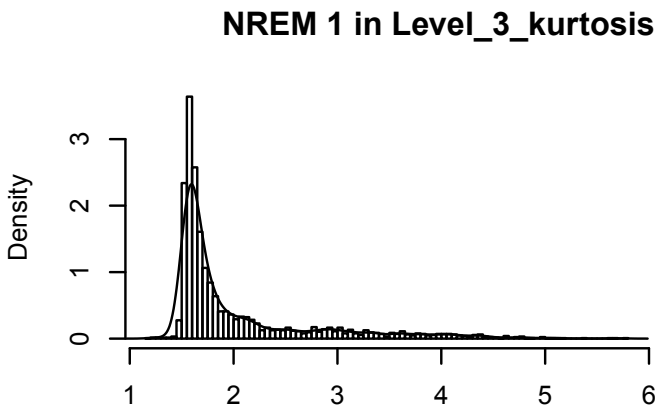

NREM 2 in Scale_Coeff_variance
Density
0.0
0.2
0.4
5
10
15

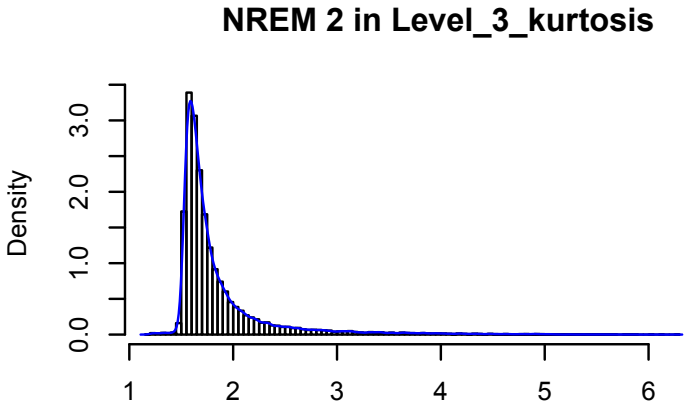

NREM 3 in Scale_Coeff_variance
Density
0.0
0.2
0.4
5
10
15

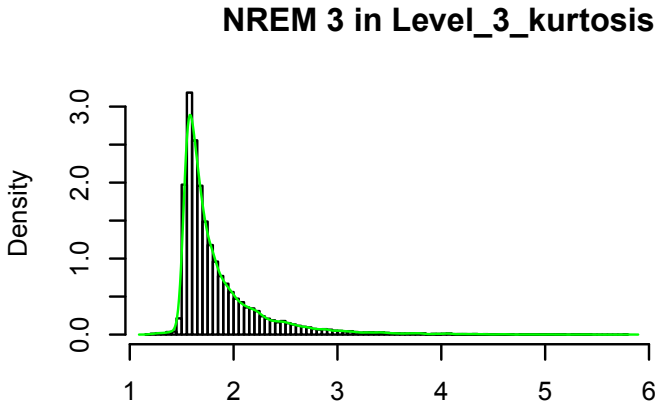

REM in Scale_Coeff_variance
Density
0.0
0.2
0.4
5
10
15

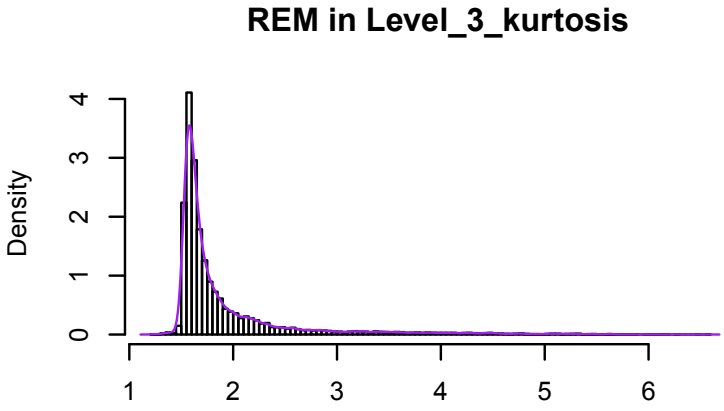

Wake in Scale_Coeff_variance
Density
0.00
0.10
0.20
5
10
15

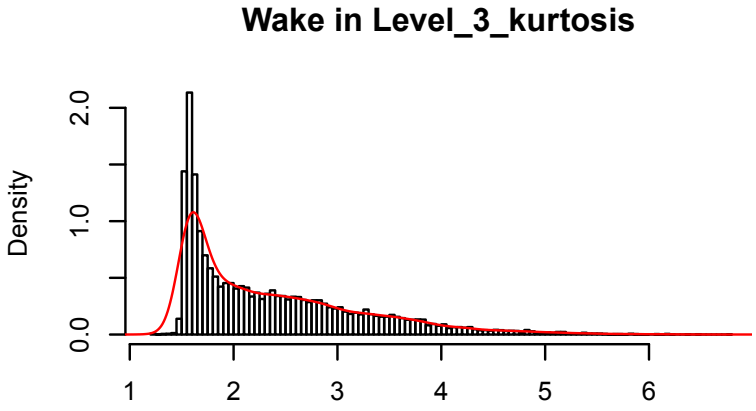

Level_3_kurtosis
Density
0.0
1.0
2.0
3.0
1
2
3
4
5
6
7
WC_trans[, kk]

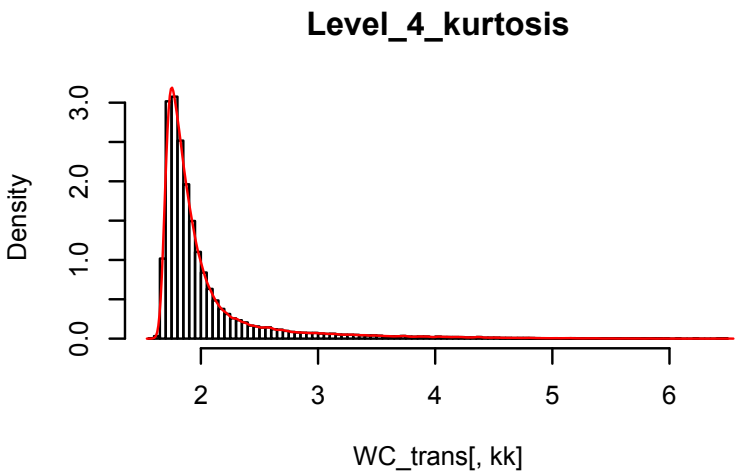

NREM 1 in Level_3_kurtosis
Density
0
1
2
3
1
2
3
4
5
6

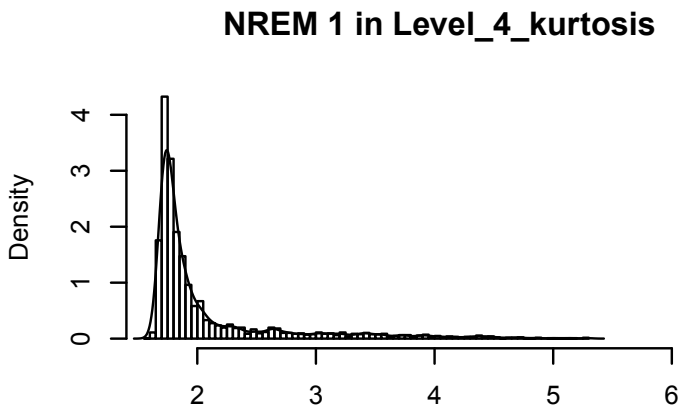

NREM 2 in Level_3_kurtosis
Density
0.0
1.0
2.0
3.0
1
2
3
4
5
6

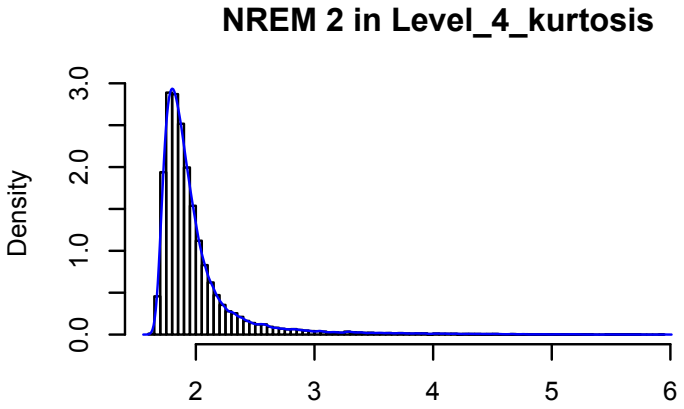

NREM 3 in Level_3_kurtosis
Density
0.0
1.0
2.0
3.0
1
2
3
4
5
6

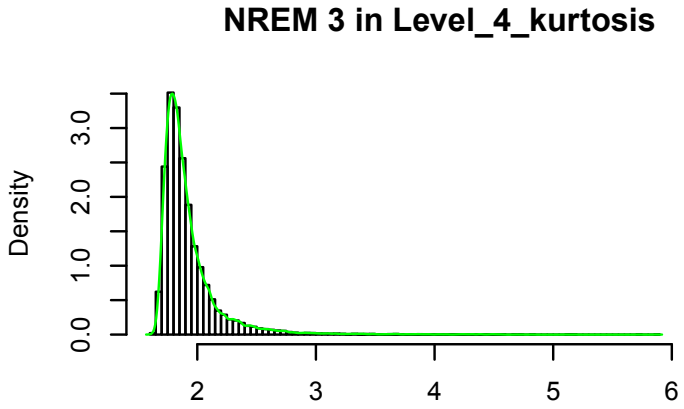

REM in Level_3_kurtosis
Density
0
1
2
3
4
1
2
3
4
5
6

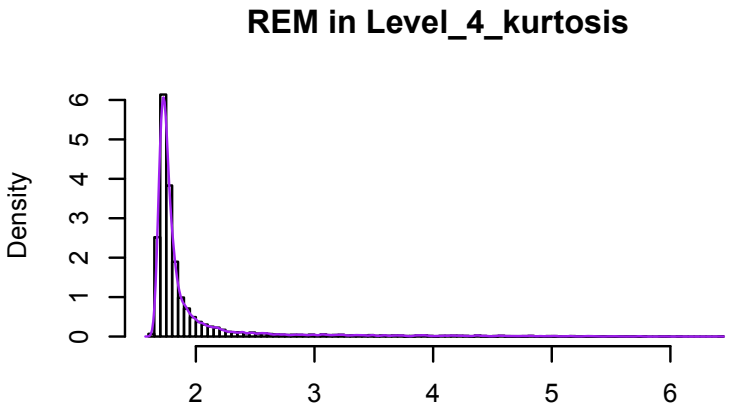

Wake in Level_3_kurtosis
Density
0.0
1.0
2.0
1
2
3
4
5
6

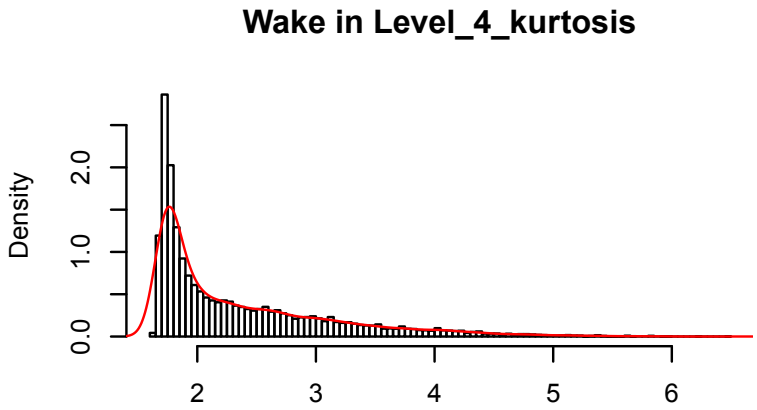

Level_4_kurtosis
Density
0.0
1.0
2.0
3.0
2
3
4
5
6
WC_trans[, kk]

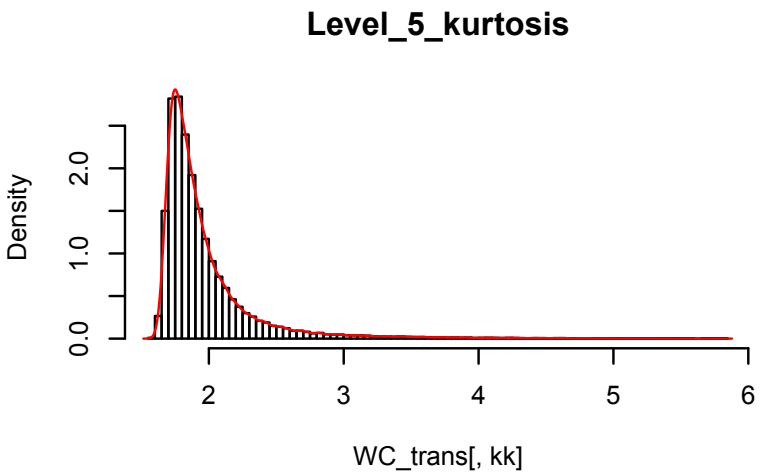

NREM 1 in Level_4_kurtosis
Density
0
1
2
3
4
2
3
4
5
6

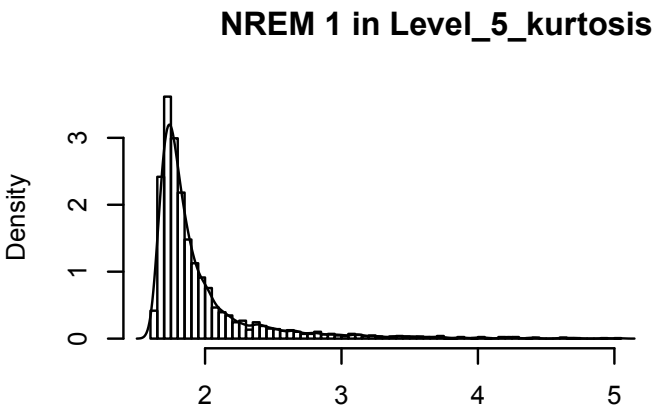

NREM 2 in Level_4_kurtosis
Density
0.0
1.0
2.0
3.0
2
3
4
5
6

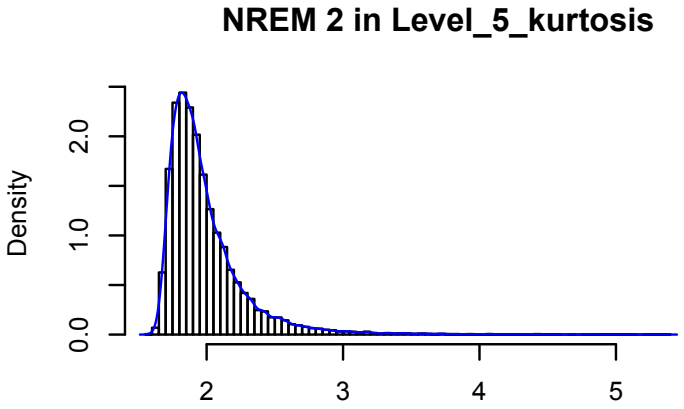

NREM 3 in Level_4_kurtosis
Density
0.0
1.0
2.0
3.0
2
3
4
5
6

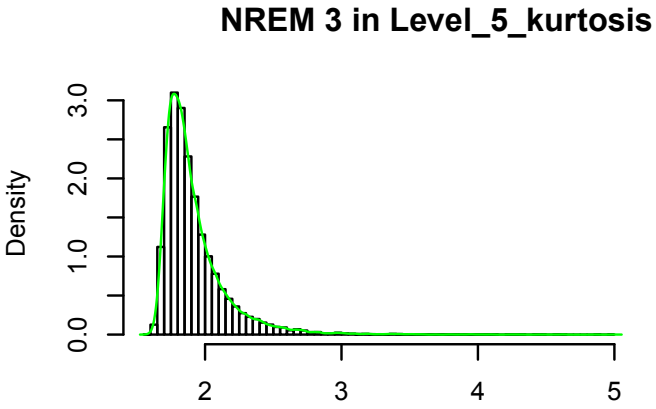

REM in Level_4_kurtosis
Density
0
1
2
3
4
5
6
2
3
4
5
6

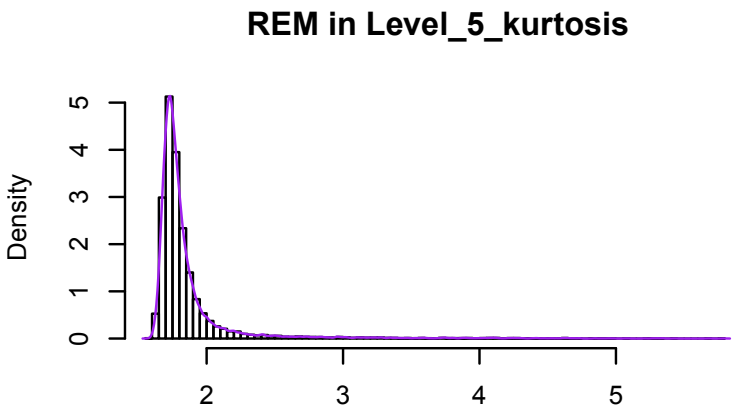

Wake in Level_4_kurtosis
Density
0.0
1.0
2.0
2
3
4
5
6

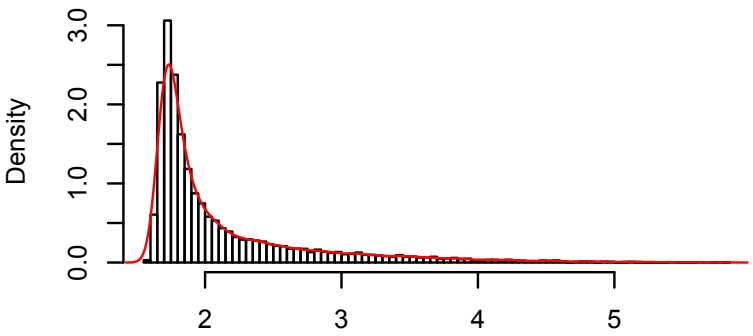

Level_5_kurtosis
Density
0.0
1.0
2.0
2
3
4
5
6
WC_trans[, kk]

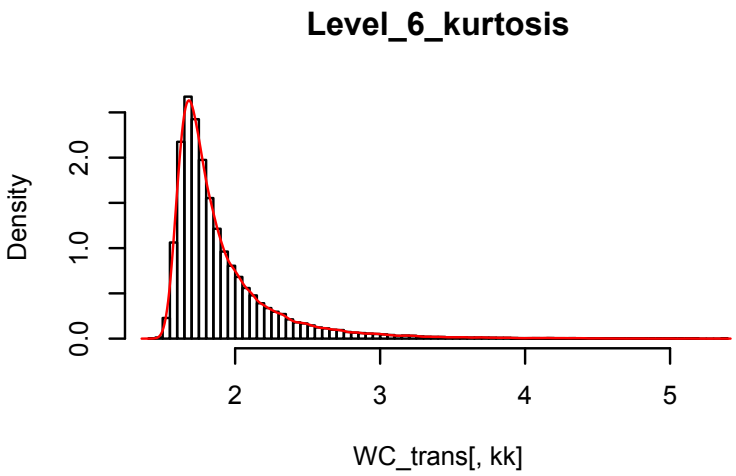

NREM 1 in Level_5_kurtosis
Density
0
1
2
3
2
3
4
5

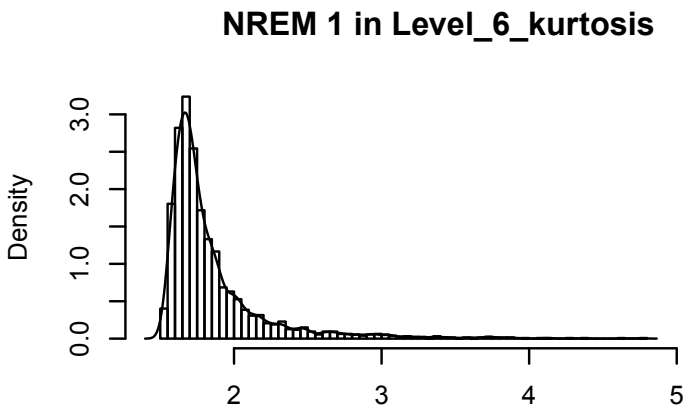

NREM 2 in Level_5_kurtosis
Density
0.0
1.0
2.0
2
3
4
5

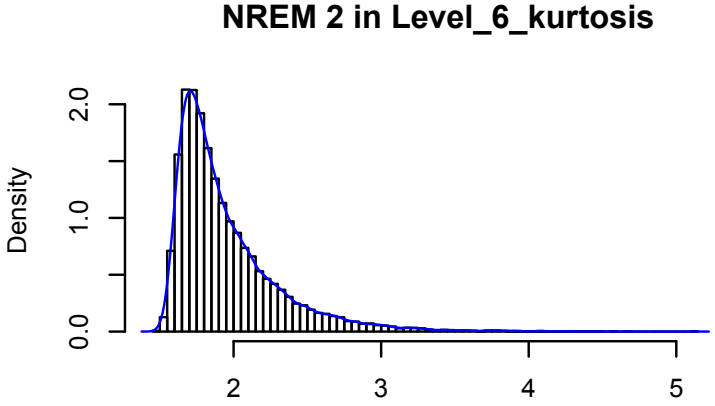

NREM 3 in Level_5_kurtosis
Density
0.0
1.0
2.0
3.0
2
3
4
5

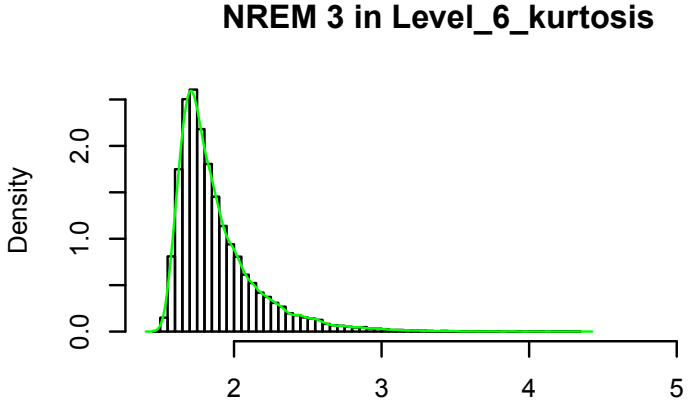

REM in Level_5_kurtosis
Density
0
1
2
3
4
5
2
3
4
5

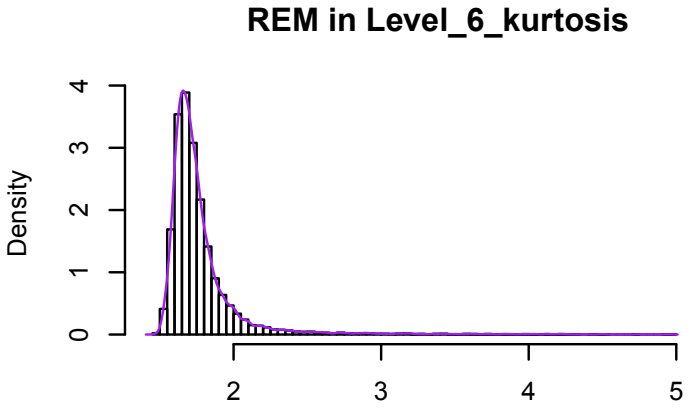

Wake in Level_5_kurtosis
Density
0.0
1.0
2.0
3.0
2
3
4
5

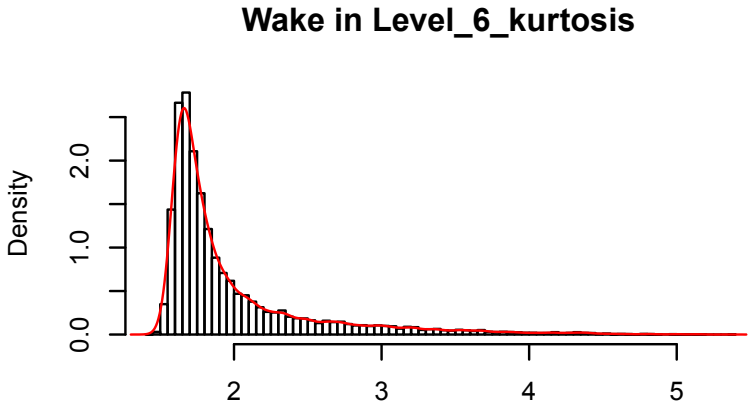

Level_6_kurtosis
Density
0.0
1.0
2.0
2
3
4
5
WC_trans[, kk]

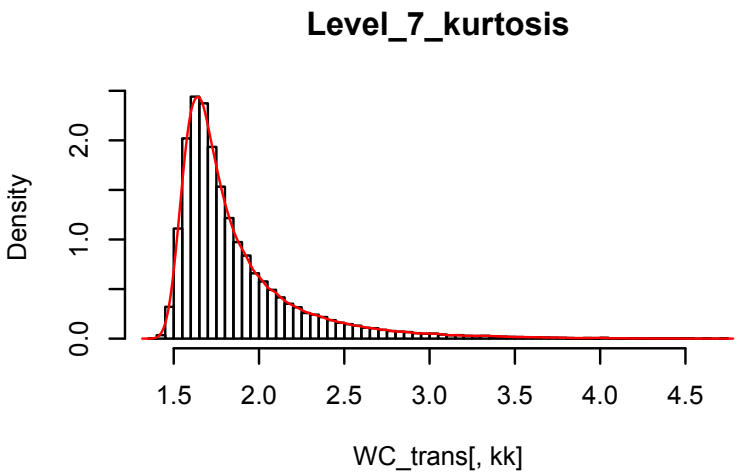

NREM 1 in Level_6_kurtosis
Density
0.0
1.0
2.0
3.0
2
3
4
5

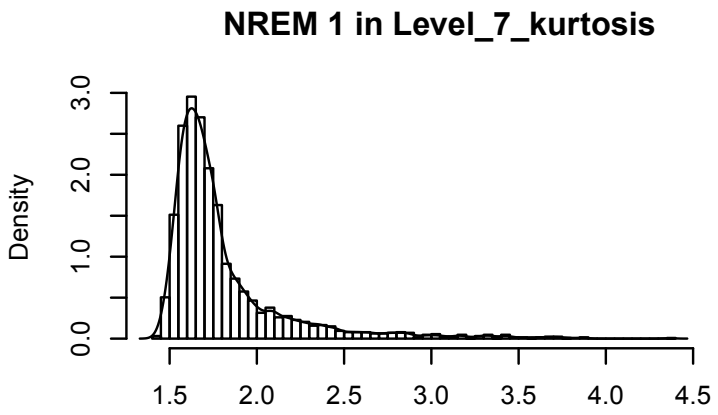

NREM 2 in Level_6_kurtosis
Density
0.0
1.0
2.0
2
3
4
5

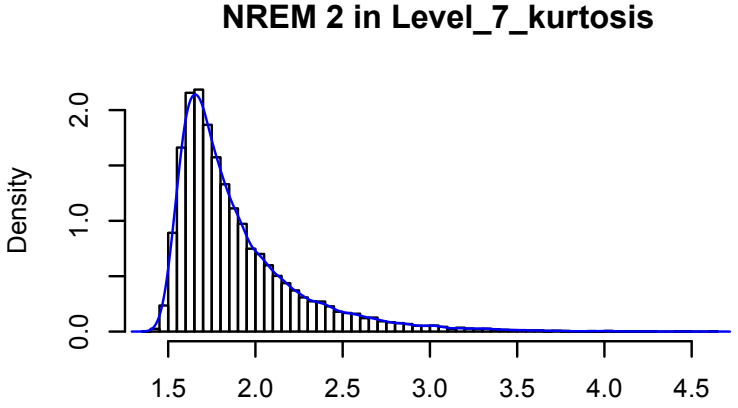

NREM 3 in Level_6_kurtosis
Density
0.0
1.0
2.0
2
3
4
5

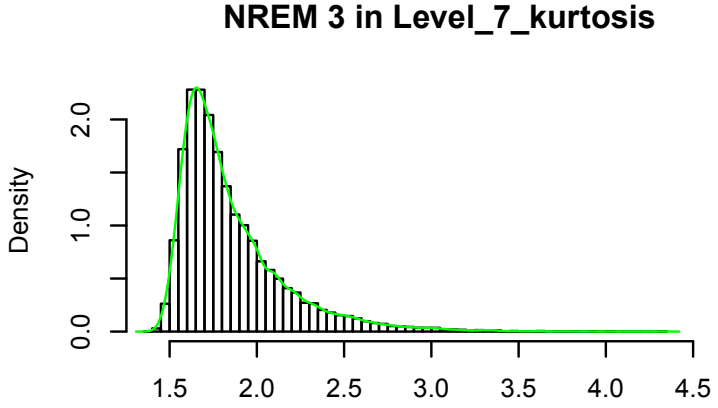

REM in Level_6_kurtosis
Density
0
1
2
3
4
2
3
4
5

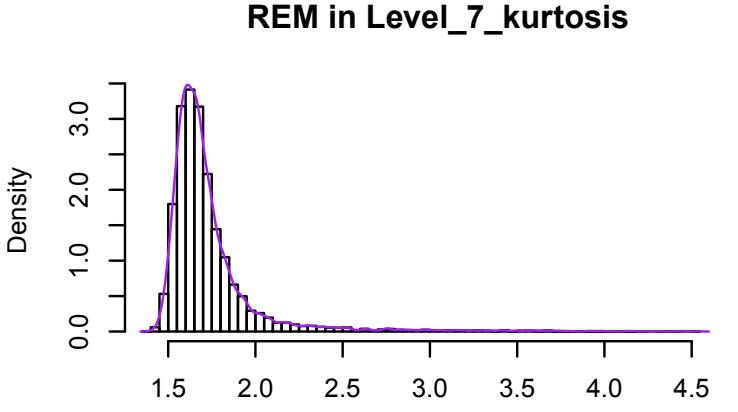

Wake in Level_6_kurtosis
Density
0.0
1.0
2.0
2
3
4
5

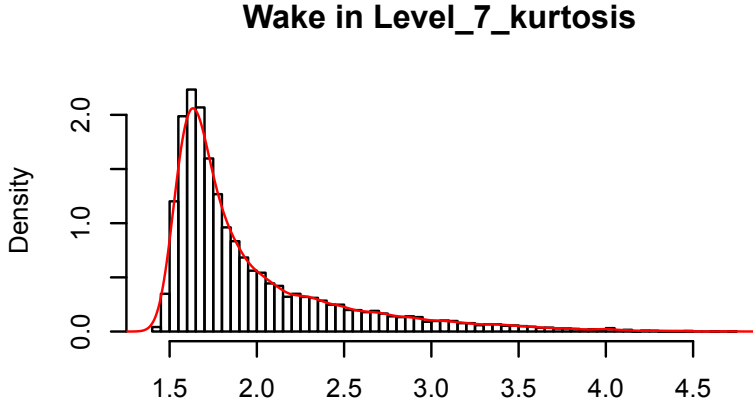

Level_7_kurtosis
Density
0.0
1.0
2.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
WC_trans[, kk]

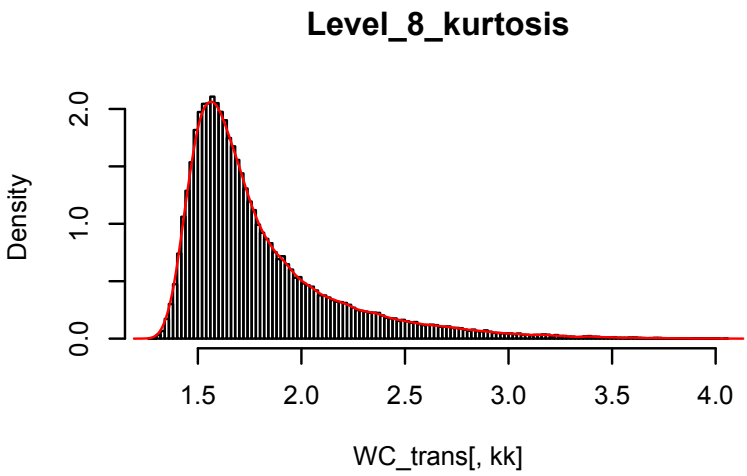

NREM 1 in Level_7_kurtosis
Density
0.0
1.0
2.0
3.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5

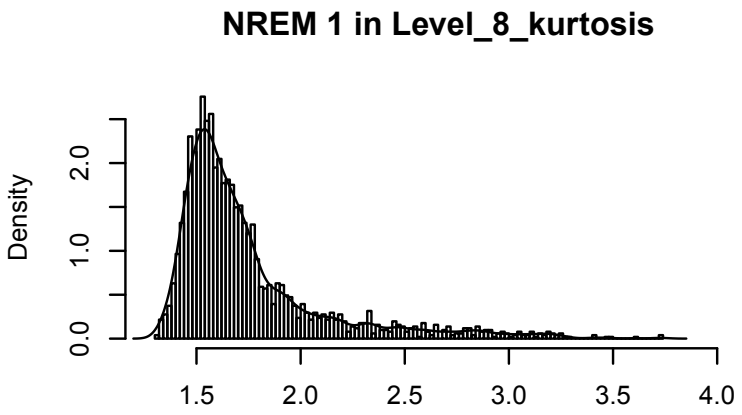

NREM 2 in Level_7_kurtosis
Density
0.0
1.0
2.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5

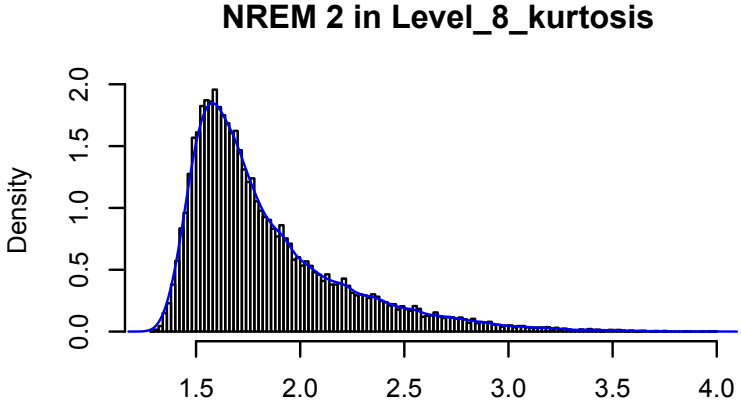

NREM 3 in Level_7_kurtosis
Density
0.0
1.0
2.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5

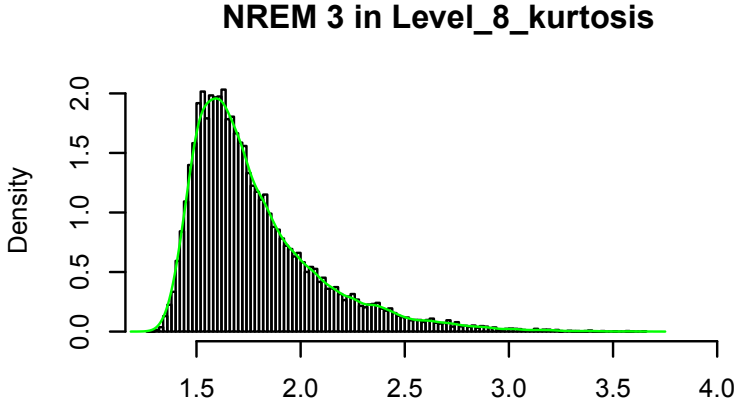

REM in Level_7_kurtosis
Density
0.0
1.0
2.0
3.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5

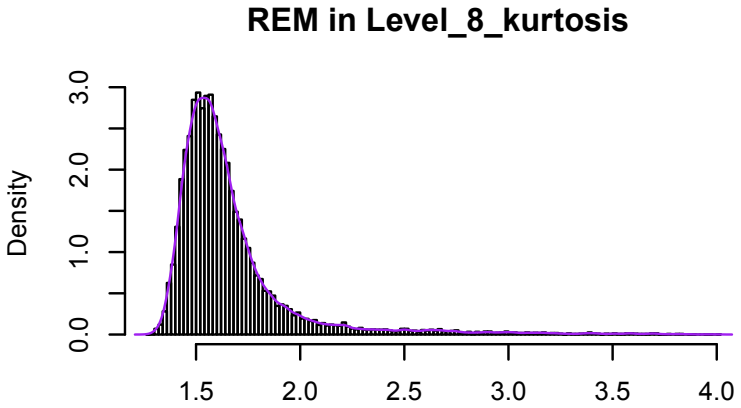

Wake in Level_7_kurtosis
Density
0.0
1.0
2.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5

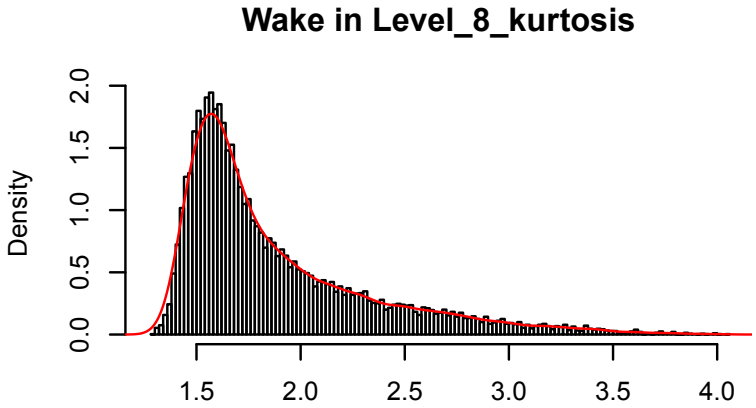

Level_8_kurtosis
Density
0.0
1.0
2.0
1.5
2.0
2.5
3.0
3.5
4.0
WC_trans[, kk]

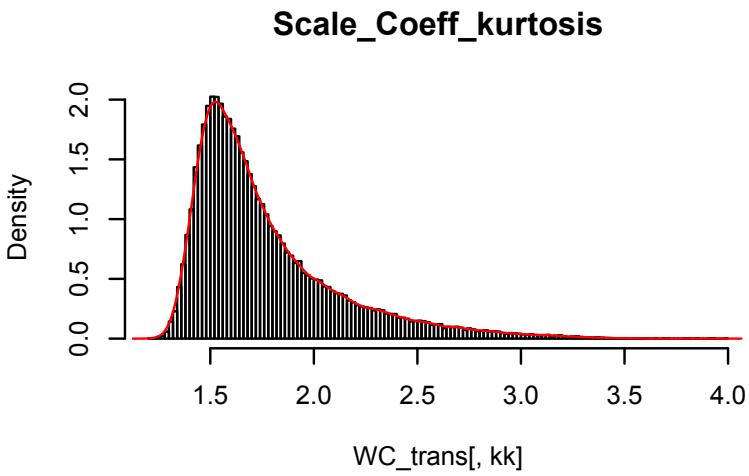

NREM 1 in Level_8_kurtosis
Density
0.0
1.0
2.0
1.5
2.0
2.5
3.0
3.5
4.0

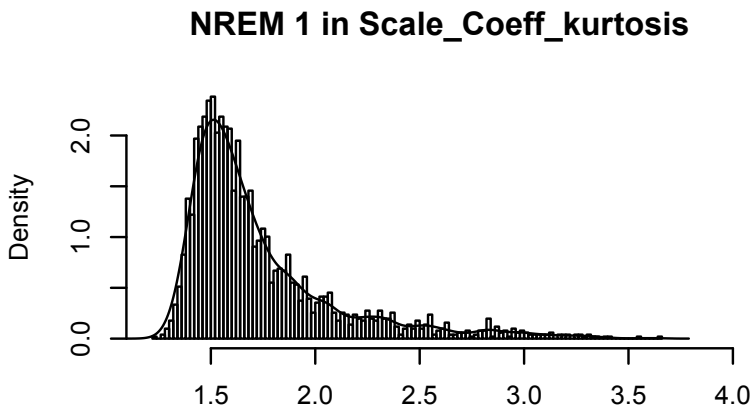

NREM 2 in Level_8_kurtosis
Density
0.0
0.5
1.0
1.5
2.0
1.5
2.0
2.5
3.0
3.5
4.0

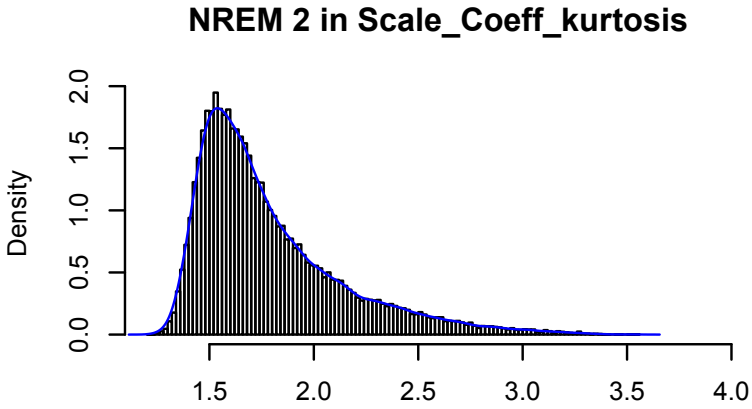

NREM 3 in Level_8_kurtosis
Density
0.0
0.5
1.0
1.5
2.0
1.5
2.0
2.5
3.0
3.5
4.0

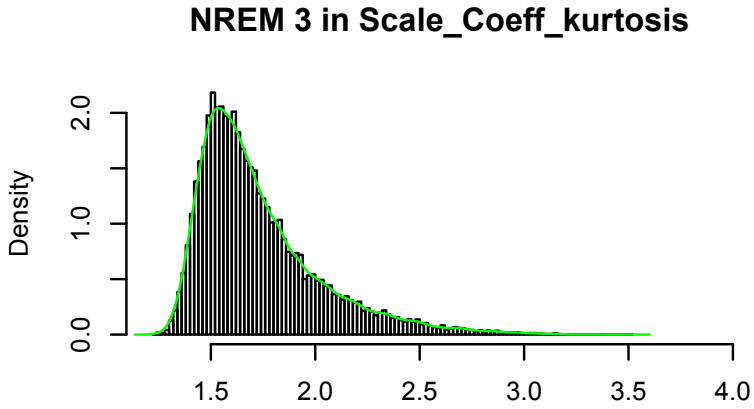

REM in Level_8_kurtosis
Density
0.0
1.0
2.0
3.0
1.5
2.0
2.5
3.0
3.5
4.0

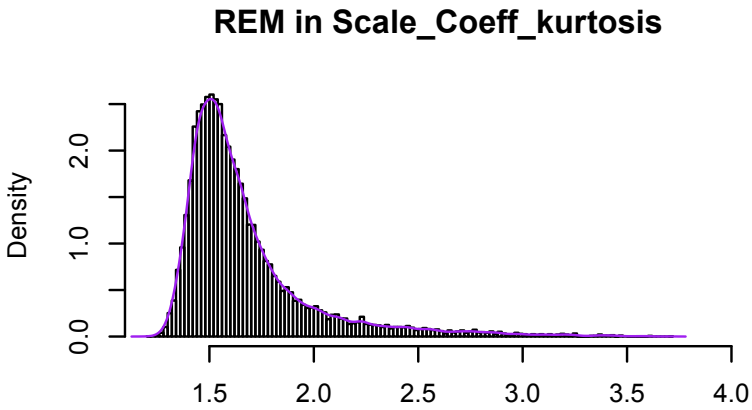

Wake in Level_8_kurtosis
Density
0.0
0.5
1.0
1.5
2.0
1.5
2.0
2.5
3.0
3.5
4.0

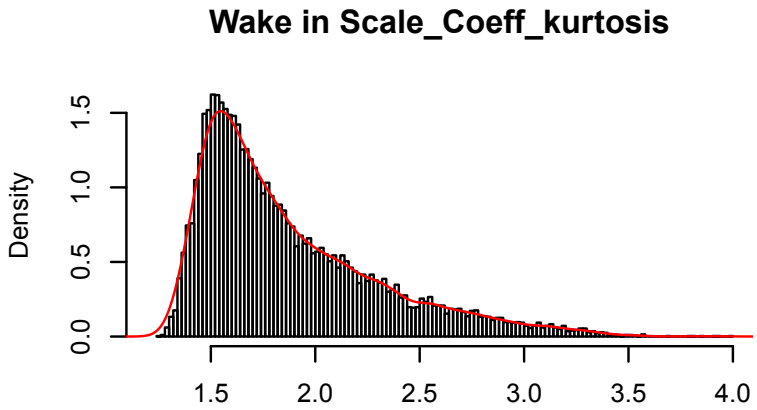

Scale_Coeff_kurtosis
Density
0.0
0.5
1.0
1.5
2.0
1.5
2.0
2.5
3.0
3.5
4.0
WC_trans[, kk]


NREM 1 in Scale_Coeff_kurtosis
Density
0.0
1.0
2.0
1.5
2.0
2.5
3.0
3.5
4.0


NREM 2 in Scale_Coeff_kurtosis
Density
0.0
0.5
1.0
1.5
2.0
1.5
2.0
2.5
3.0
3.5
4.0

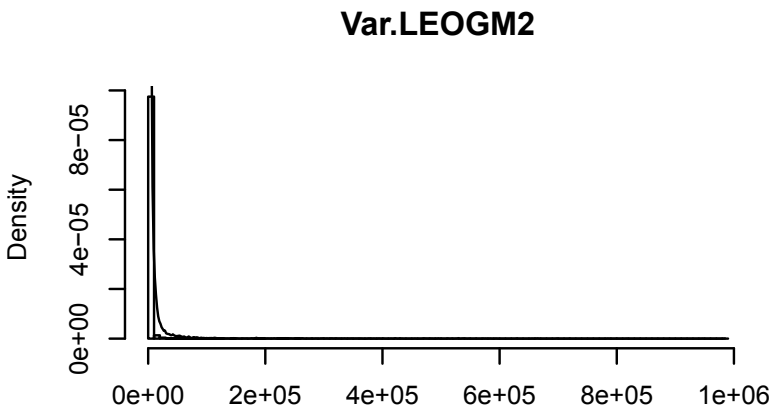

NREM 3 in Scale_Coeff_kurtosis
Density
0.0
1.0
2.0
1.5
2.0
2.5
3.0
3.5
4.0

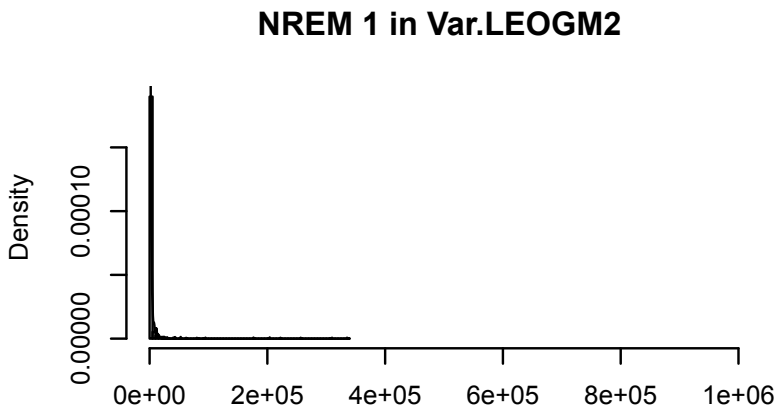

REM in Scale_Coeff_kurtosis
Density
0.0
1.0
2.0
1.5
2.0
2.5
3.0
3.5
4.0

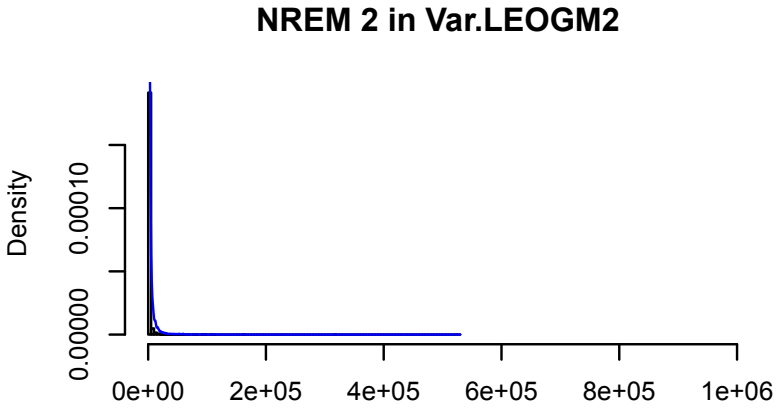

Wake in Scale_Coeff_kurtosis
Density
0.0
0.5
1.0
1.5
1.5
2.0
2.5
3.0
3.5
4.0

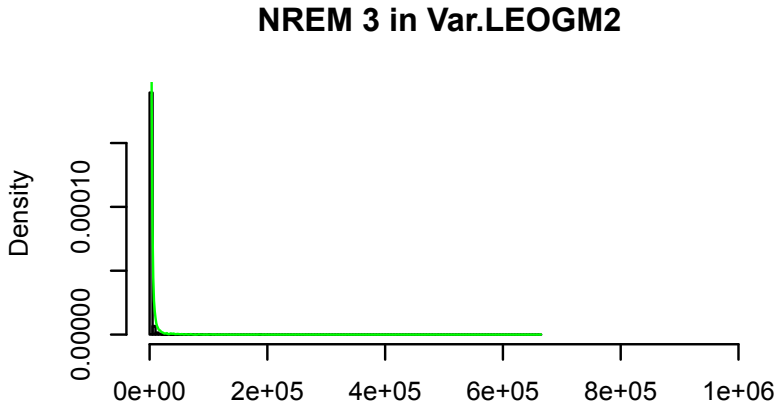
Appendix A.3

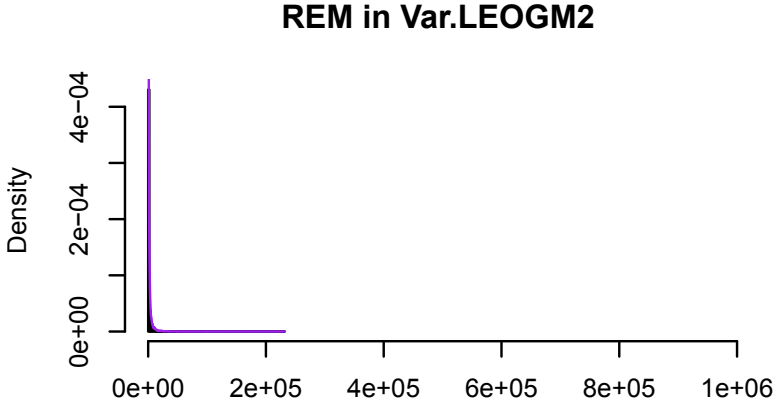
This appendix contains the density plots of all original non-EEG features, the QQ plots of the transformed non-EEG variance and kurtosis features, and the original skewness features. Lastly, Density plots of the transformed non-EEG variance and kurtosis features.

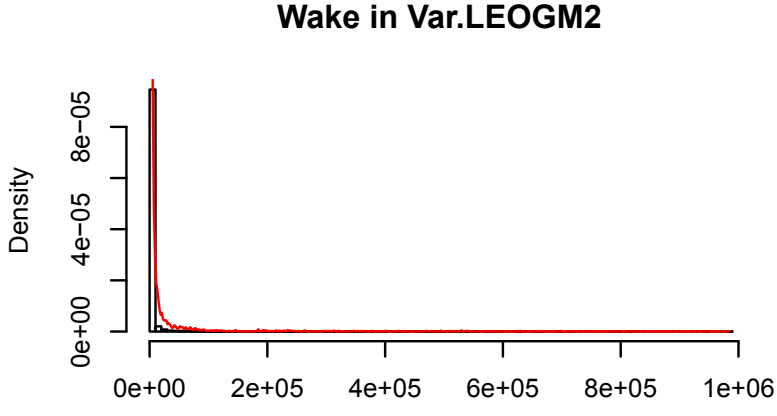

Var.LEOGM2
Density
0e+00
4e−05
8e−05
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06

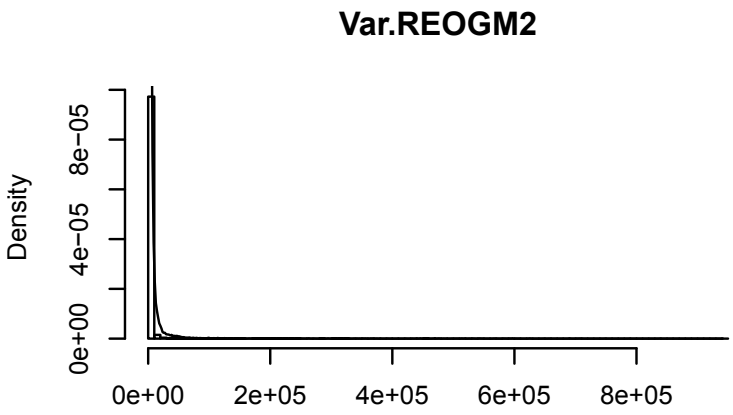

NREM 1 in Var.LEOGM2
Density
0.00000
0.00010
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06

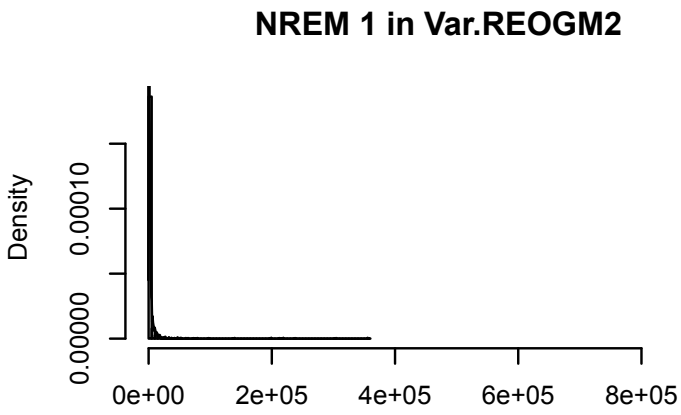

NREM 2 in Var.LEOGM2
Density
0.00000
0.00010
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06

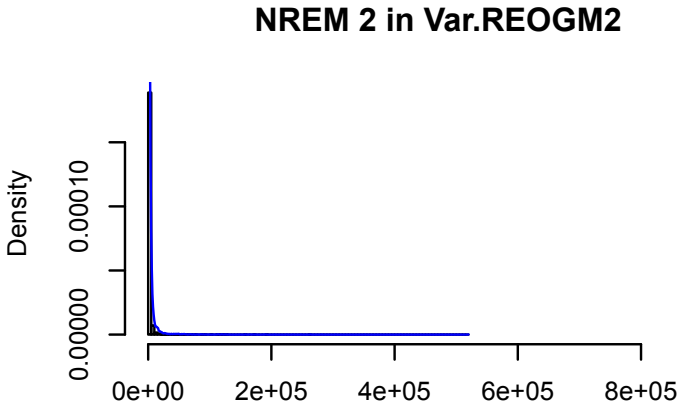

NREM 3 in Var.LEOGM2
Density
0.00000
0.00010
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06

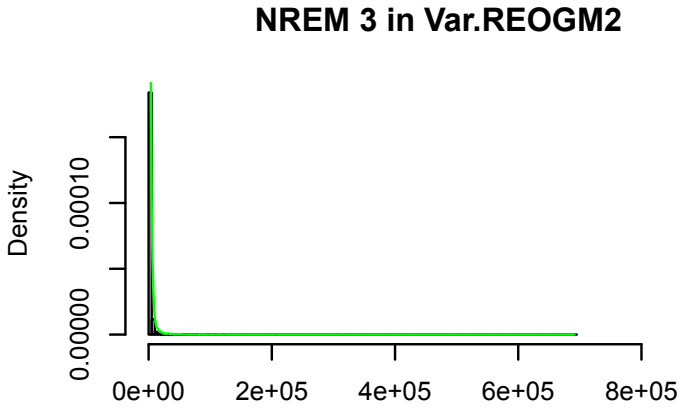

REM in Var.LEOGM2
Density
0e+00
2e−04
4e−04
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06

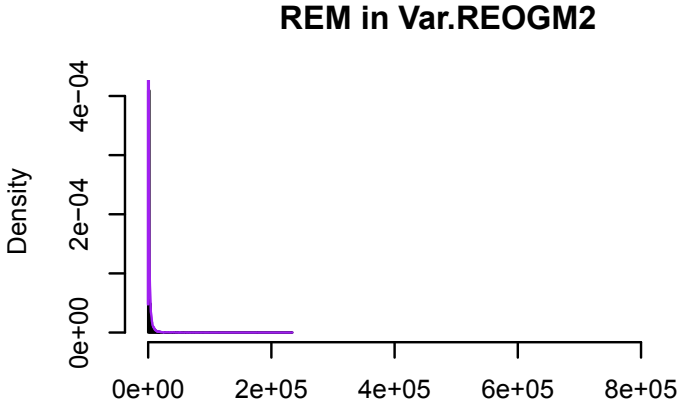

Wake in Var.LEOGM2
Density
0e+00
4e−05
8e−05
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06

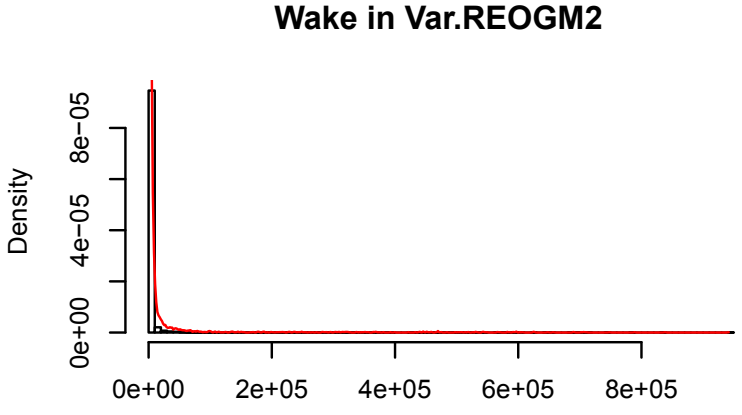

Var.REOGM2
Density
0e+00
4e−05
8e−05
0e+00
2e+05
4e+05
6e+05
8e+05

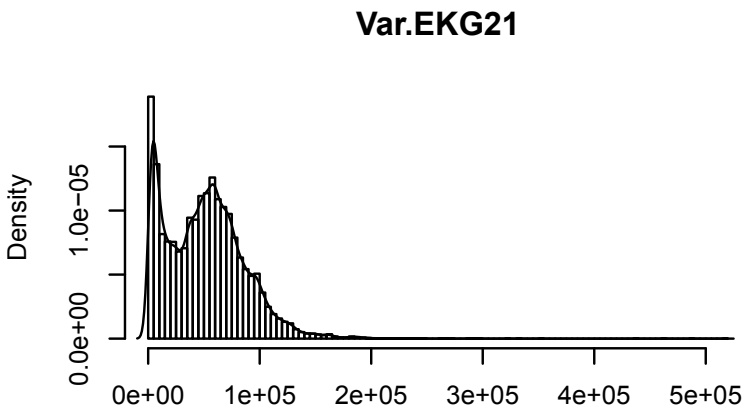

NREM 1 in Var.REOGM2
Density
0.00000
0.00010
0e+00
2e+05
4e+05
6e+05
8e+05

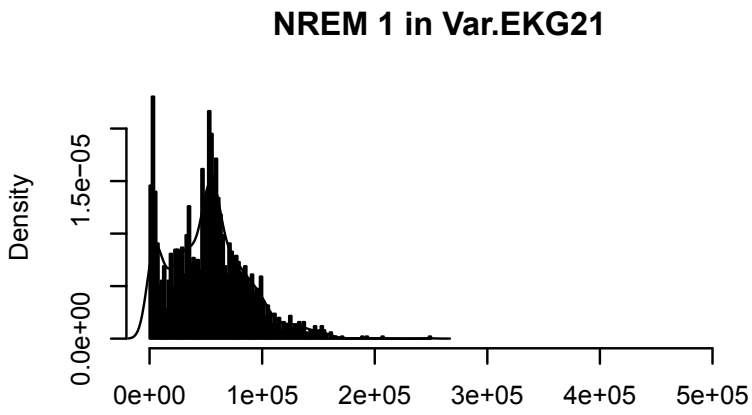

NREM 2 in Var.REOGM2
Density
0.00000
0.00010
0e+00
2e+05
4e+05
6e+05
8e+05

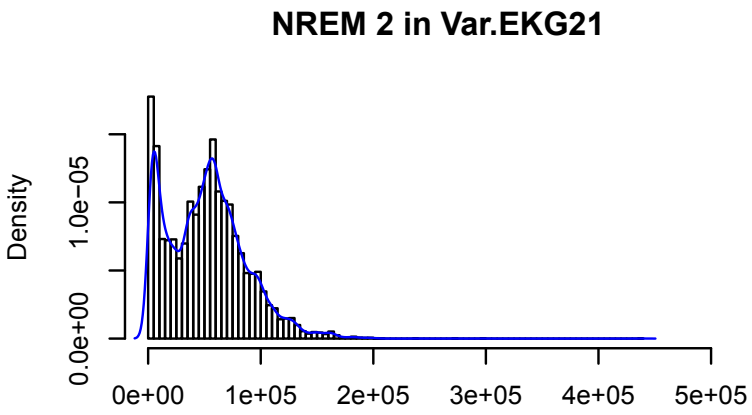

NREM 3 in Var.REOGM2
Density
0.00000
0.00010
0e+00
2e+05
4e+05
6e+05
8e+05

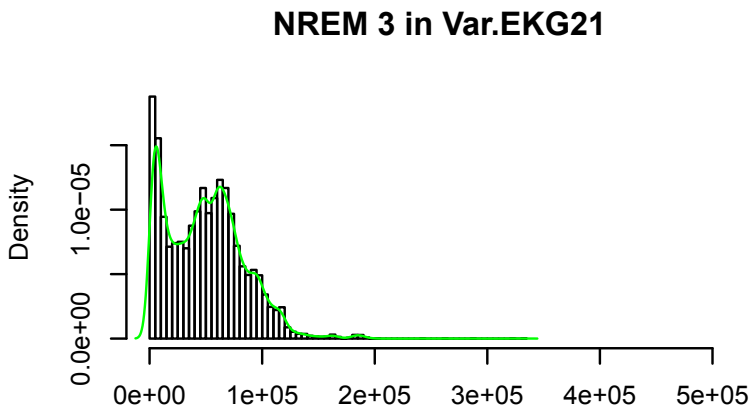

REM in Var.REOGM2
Density
0e+00
2e−04
4e−04
0e+00
2e+05
4e+05
6e+05
8e+05

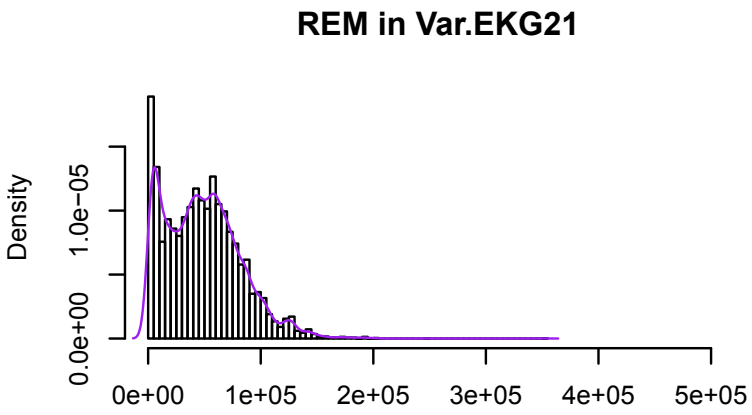

Wake in Var.REOGM2
Density
0e+00
4e−05
8e−05
0e+00
2e+05
4e+05
6e+05
8e+05

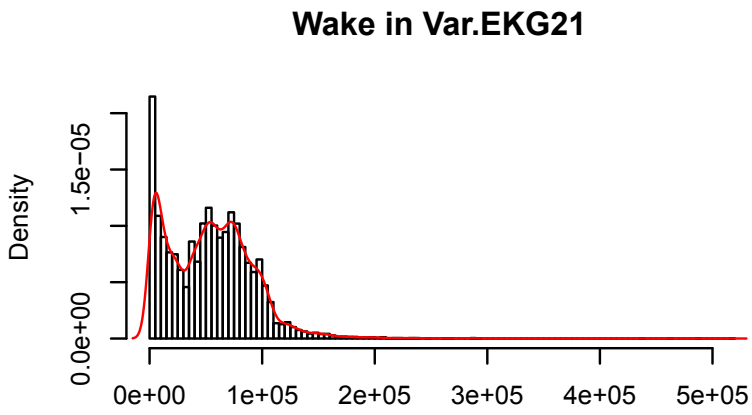

Var.EKG21
Density
0.0e+00
1.0e−05
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05

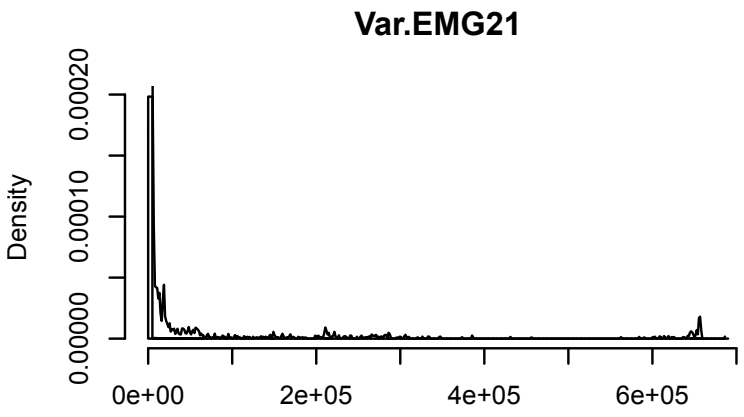

NREM 1 in Var.EKG21
Density
0.0e+00
1.5e−05
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05

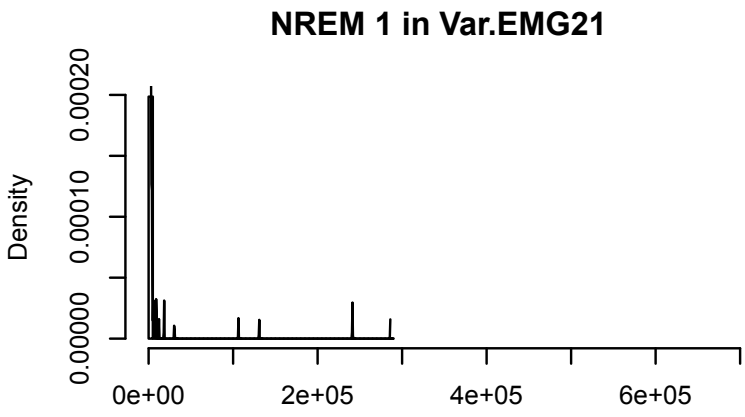

NREM 2 in Var.EKG21
Density
0.0e+00
1.0e−05
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05

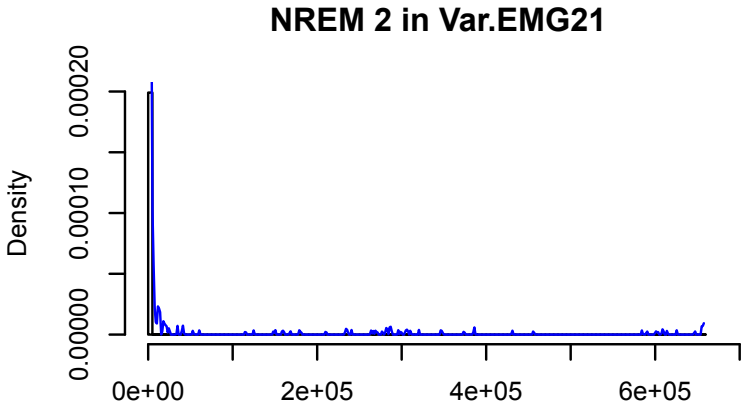

NREM 3 in Var.EKG21
Density
0.0e+00
1.0e−05
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05

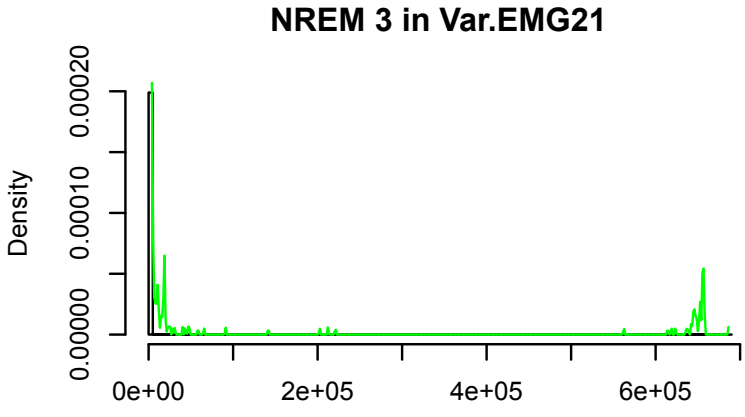

REM in Var.EKG21
Density
0.0e+00
1.0e−05
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05

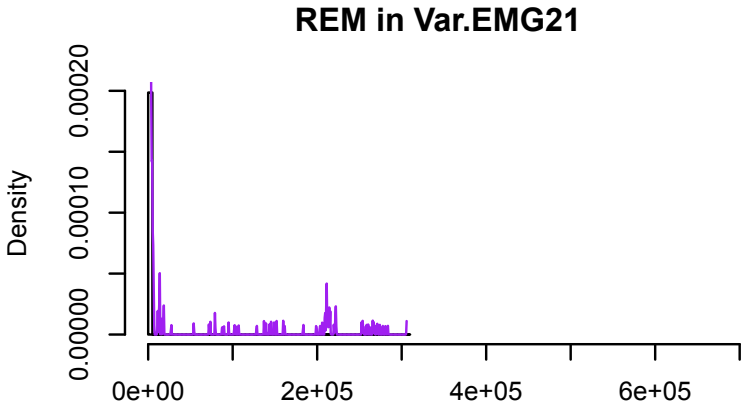

Wake in Var.EKG21
Density
0.0e+00
1.5e−05
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05

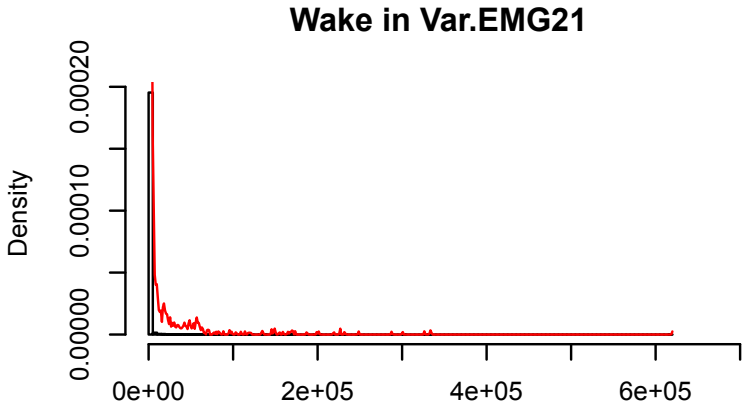

Var.EMG21
Density
0.00000
0.00010
0.00020
0e+00
2e+05
4e+05
6e+05

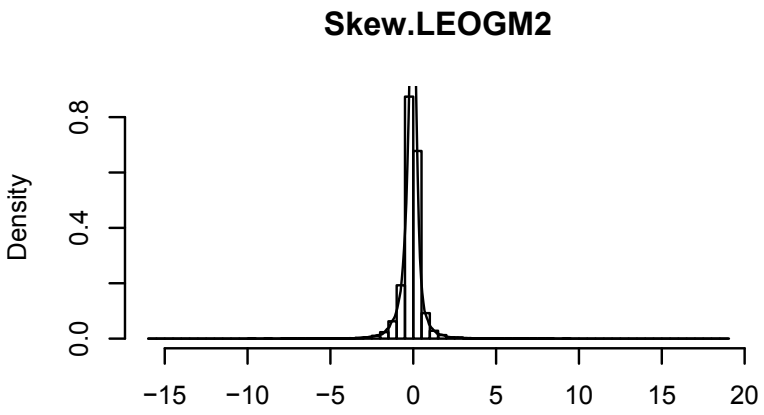

NREM 1 in Var.EMG21
Density
0.00000
0.00010
0.00020
0e+00
2e+05
4e+05
6e+05

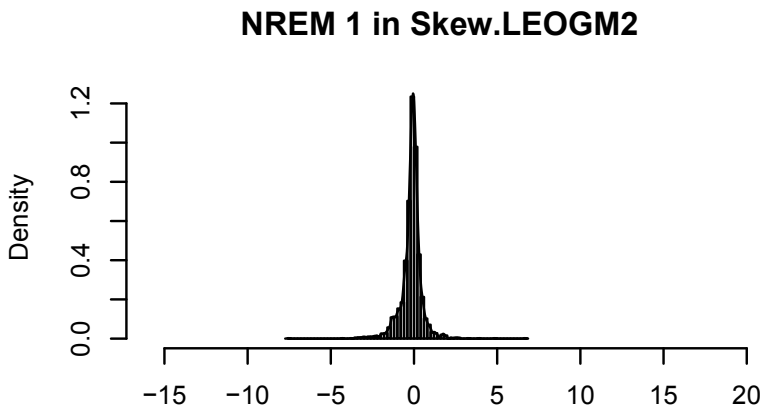

NREM 2 in Var.EMG21
Density
0.00000
0.00010
0.00020
0e+00
2e+05
4e+05
6e+05

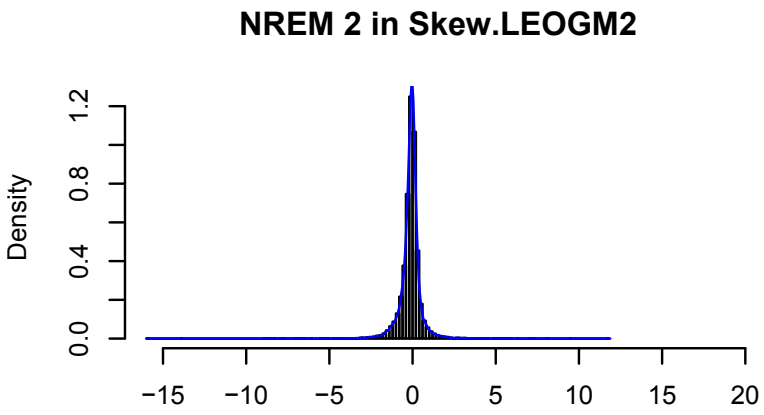

NREM 3 in Var.EMG21
Density
0.00000
0.00010
0.00020
0e+00
2e+05
4e+05
6e+05

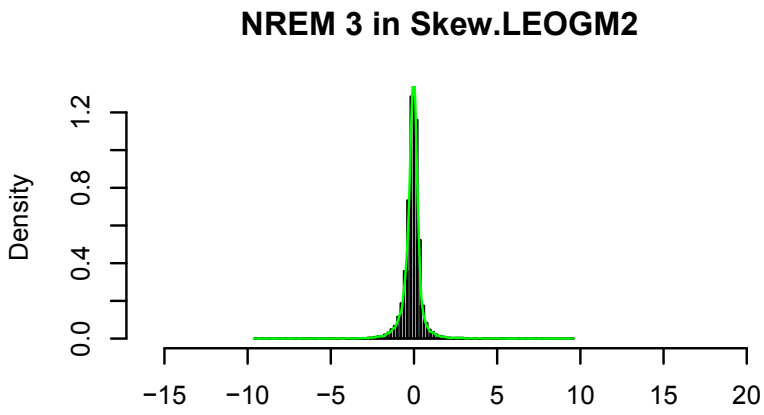

REM in Var.EMG21
Density
0.00000
0.00010
0.00020
0e+00
2e+05
4e+05
6e+05

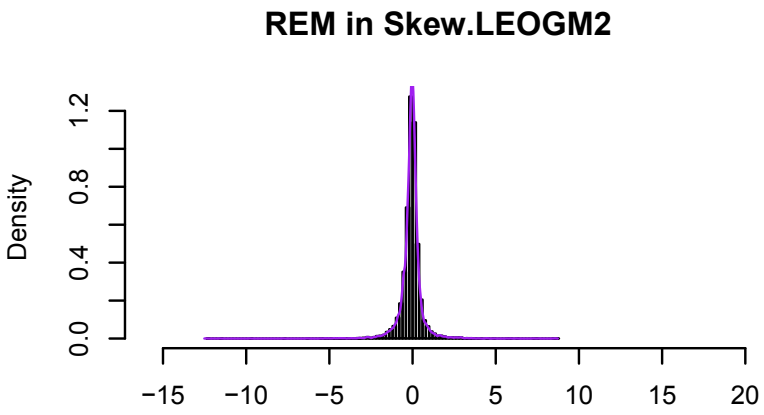

Wake in Var.EMG21
Density
0.00000
0.00010
0.00020
0e+00
2e+05
4e+05
6e+05

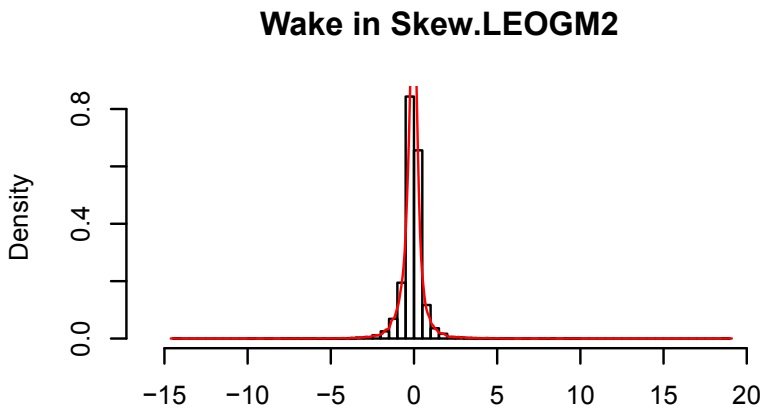

Skew.LEOGM2
Density
0.0
0.4
0.8
−15
−10
−5
0
5
10
15
20

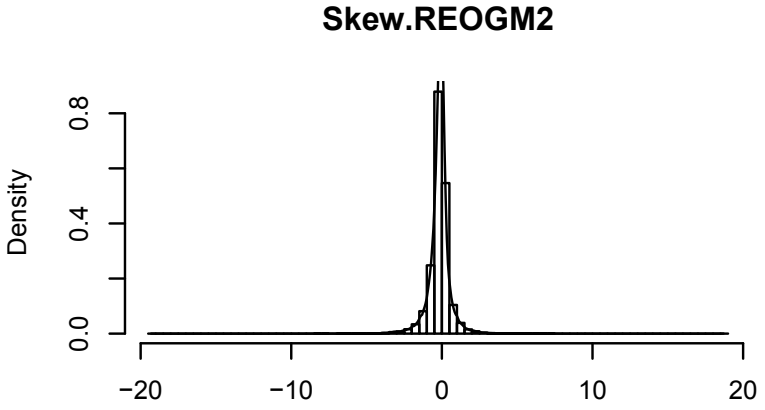

NREM 1 in Skew.LEOGM2
Density
0.0
0.4
0.8
1.2
−15
−10
−5
0
5
10
15
20

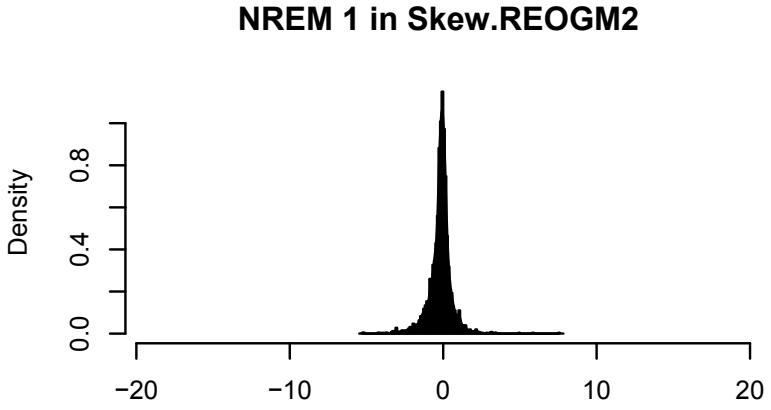

NREM 2 in Skew.LEOGM2
Density
0.0
0.4
0.8
1.2
−15
−10
−5
0
5
10
15
20

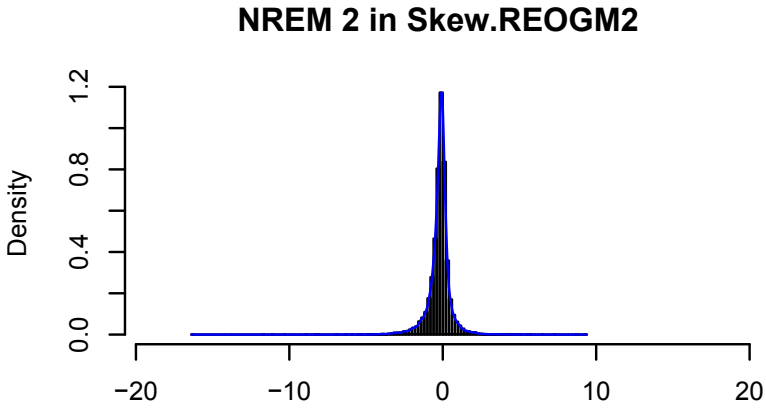

NREM 3 in Skew.LEOGM2
Density
0.0
0.4
0.8
1.2
−15
−10
−5
0
5
10
15
20

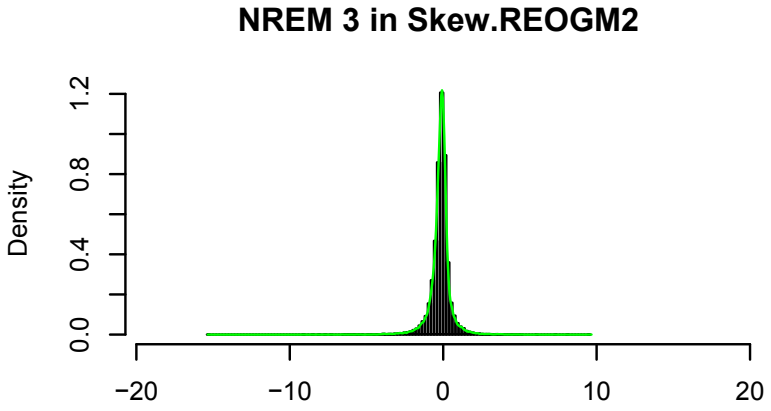

REM in Skew.LEOGM2
Density
0.0
0.4
0.8
1.2
−15
−10
−5
0
5
10
15
20

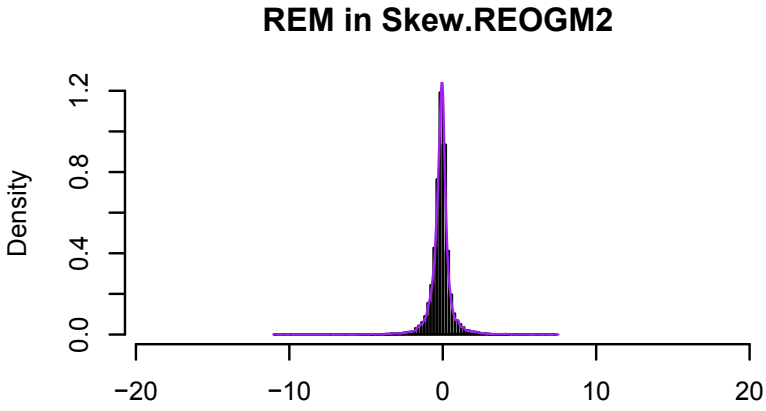

Wake in Skew.LEOGM2
Density
0.0
0.4
0.8
−15
−10
−5
0
5
10
15
20

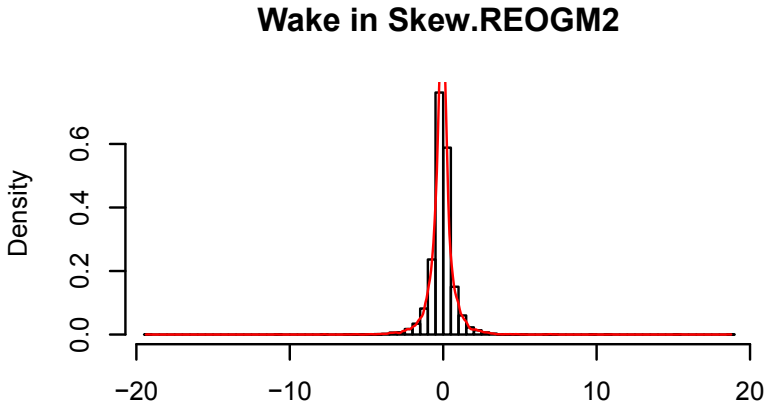

Skew.REOGM2
Density
0.0
0.4
0.8
−20
−10
0
10
20

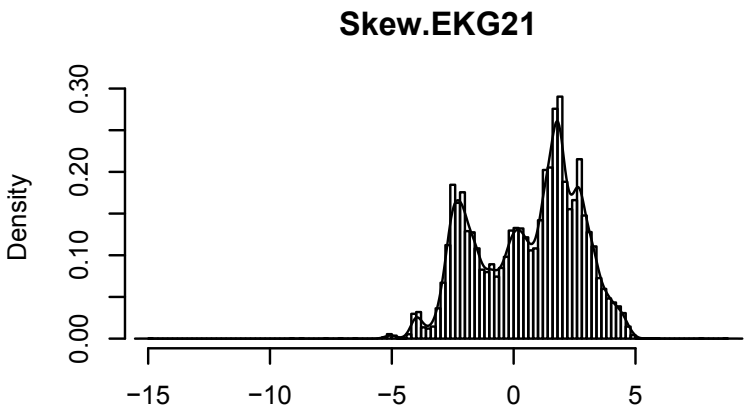

NREM 1 in Skew.REOGM2
Density
0.0
0.4
0.8
−20
−10
0
10
20

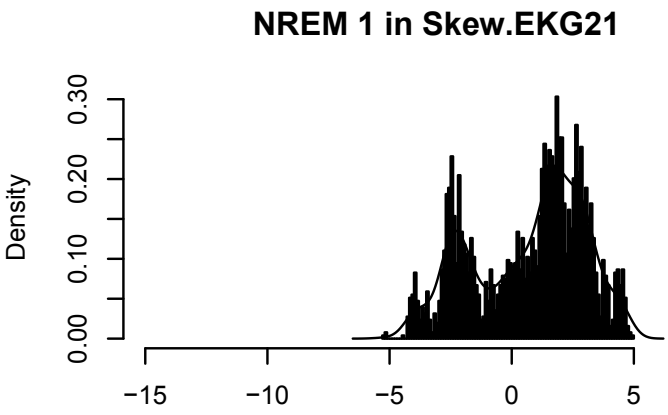

NREM 2 in Skew.REOGM2
Density
0.0
0.4
0.8
1.2
−20
−10
0
10
20

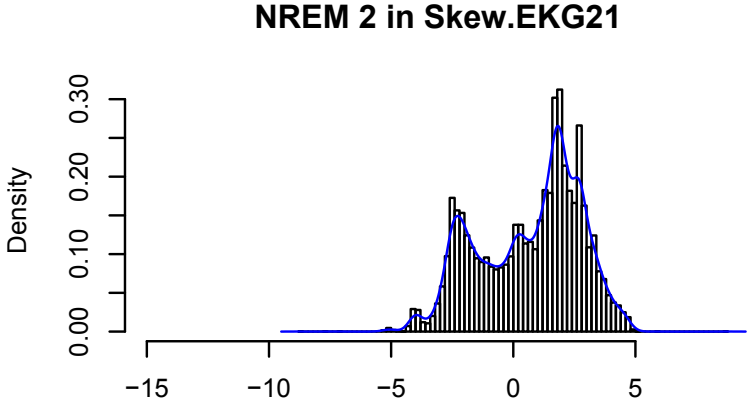

NREM 3 in Skew.REOGM2
Density
0.0
0.4
0.8
1.2
−20
−10
0
10
20

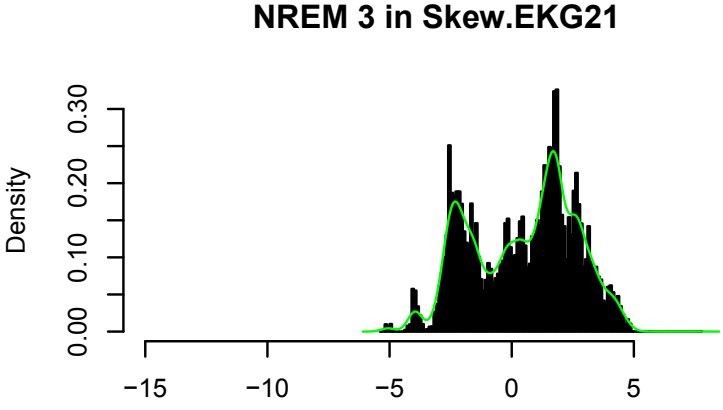

REM in Skew.REOGM2
Density
0.0
0.4
0.8
1.2
−20
−10
0
10
20

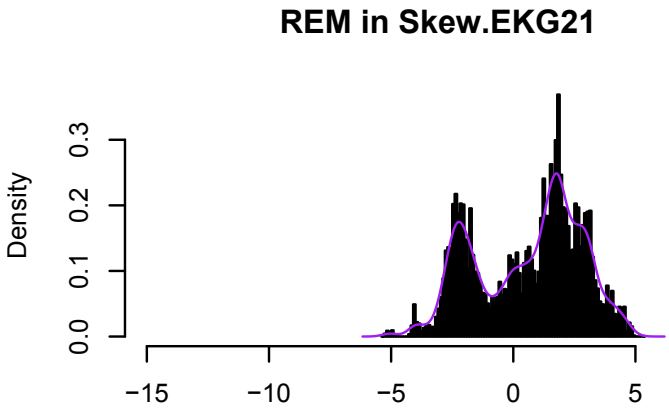

Wake in Skew.REOGM2
Density
0.0
0.2
0.4
0.6
−20
−10
0
10
20

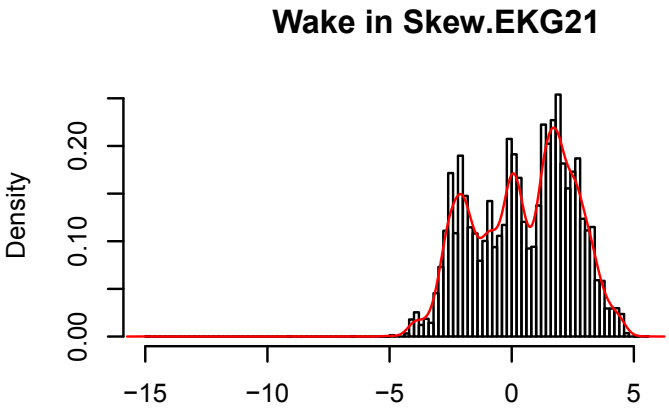

Skew.EKG21
Density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5

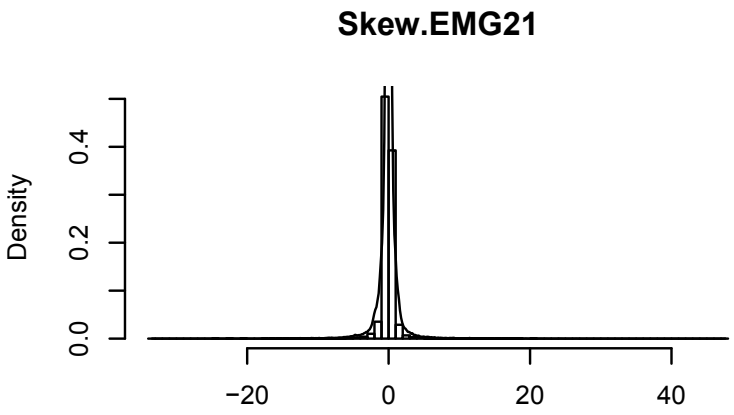

NREM 1 in Skew.EKG21
Density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5

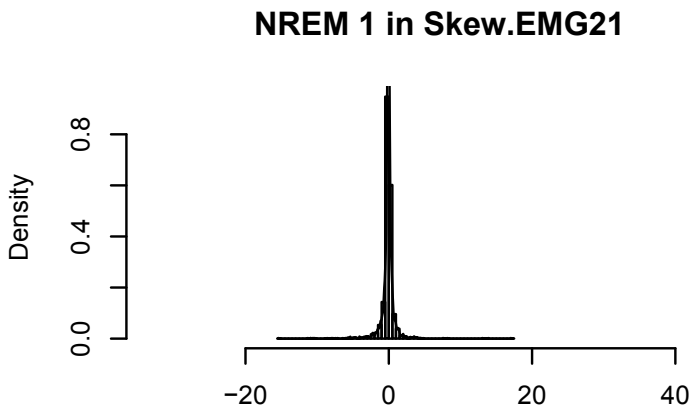

NREM 2 in Skew.EKG21
Density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5

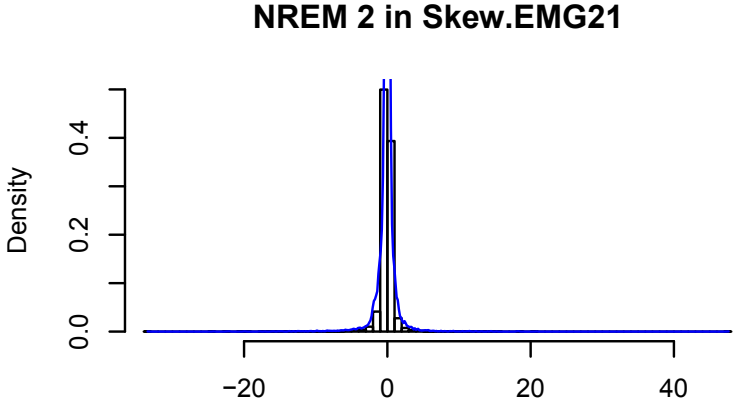

NREM 3 in Skew.EKG21
Density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5

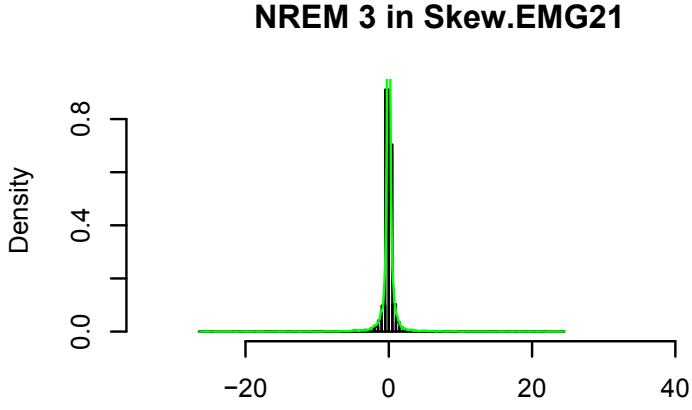

REM in Skew.EKG21
Density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5

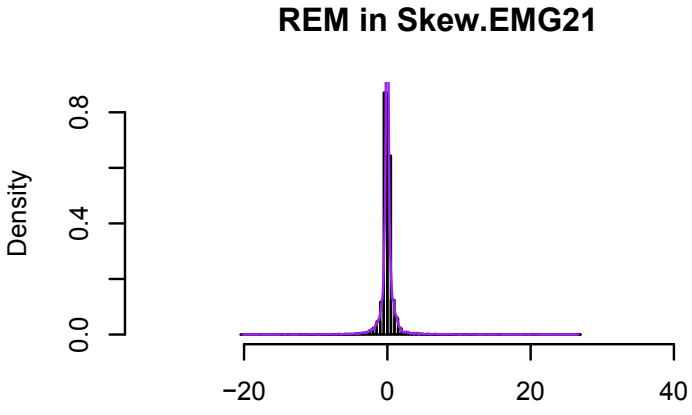

Wake in Skew.EKG21
Density
0.00
0.10
0.20
−15
−10
−5
0
5

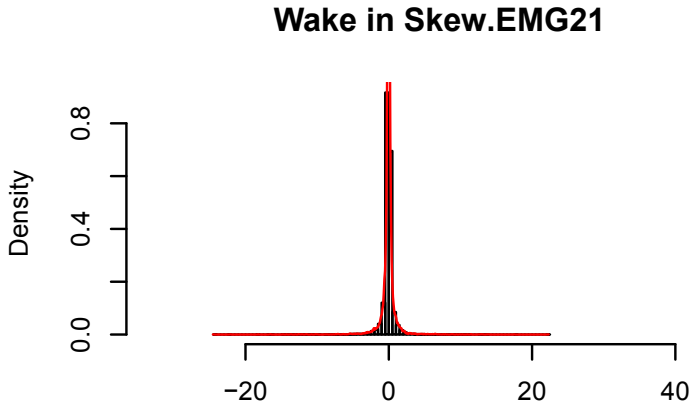

Skew.EMG21
Density
0.0
0.2
0.4
−20
0
20
40

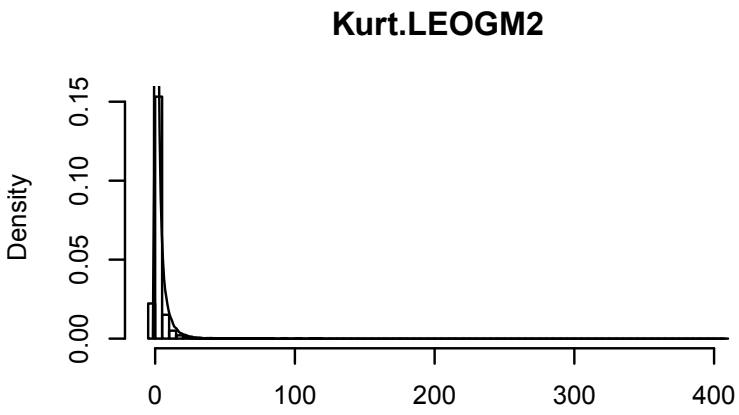

NREM 1 in Skew.EMG21
Density
0.0
0.4
0.8
−20
0
20
40

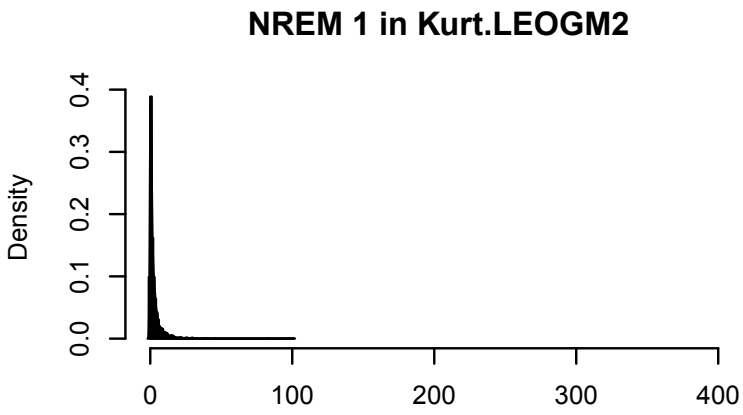

NREM 2 in Skew.EMG21
Density
0.0
0.2
0.4
−20
0
20
40

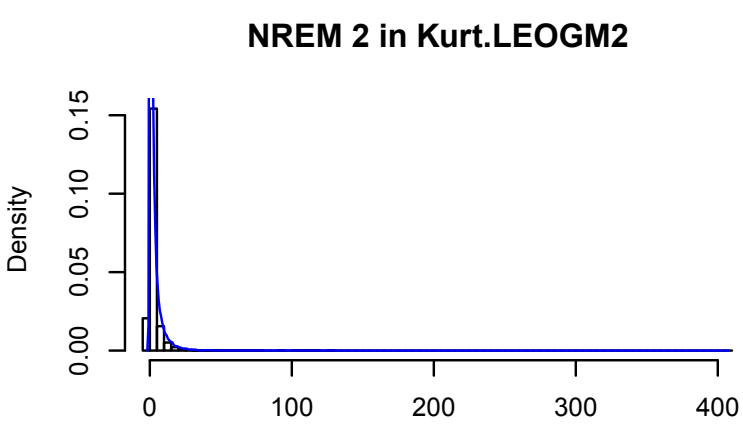

NREM 3 in Skew.EMG21
Density
0.0
0.4
0.8
−20
0
20
40

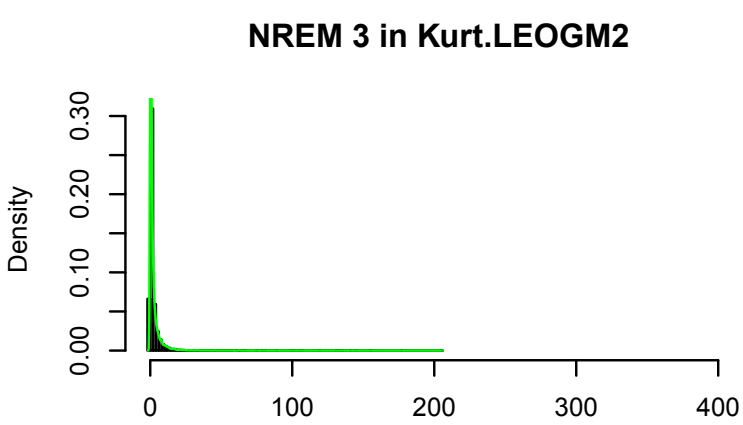

REM in Skew.EMG21
Density
0.0
0.4
0.8
−20
0
20
40

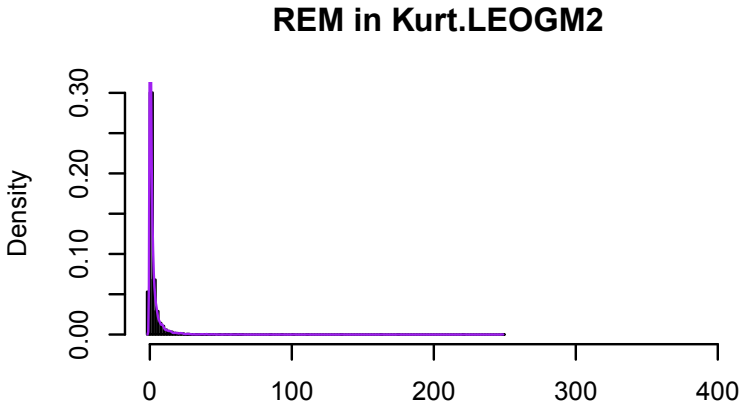

Wake in Skew.EMG21
Density
0.0
0.4
0.8
−20
0
20
40

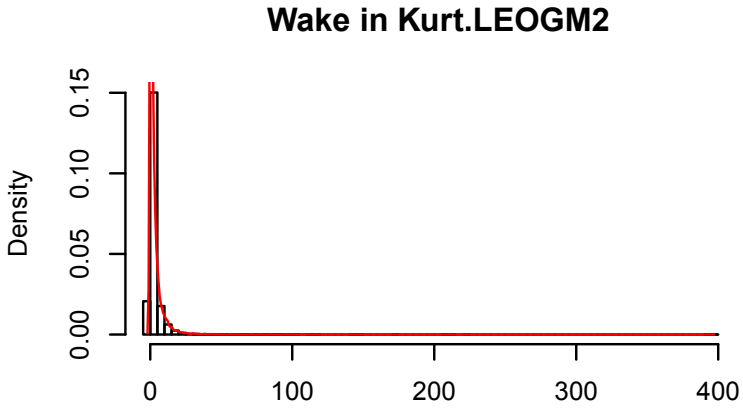

Kurt.LEOGM2
Density
0.00
0.05
0.10
0.15
0
100
200
300
400

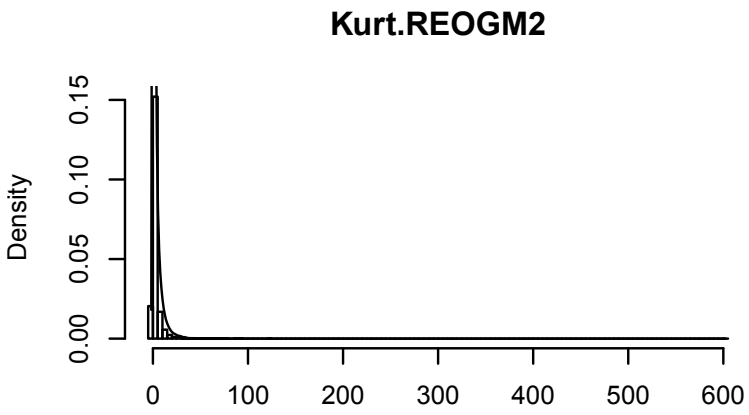

NREM 1 in Kurt.LEOGM2
Density
0.0
0.1
0.2
0.3
0.4
0
100
200
300
400

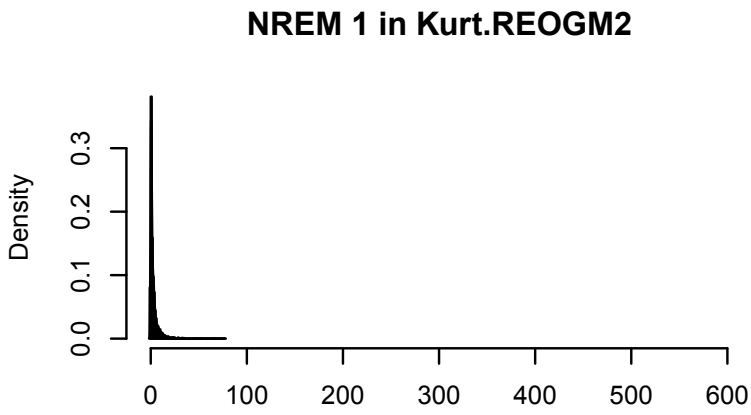

NREM 2 in Kurt.LEOGM2
Density
0.00
0.05
0.10
0.15
0
100
200
300
400

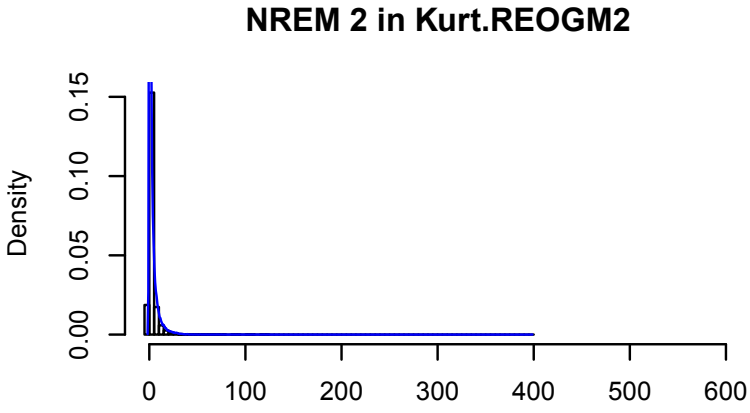

NREM 3 in Kurt.LEOGM2
Density
0.00
0.10
0.20
0.30
0
100
200
300
400

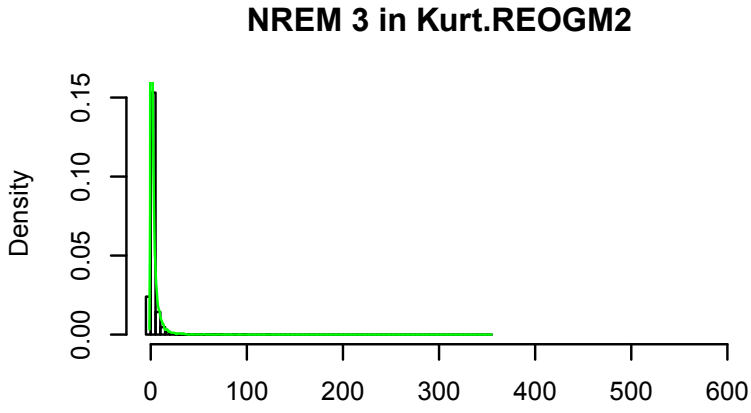

REM in Kurt.LEOGM2
Density
0.00
0.10
0.20
0.30
0
100
200
300
400

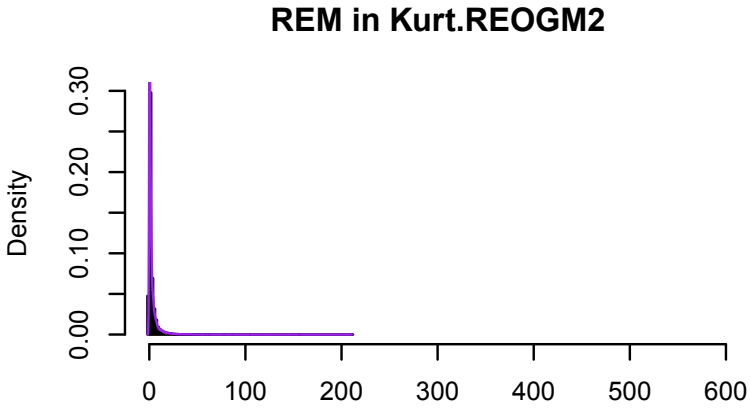

Wake in Kurt.LEOGM2
Density
0.00
0.05
0.10
0.15
0
100
200
300
400

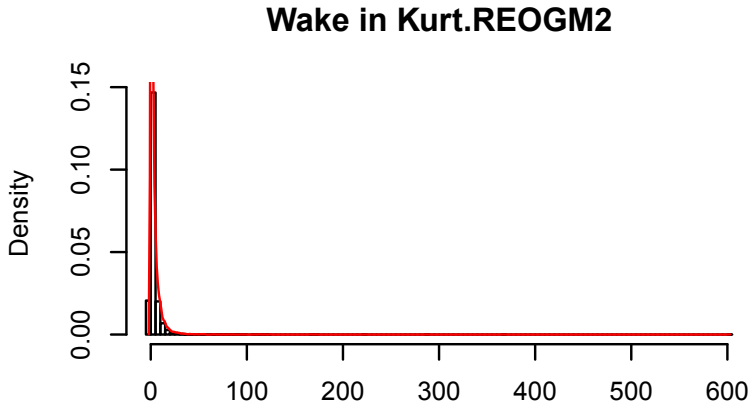

Kurt.REOGM2
Density
0.00
0.05
0.10
0.15
0
100
200
300
400
500
600

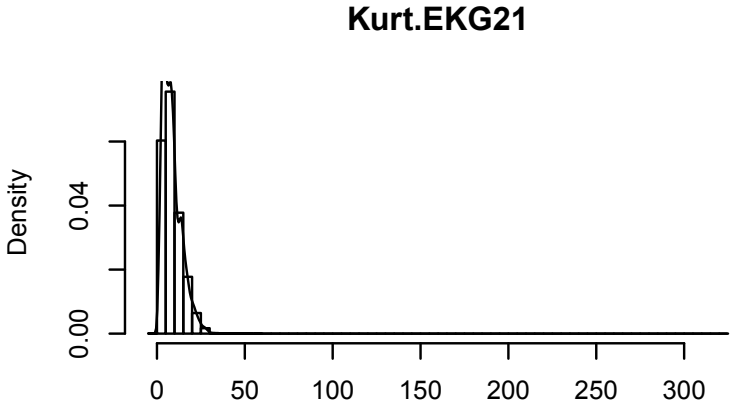

NREM 1 in Kurt.REOGM2
Density
0.0
0.1
0.2
0.3
0
100
200
300
400
500
600

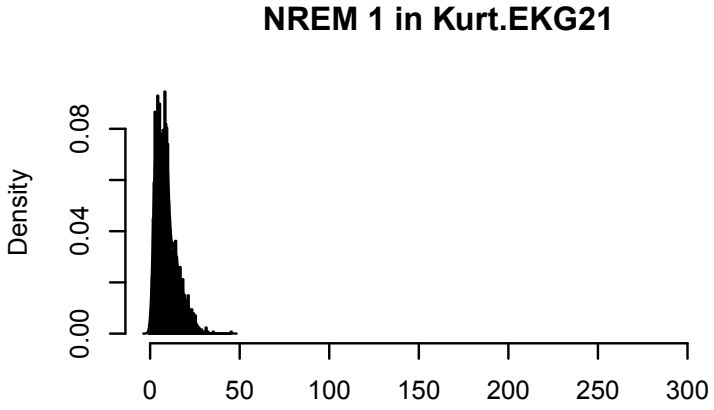

NREM 2 in Kurt.REOGM2
Density
0.00
0.05
0.10
0.15
0
100
200
300
400
500
600

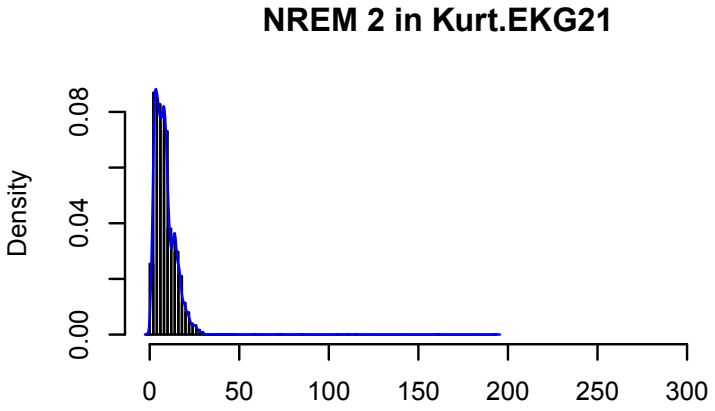

NREM 3 in Kurt.REOGM2
Density
0.00
0.05
0.10
0.15
0
100
200
300
400
500
600

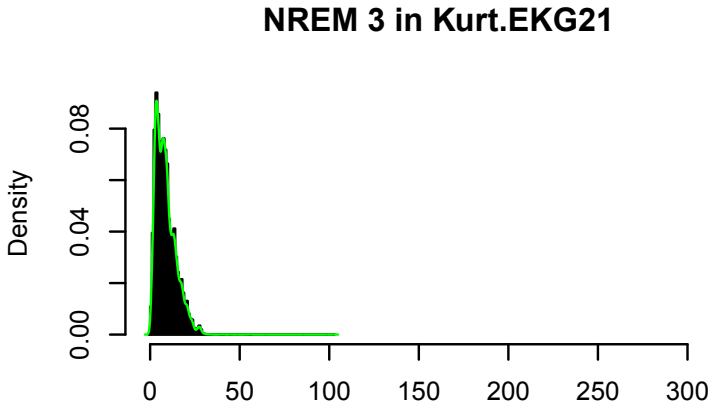

REM in Kurt.REOGM2
Density
0.00
0.10
0.20
0.30
0
100
200
300
400
500
600

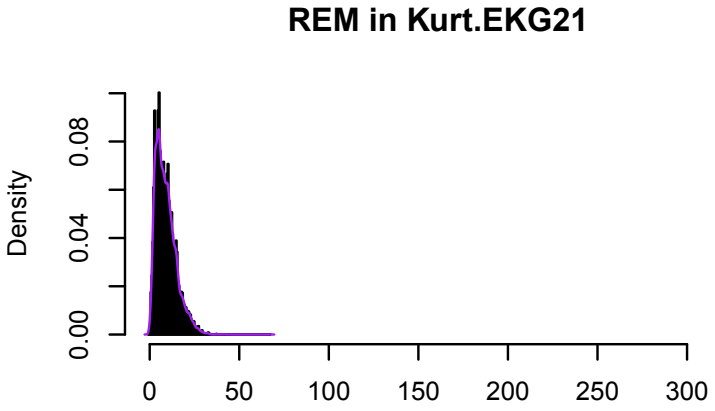

Wake in Kurt.REOGM2
Density
0.00
0.05
0.10
0.15
0
100
200
300
400
500
600

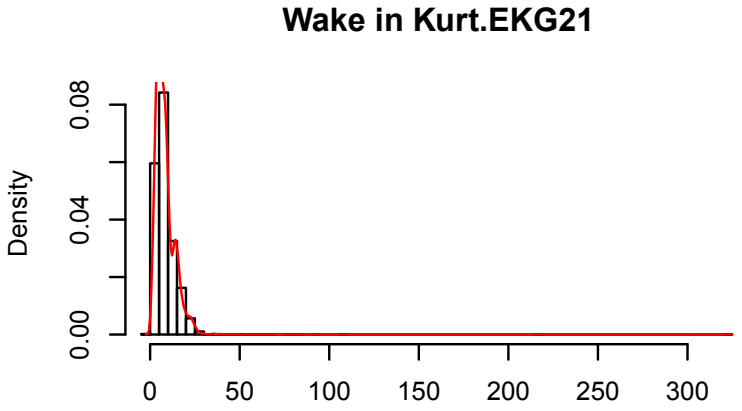

Kurt.EKG21
Density
0.00
0.04
0 50 100 150 200 250 300

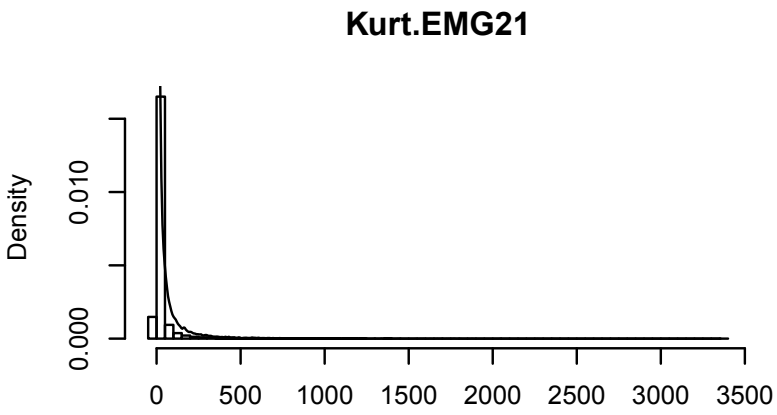

NREM 1 in Kurt.EKG21
Density
0.00
0.04
0.08
0 50 100 150 200 250 300

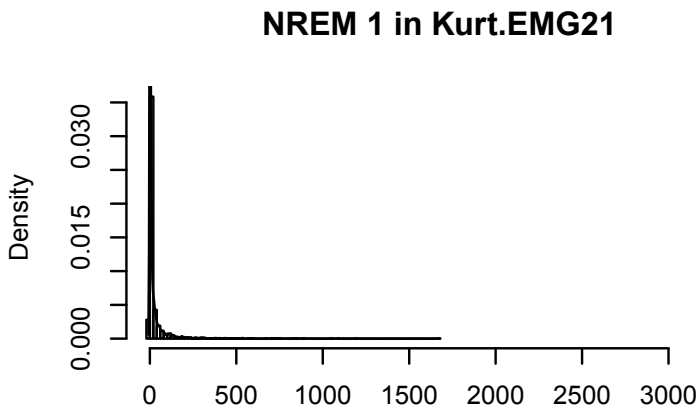

NREM 2 in Kurt.EKG21
Density
0.00
0.04
0.08
0 50 100 150 200 250 300

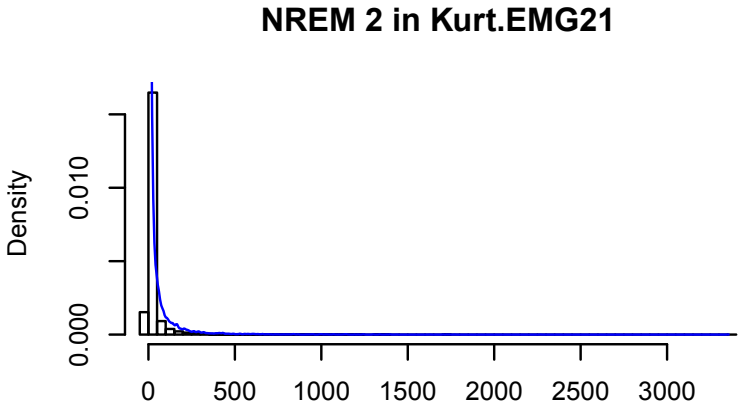

NREM 3 in Kurt.EKG21
Density
0.00
0.04
0.08
0 50 100 150 200 250 300

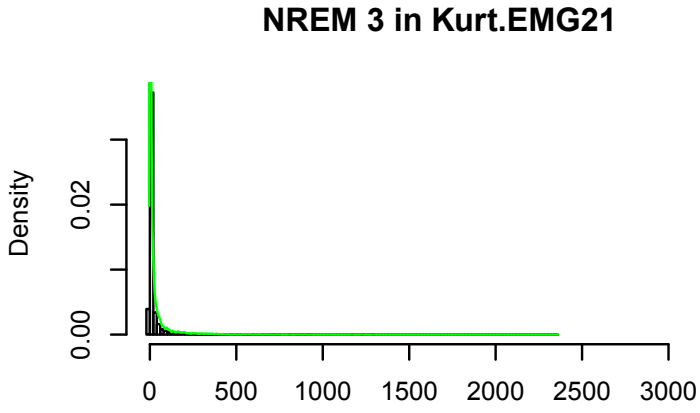

REM in Kurt.EKG21
Density
0.00
0.04
0.08
0 50 100 150 200 250 300

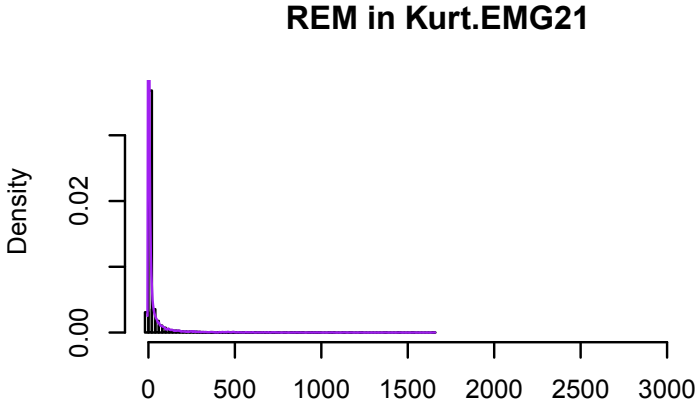

Wake in Kurt.EKG21
Density
0.00
0.04
0.08
0 50 100 150 200 250 300

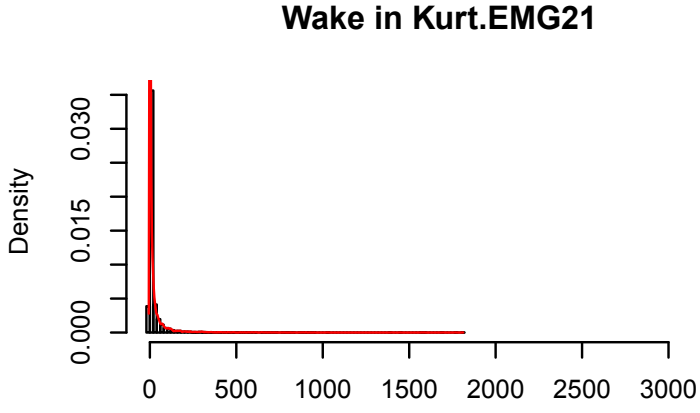

Kurt.EMG21
Density
0.000
0.010
0 500 1000 1500 2000 2500 3000 3500

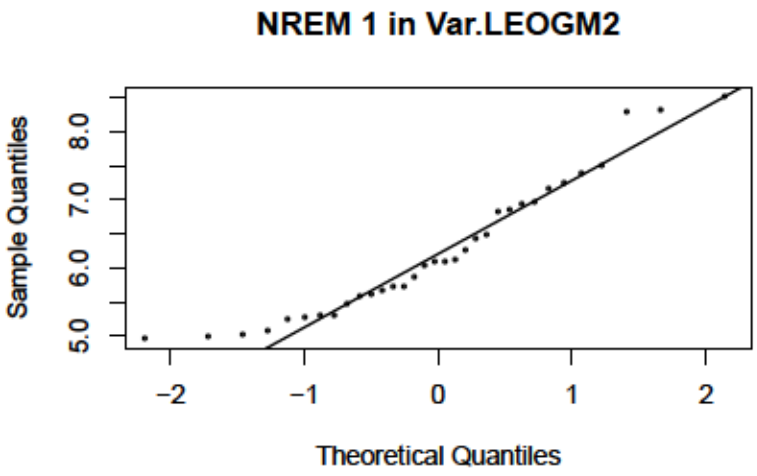

NREM 1 in Kurt.EMG21
Density
0.000
0.015
0.030
0 500 1000 1500 2000 2500 3000

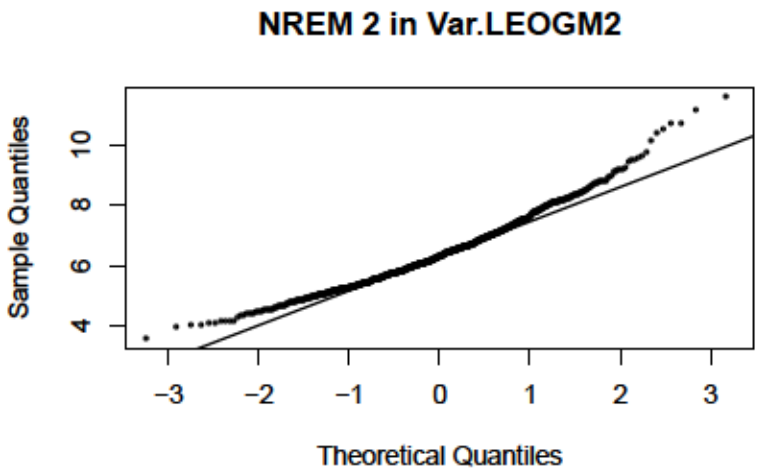

NREM 2 in Kurt.EMG21
Density
0.000
0.010
0 500 1000 1500 2000 2500 3000

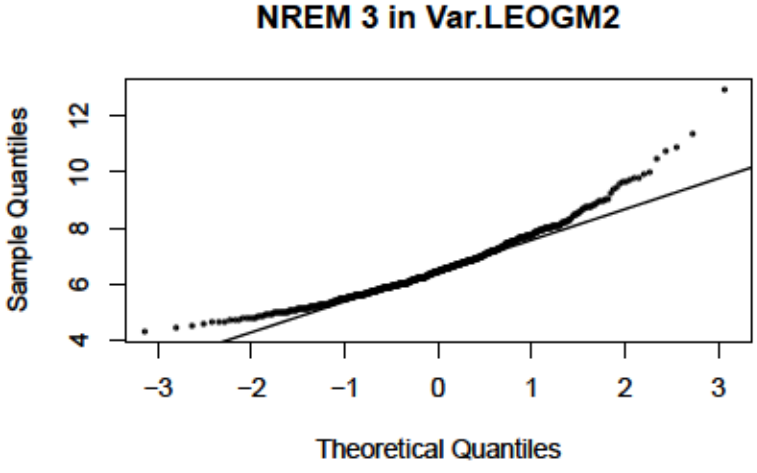

NREM 3 in Kurt.EMG21
Density
0.00
0.02
0 500 1000 1500 2000 2500 3000

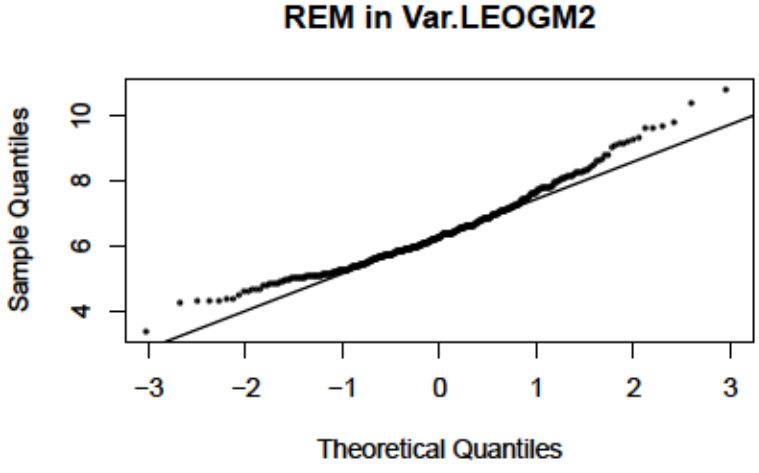

REM in Kurt.EMG21
Density
0.00
0.02
0 500 1000 1500 2000 2500 3000

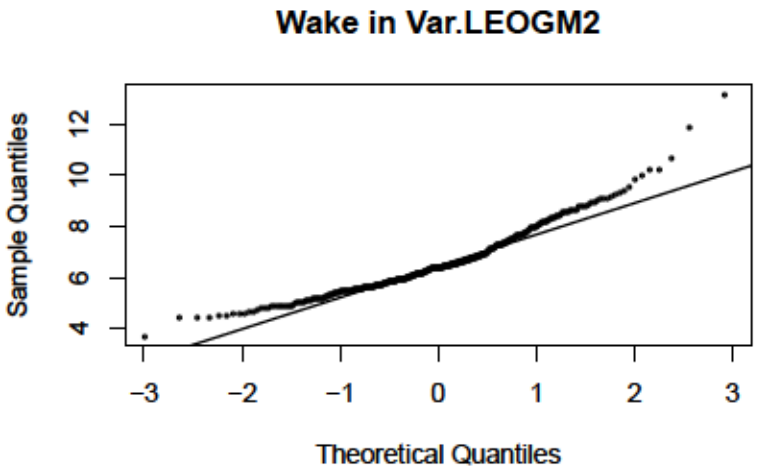

Wake in Kurt.EMG21
Density
0.000
0.015
0.030
0 500 1000 1500 2000 2500 3000

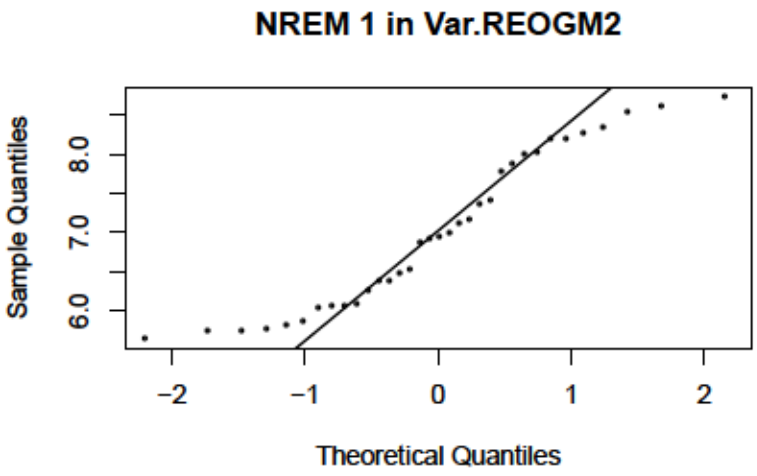

NREM 1 in Var.LEOGM2
Sample Quantiles
Theoretical Quantiles

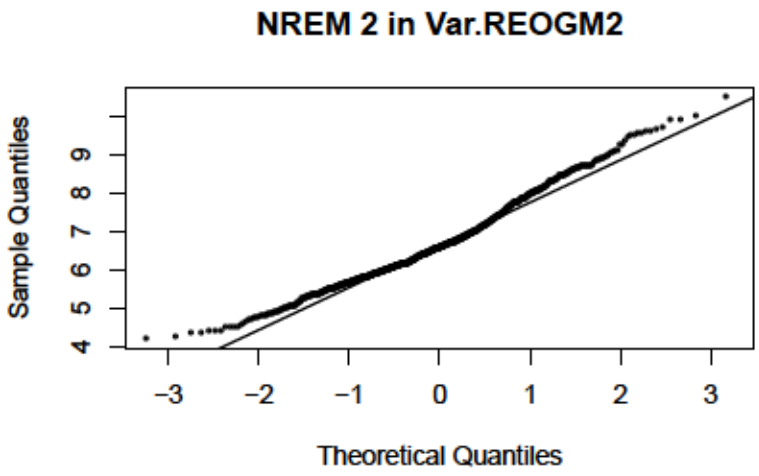

NREM 2 in Var.LEOGM2
Sample Quantiles
Theoretical Quantiles

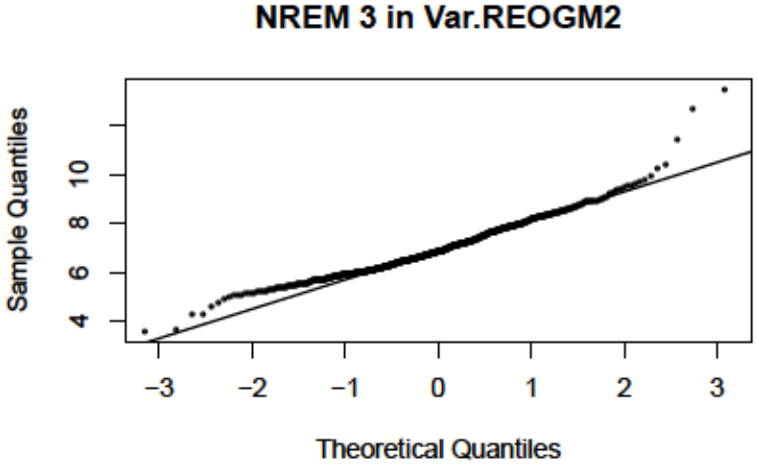

NREM 3 in Var.LEOGM2
Sample Quantiles
Theoretical Quantiles

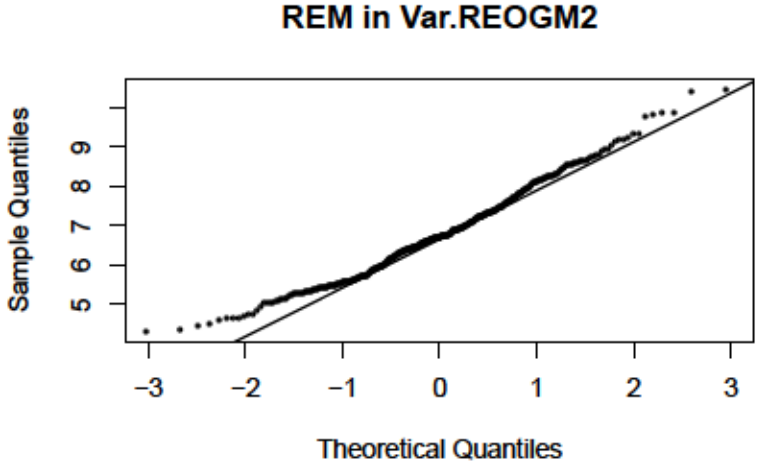

REM in Var.LEOGM2
Sample Quantiles
Theoretical Quantiles

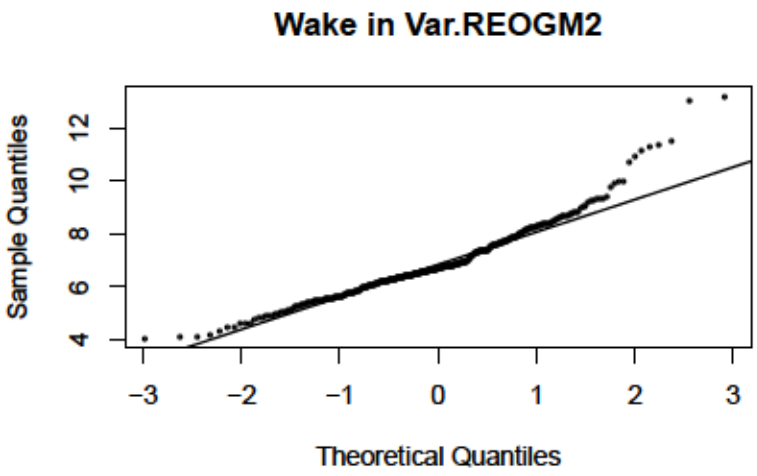

Wake in Var.LEOGM2
Sample Quantiles
Theoretical Quantiles

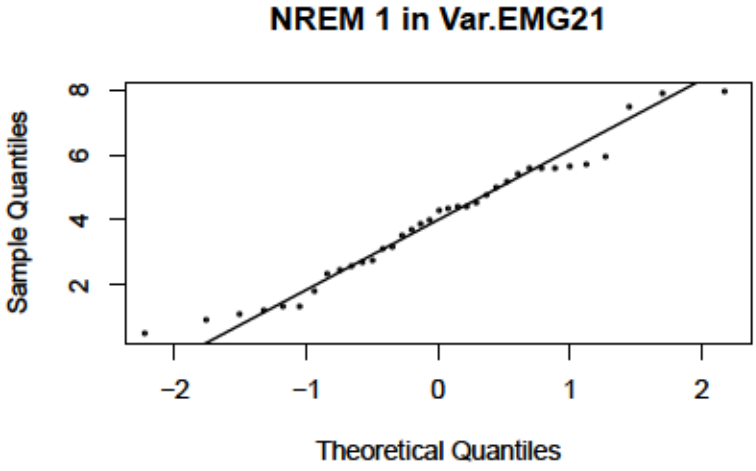

NREM 1 in Var.REOGM2
Sample Quantiles
6.0
7.0
8.0
−2
−1
0
1
2
Theoretical Quantiles

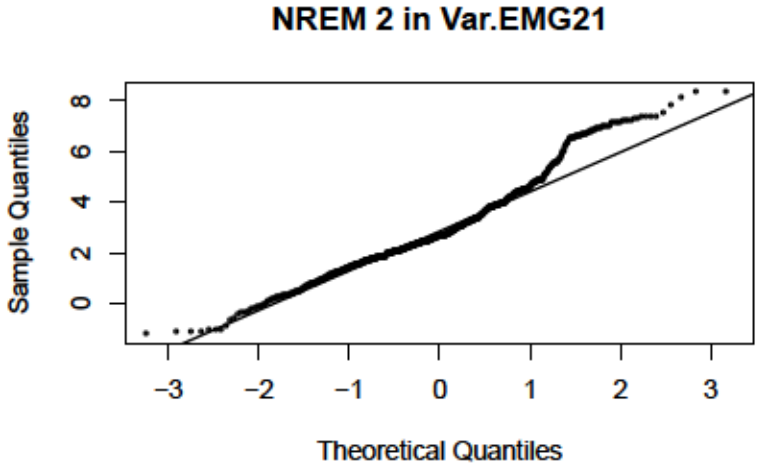

NREM 2 in Var.REOGM2
Sample Quantiles
4
5
6
7
8
9
−3
−2
−1
0
1
2
3
Theoretical Quantiles

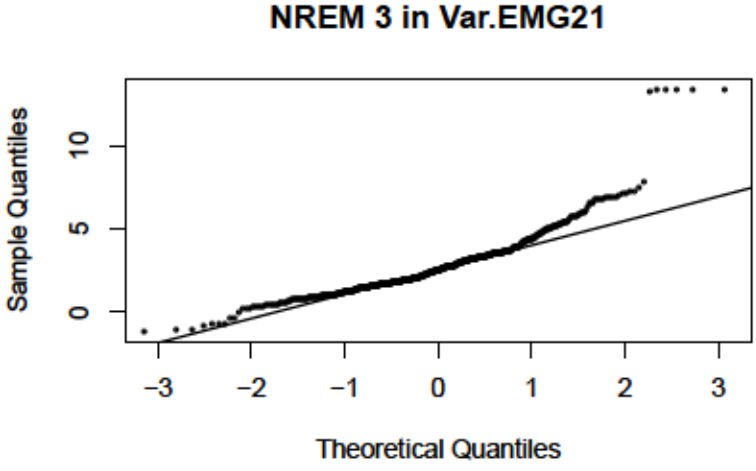

NREM 3 in Var.REOGM2
Sample Quantiles
4
6
8
10
−3
−2
−1
0
1
2
3
Theoretical Quantiles

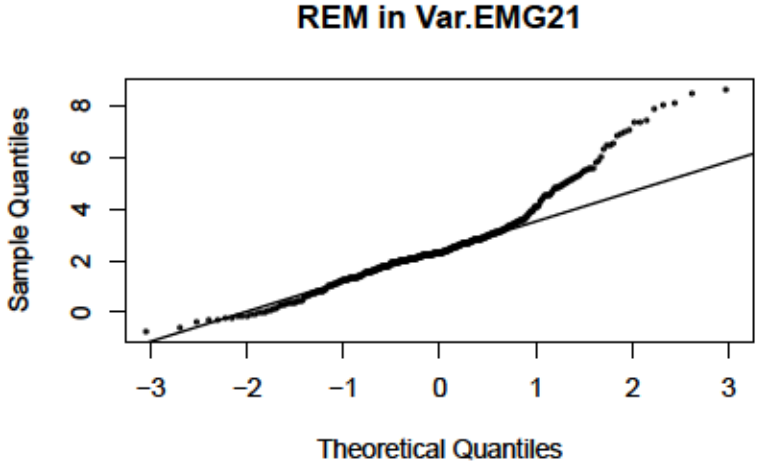

REM in Var.REOGM2
Sample Quantiles
5
6
7
8
9
−3
−2
−1
0
1
2
3
Theoretical Quantiles

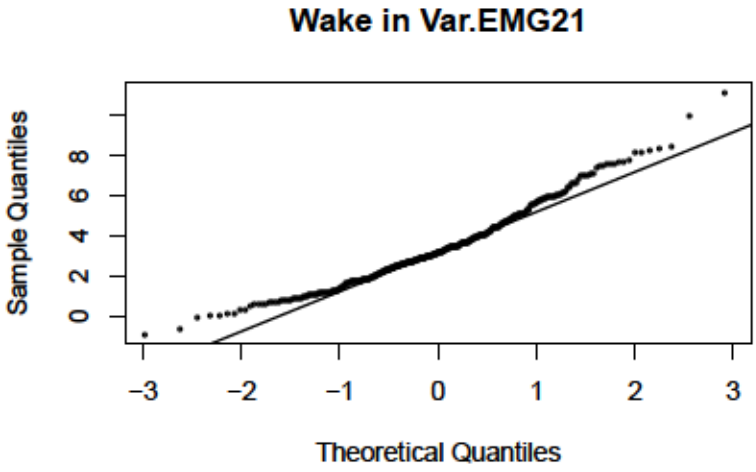

Wake in Var.REOGM2
Sample Quantiles
4
6
8
10
12
−3
−2
−1
0
1
2
3
Theoretical Quantiles

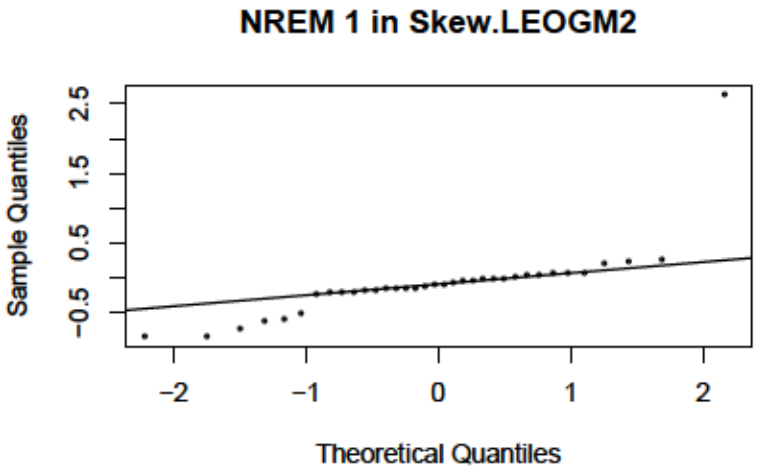

NREM 1 in Var.EMG21
Sample Quantiles
Theoretical Quantiles

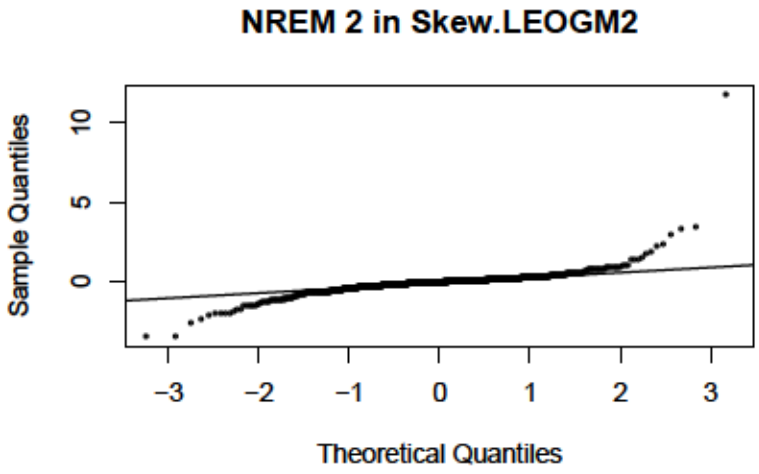

NREM 2 in Var.EMG21
Sample Quantiles
Theoretical Quantiles

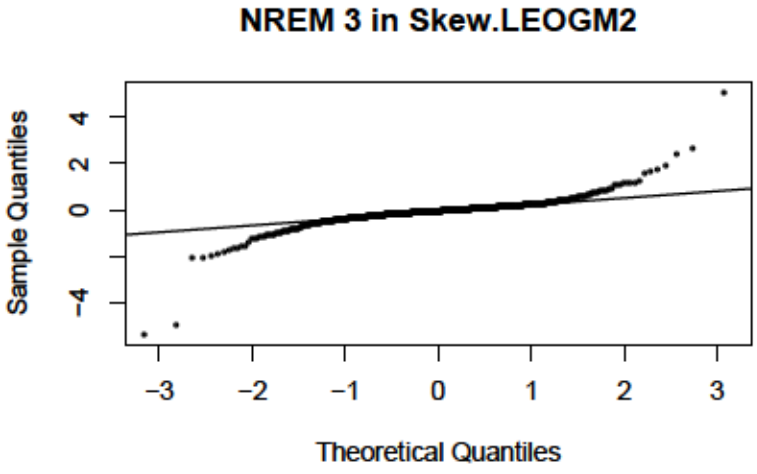

NREM 3 in Var.EMG21
Sample Quantiles
Theoretical Quantiles

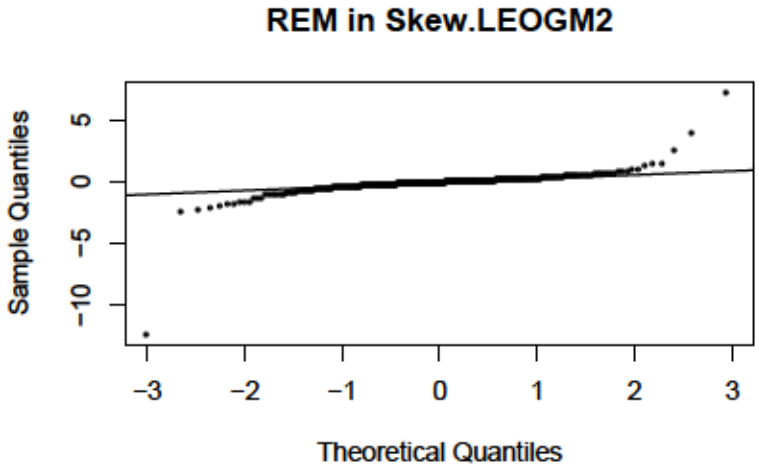

REM in Var.EMG21
Sample Quantiles
Theoretical Quantiles

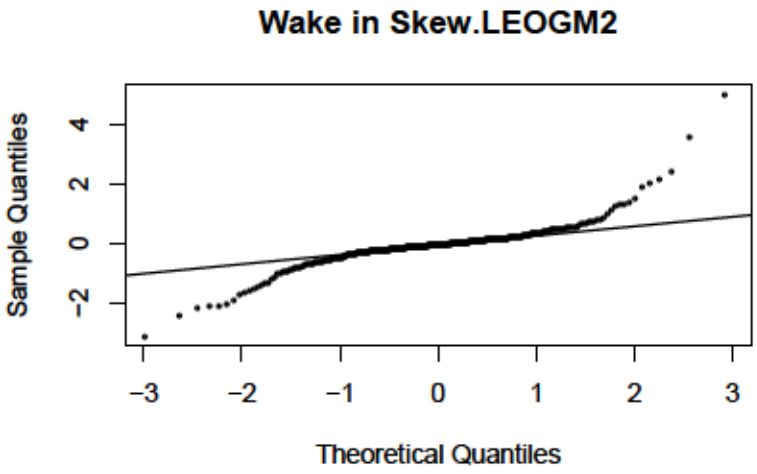

Wake in Var.EMG21
Sample Quantiles
Theoretical Quantiles

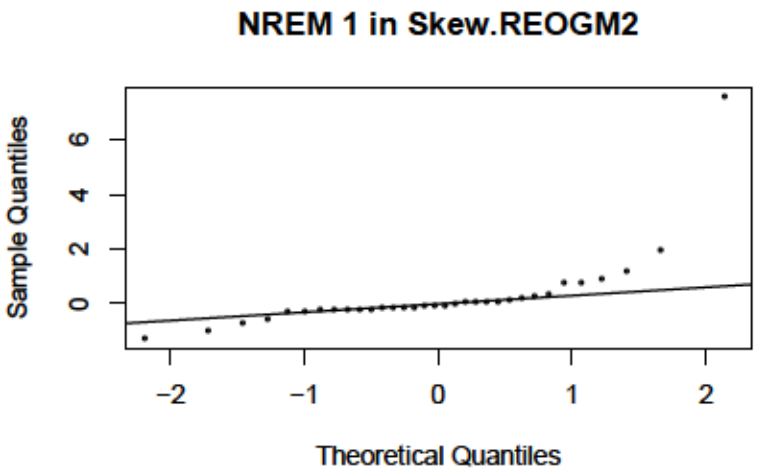

NREM 1 in Skew.LEOGM2
Sample Quantiles
Theoretical Quantiles

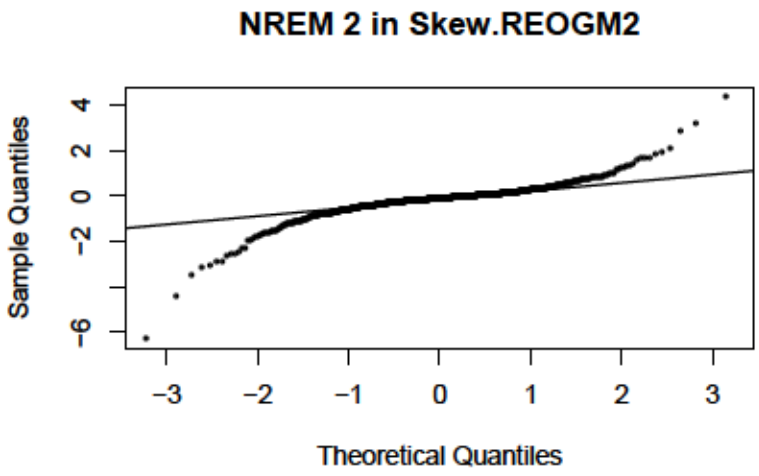

NREM 2 in Skew.LEOGM2
Sample Quantiles
Theoretical Quantiles

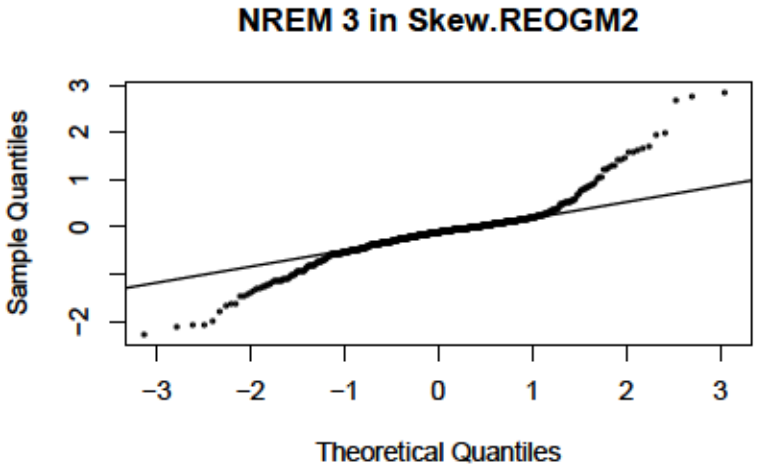

NREM 3 in Skew.LEOGM2
Sample Quantiles
Theoretical Quantiles

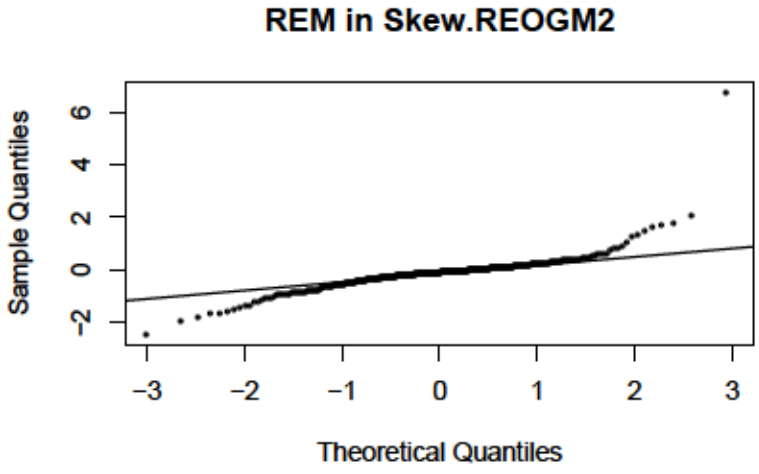

REM in Skew.LEOGM2
Sample Quantiles
Theoretical Quantiles

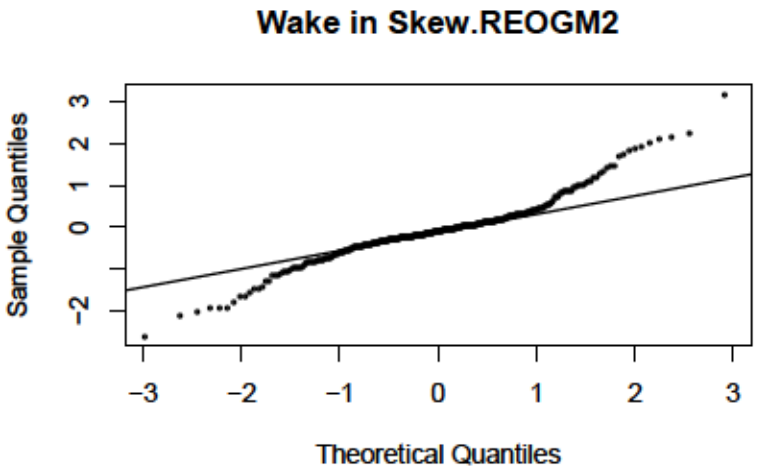

Wake in Skew.LEOGM2
Sample Quantiles
Theoretical Quantiles

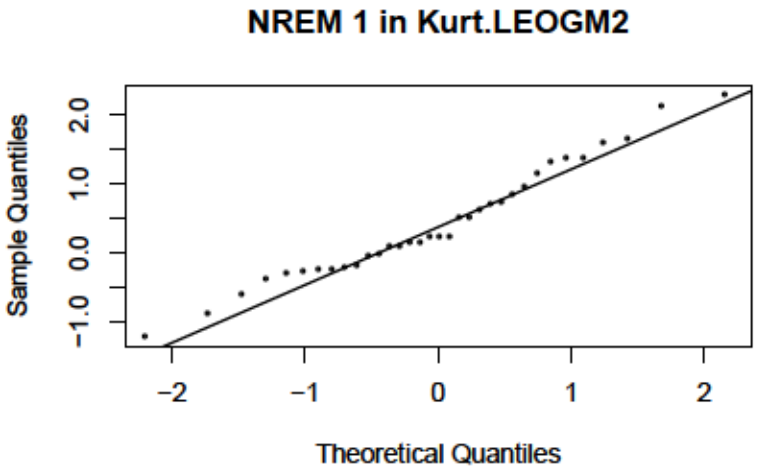

NREM 1 in Skew.REOGM2
Sample Quantiles
Theoretical Quantiles

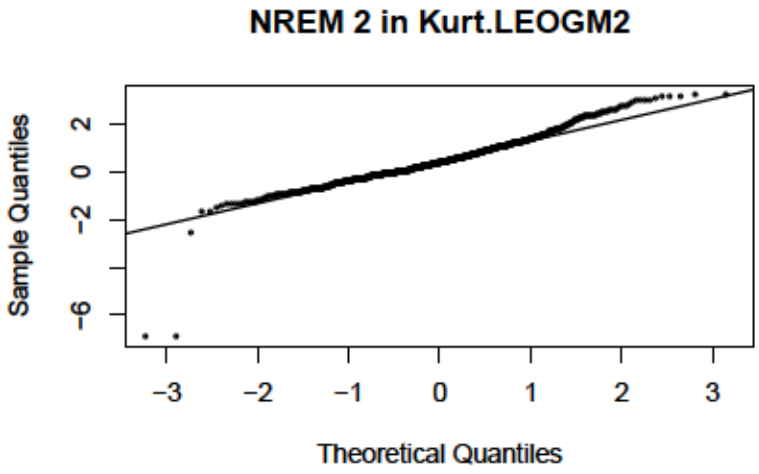

NREM 2 in Skew.REOGM2
Sample Quantiles
Theoretical Quantiles

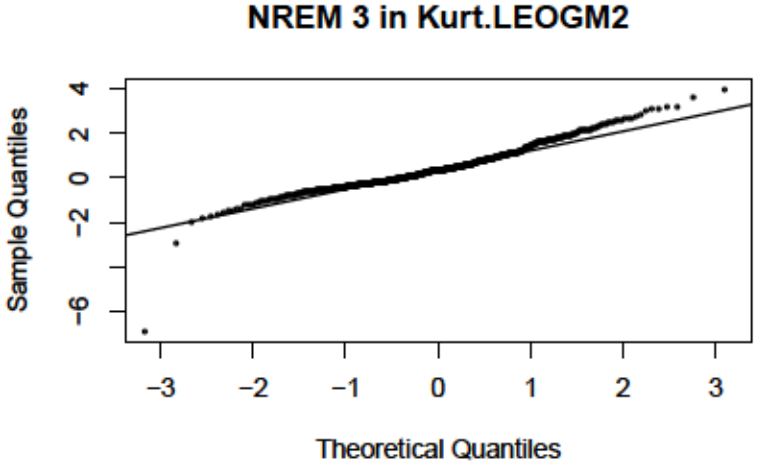

NREM 3 in Skew.REOGM2
Sample Quantiles
Theoretical Quantiles

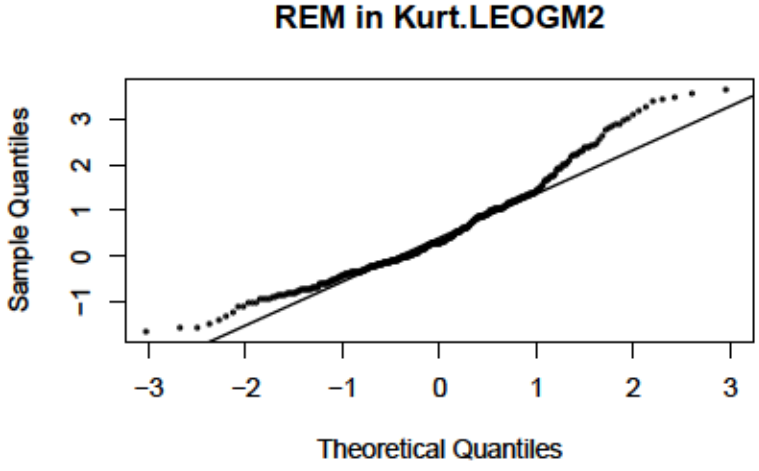

REM in Skew.REOGM2
Sample Quantiles
Theoretical Quantiles

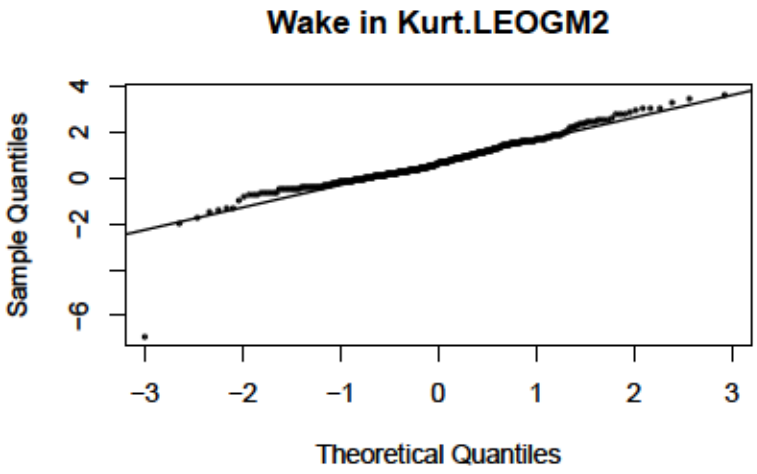

Wake in Skew.REOGM2
Sample Quantiles
Theoretical Quantiles

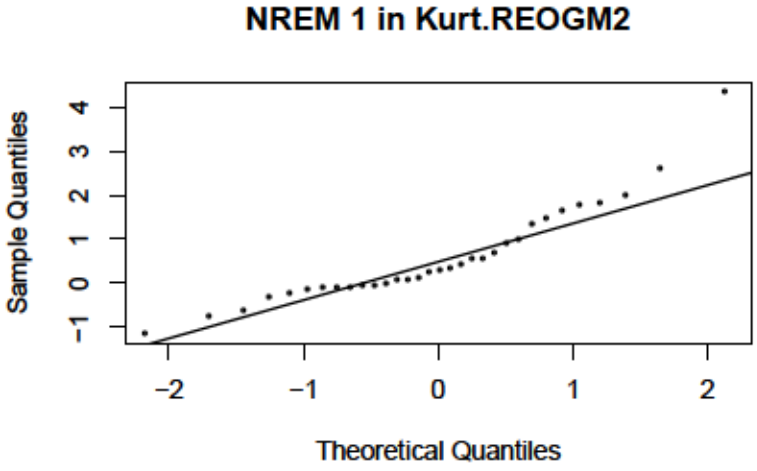

NREM 1 in Kurt.LEOGM2
Sample Quantiles
Theoretical Quantiles

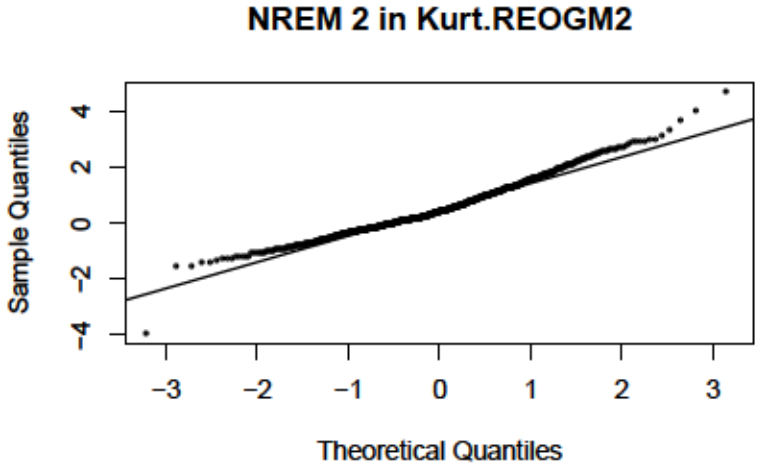

NREM 2 in Kurt.LEOGM2
Sample Quantiles
Theoretical Quantiles

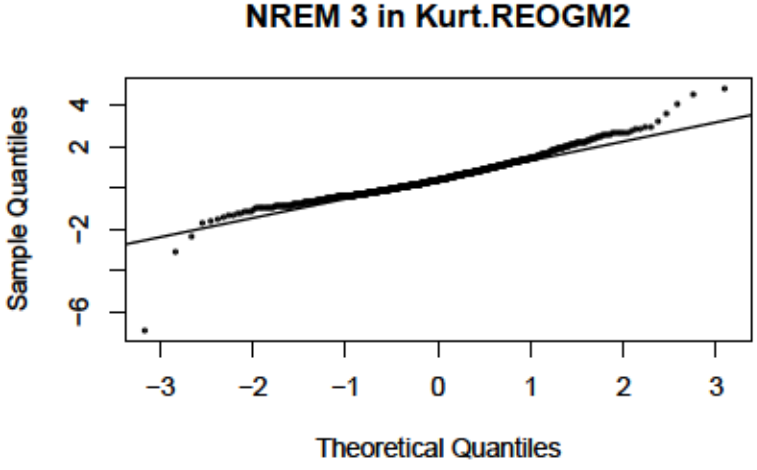

NREM 3 in Kurt.LEOGM2
Sample Quantiles
Theoretical Quantiles

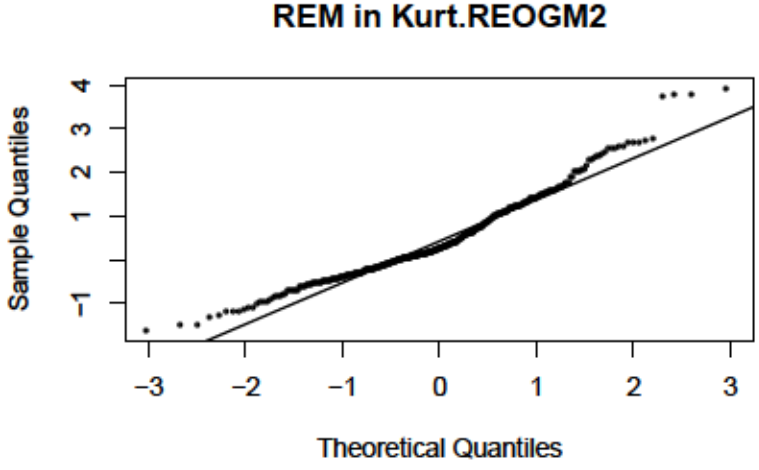

REM in Kurt.LEOGM2
Sample Quantiles
Theoretical Quantiles

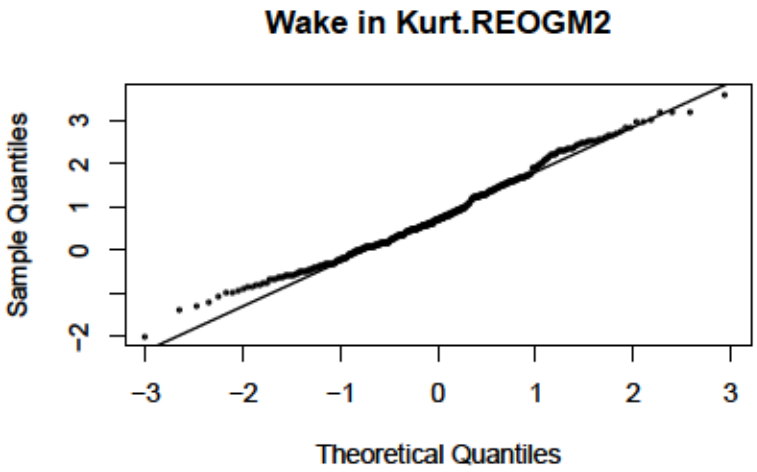

Wake in Kurt.LEOGM2
Sample Quantiles
Theoretical Quantiles

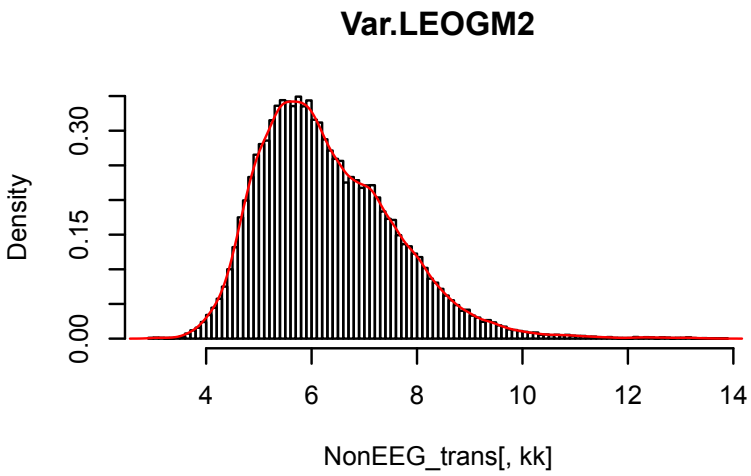

NREM 1 in Kurt.REOGM2
Sample Quantiles
Theoretical Quantiles

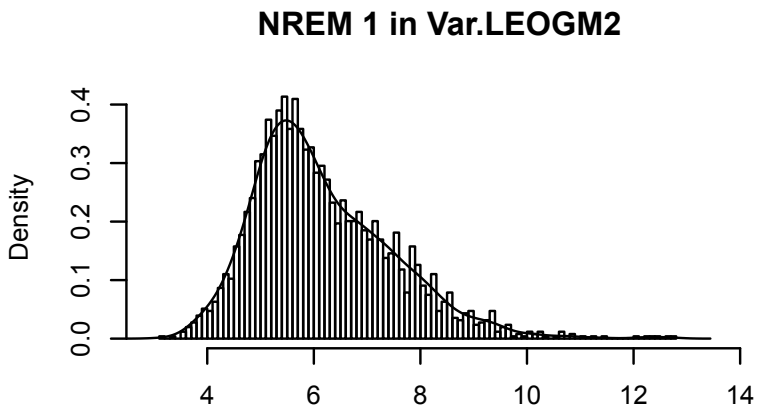

NREM 2 in Kurt.REOGM2
Sample Quantiles
Theoretical Quantiles

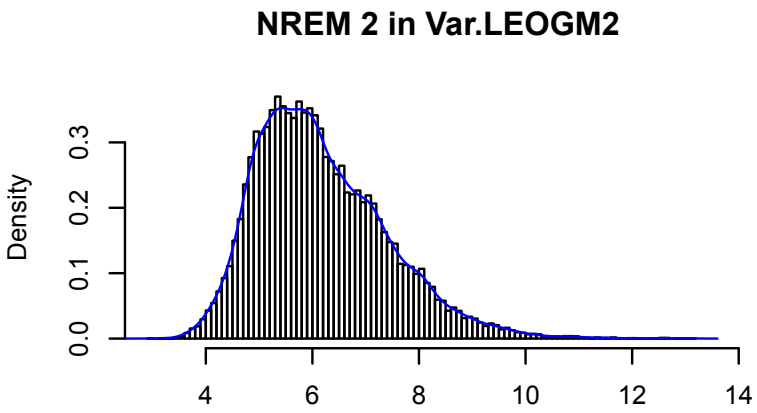

NREM 3 in Kurt.REOGM2
Sample Quantiles
Theoretical Quantiles

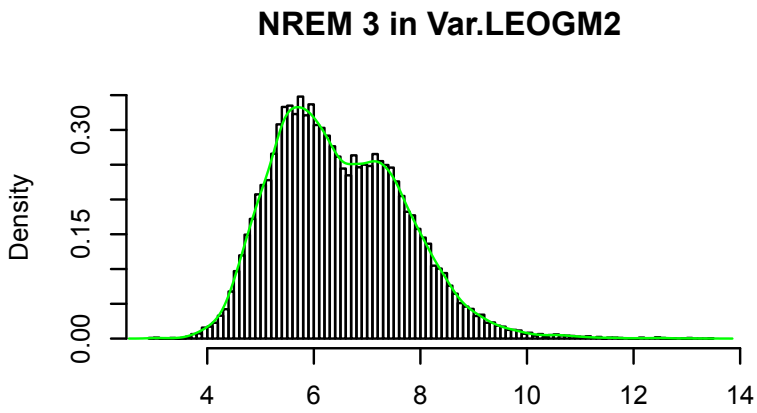

REM in Kurt.REOGM2
Sample Quantiles
Theoretical Quantiles

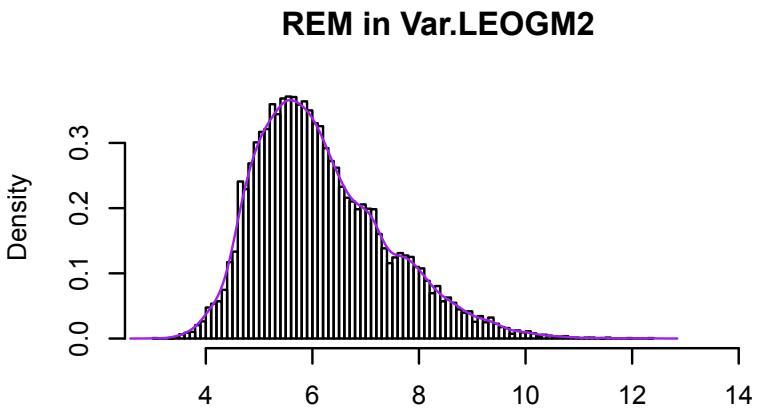

Wake in Kurt.REOGM2
Sample Quantiles
Theoretical Quantiles

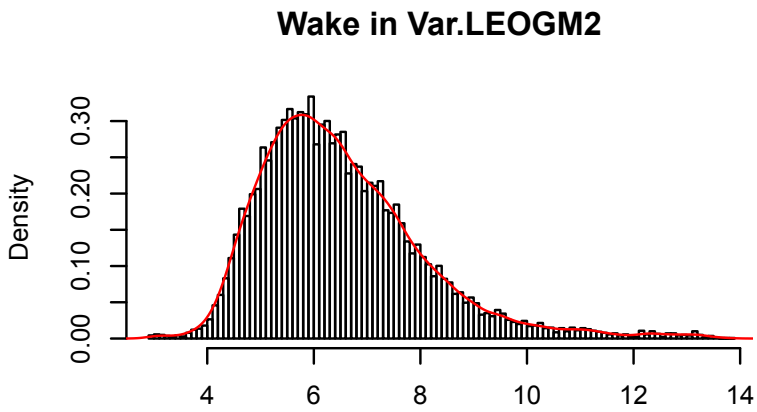

Var.LEOGM2
Density
0.00
0.15
0.30
4
6
8
10
12
14
NonEEG_trans[, kk]

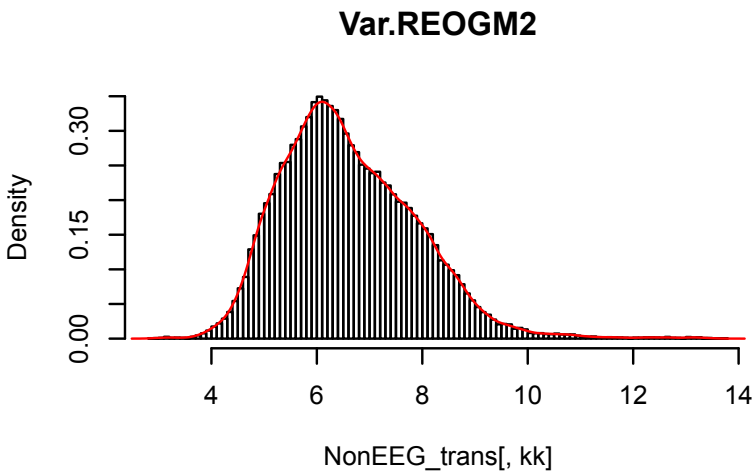

NREM 1 in Var.LEOGM2
Density
0.0
0.1
0.2
0.3
0.4
4
6
8
10
12
14

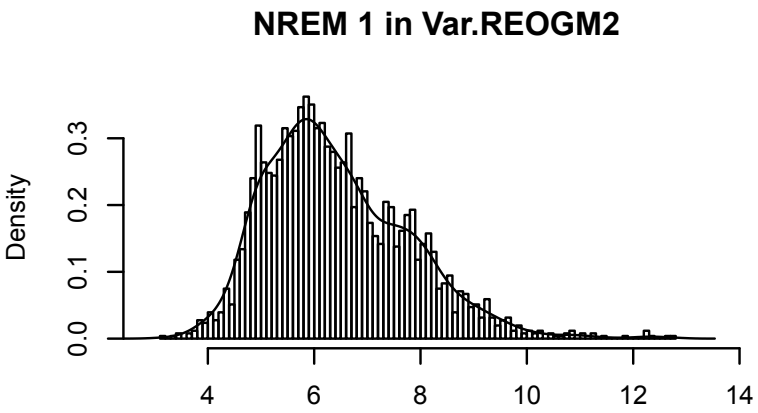

NREM 2 in Var.LEOGM2
Density
0.0
0.1
0.2
0.3
4
6
8
10
12
14

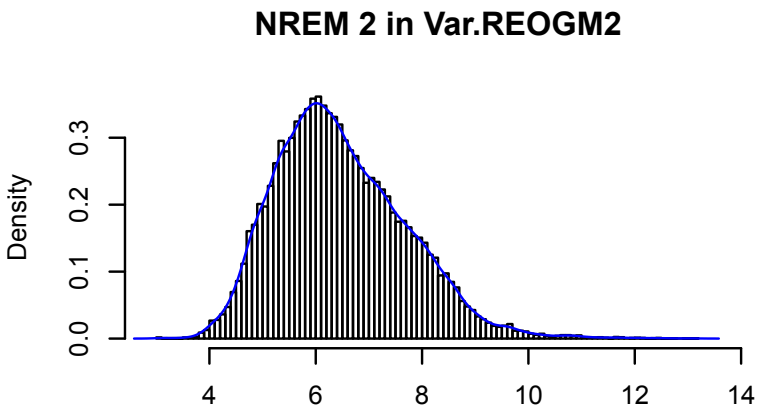

NREM 3 in Var.LEOGM2
Density
0.00
0.15
0.30
4
6
8
10
12
14

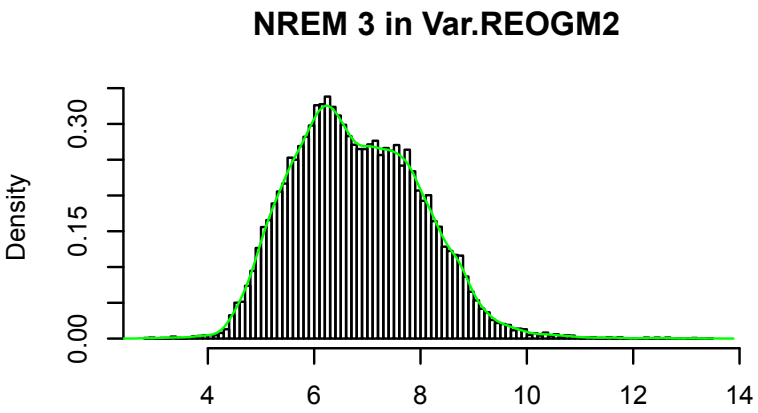

REM in Var.LEOGM2
Density
0.0
0.1
0.2
0.3
4
6
8
10
12
14

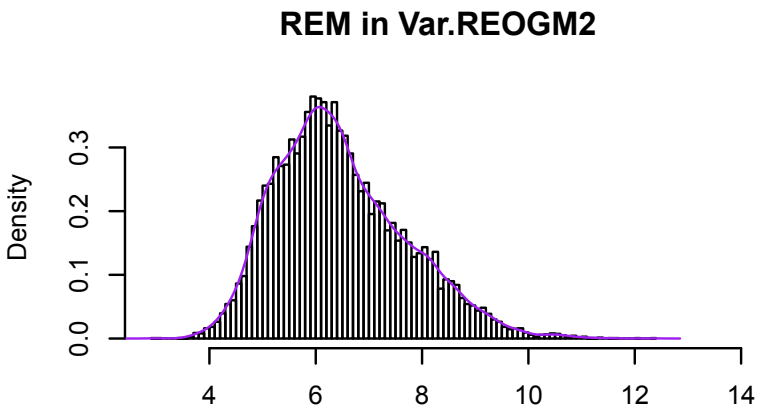

Wake in Var.LEOGM2
Density
0.00
0.10
0.20
0.30
4
6
8
10
12
14

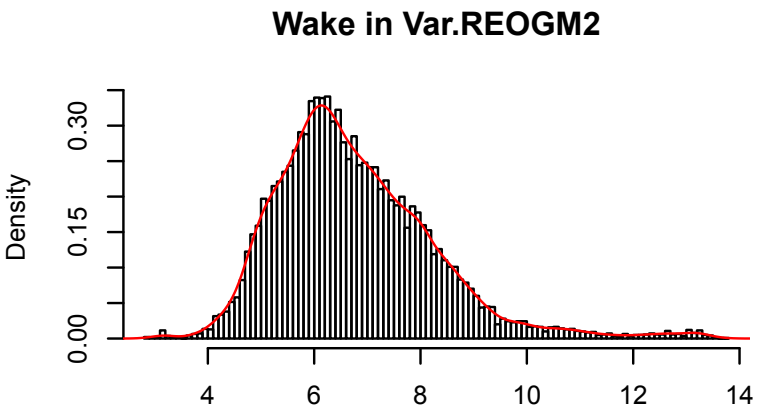

Var.REOGM2
Density
0.00
0.15
0.30
4
6
8
10
12
14
NonEEG_trans[, kk]

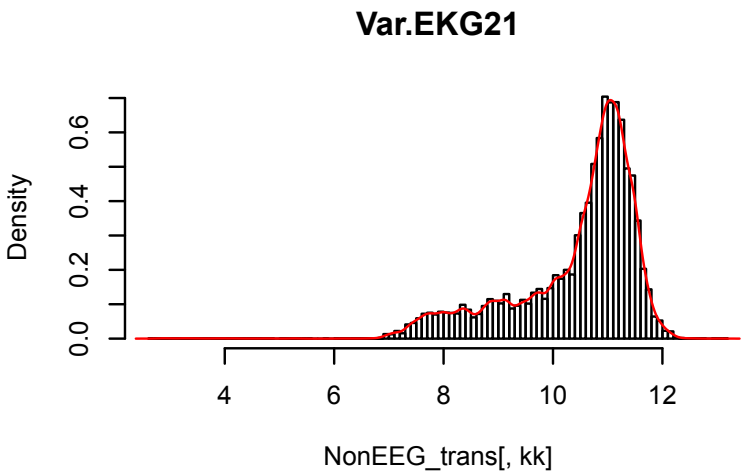

NREM 1 in Var.REOGM2
Density
0.0
0.1
0.2
0.3
4
6
8
10
12
14

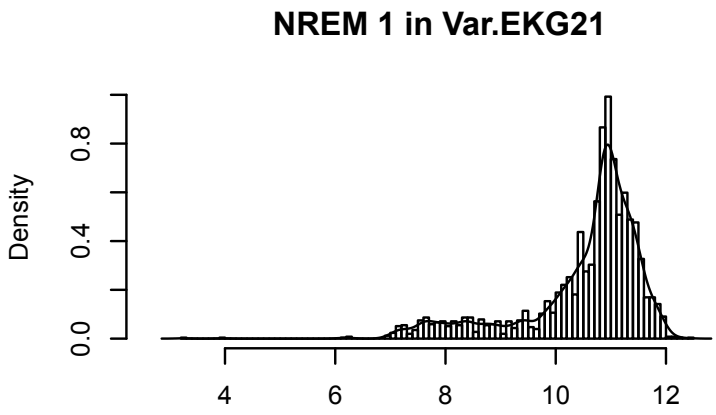

NREM 2 in Var.REOGM2
Density
0.0
0.1
0.2
0.3
4
6
8
10
12
14

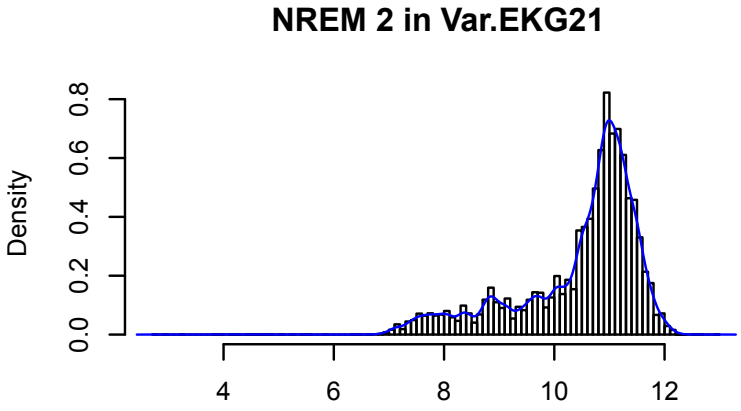

NREM 3 in Var.REOGM2
Density
0.00
0.15
0.30
4
6
8
10
12
14

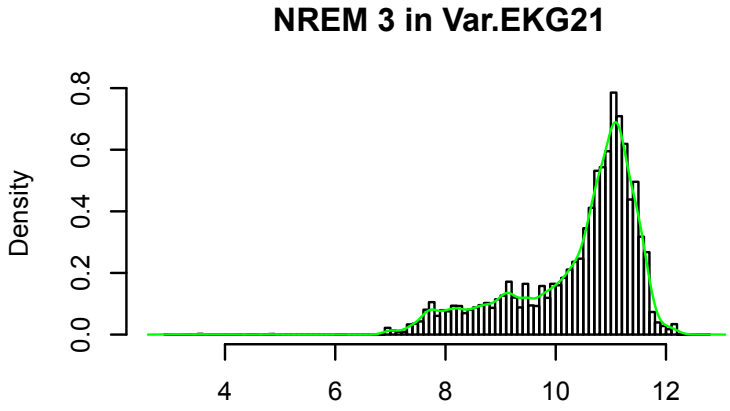

REM in Var.REOGM2
Density
0.0
0.1
0.2
0.3
4
6
8
10
12
14

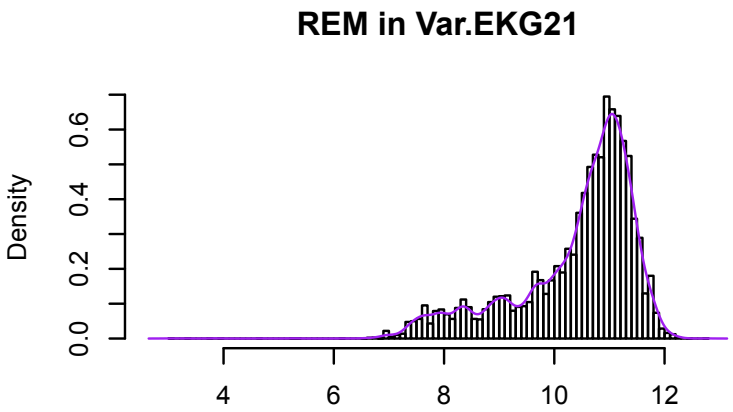

Wake in Var.REOGM2
Density
0.00
0.15
0.30
4
6
8
10
12
14

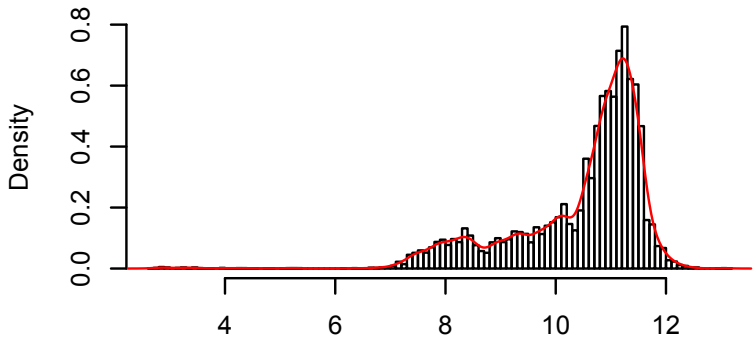

Var.EKG21
Density
0.0
0.2
0.4
0.6
4
6
8
10
12
NonEEG_trans[, kk]

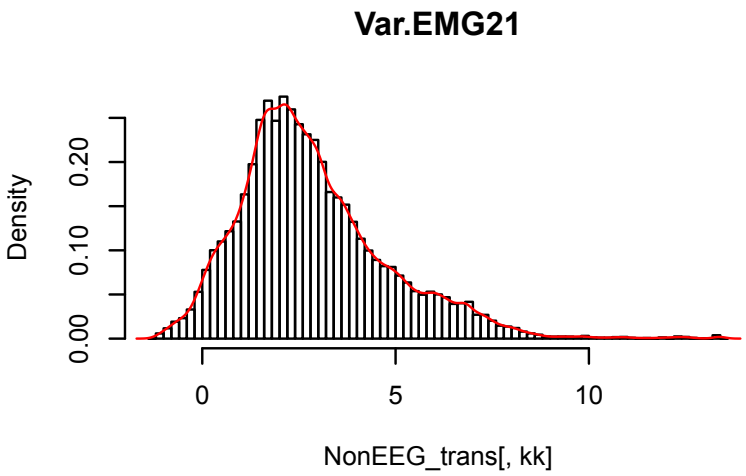

NREM 1 in Var.EKG21
Density
0.0
0.4
0.8
4
6
8
10
12

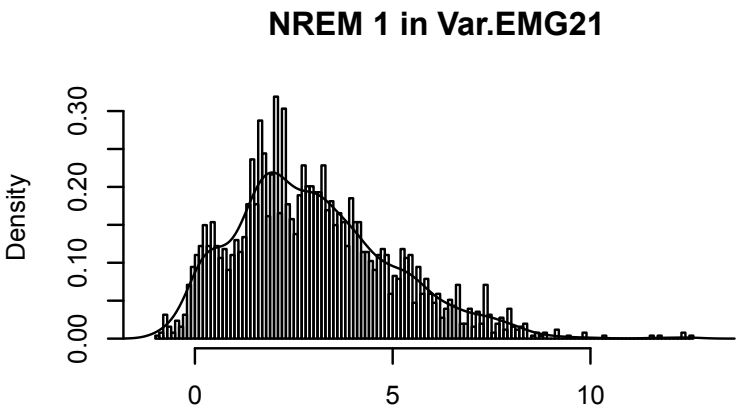

NREM 2 in Var.EKG21
Density
0.0
0.2
0.4
0.6
0.8
4
6
8
10
12

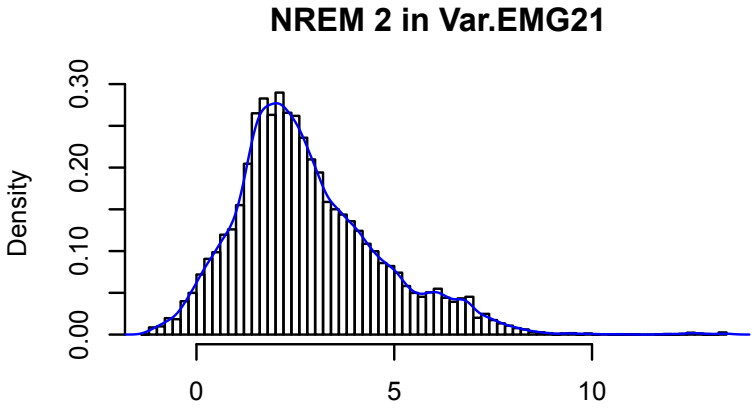

NREM 3 in Var.EKG21
Density
0.0
0.2
0.4
0.6
0.8
4
6
8
10
12

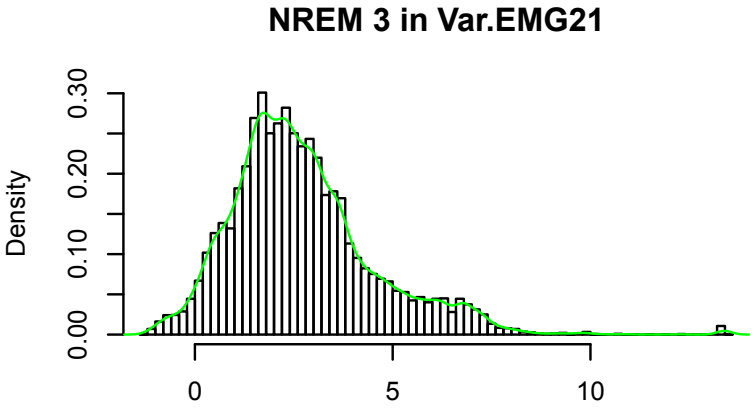

REM in Var.EKG21
Density
0.0
0.2
0.4
0.6
4
6
8
10
12

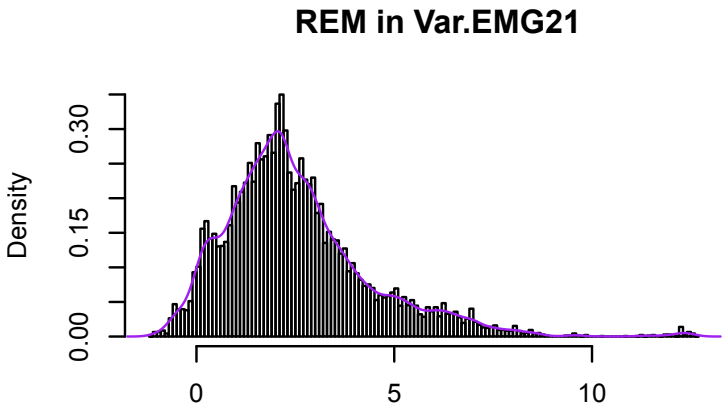

Wake in Var.EKG21
Density
0.0
0.2
0.4
0.6
0.8
4
6
8
10
12

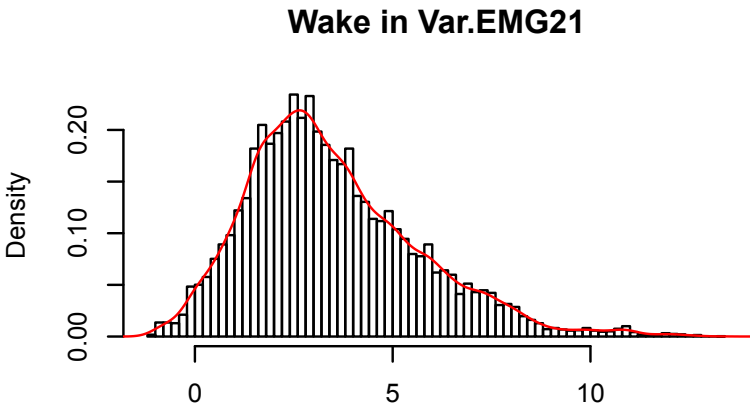

Var.EMG21
Density
0.00
0.10
0.20
0
5
10
NonEEG_trans[, kk]

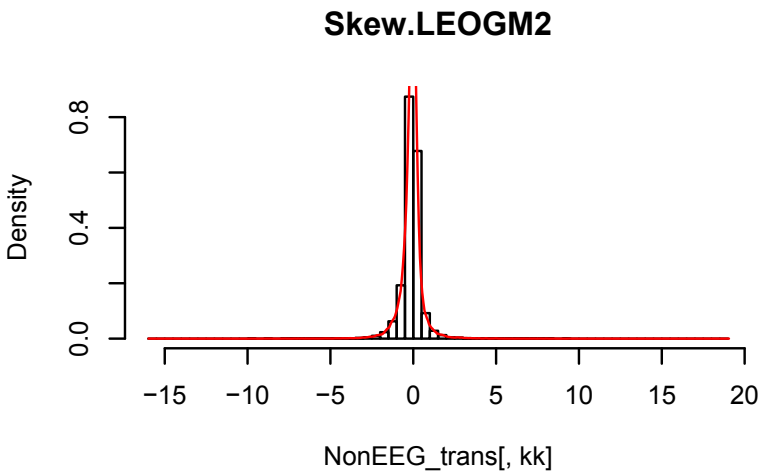

NREM 1 in Var.EMG21
Density
0.00
0.10
0.20
0.30
0
5
10

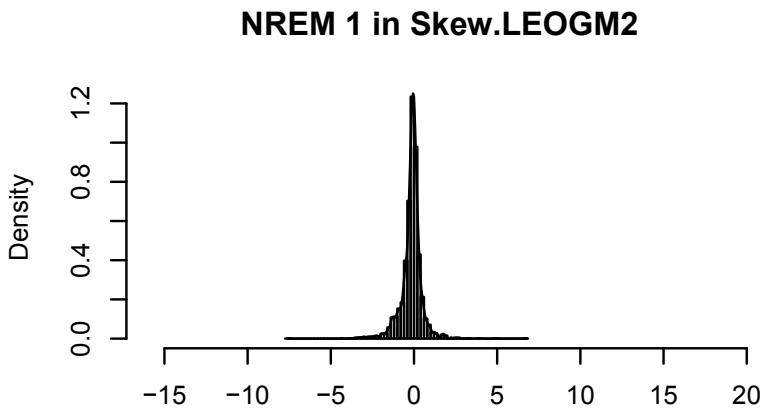

NREM 2 in Var.EMG21
Density
0.00
0.10
0.20
0.30
0
5
10

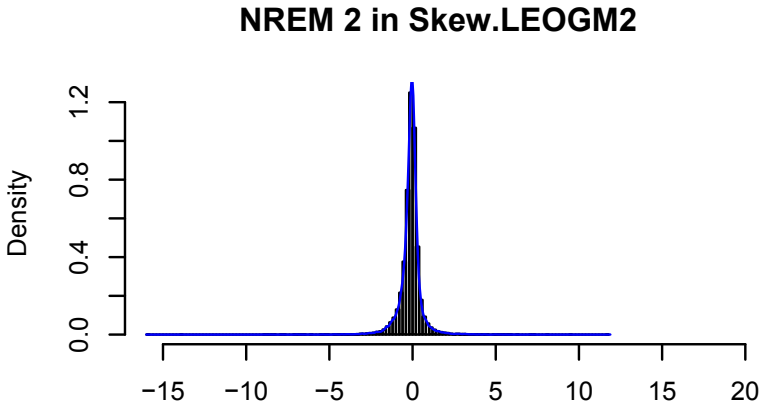

NREM 3 in Var.EMG21
Density
0.00
0.10
0.20
0.30
0
5
10

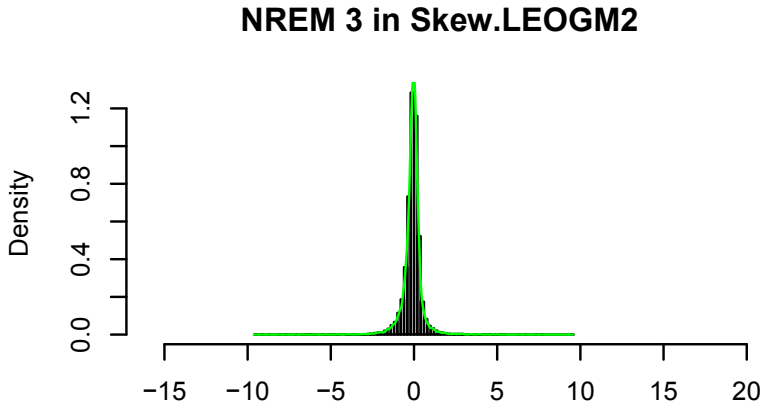

REM in Var.EMG21
Density
0.00
0.15
0.30
0
5
10

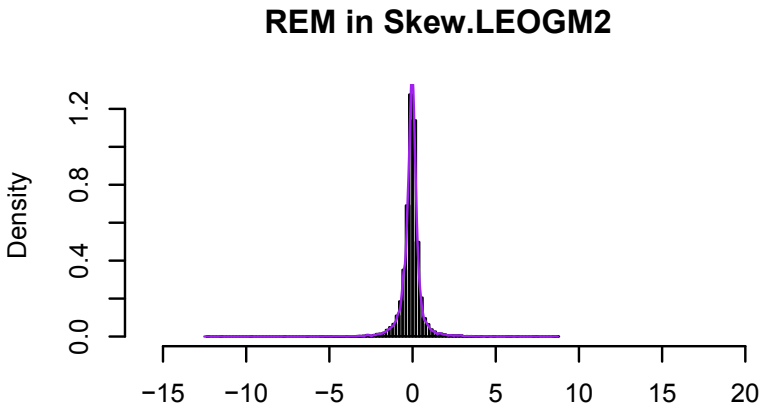

Wake in Var.EMG21
Density
0.00
0.10
0.20
0
5
10

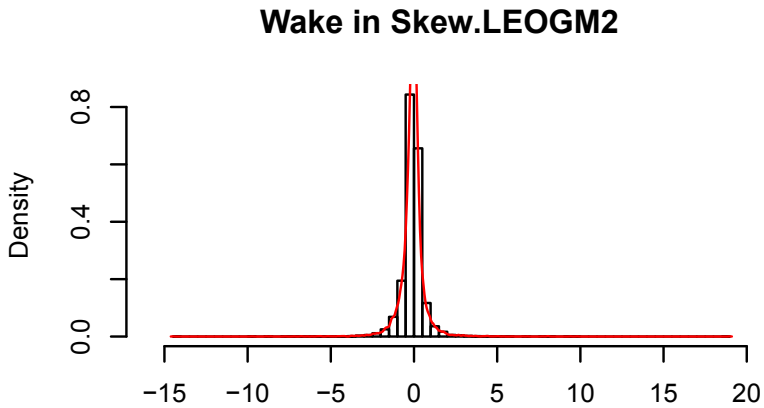

Skew.LEOGM2
Density
0.0
0.4
0.8
−15
−10
−5
0
5
10
15
20
NonEEG_trans[, kk]

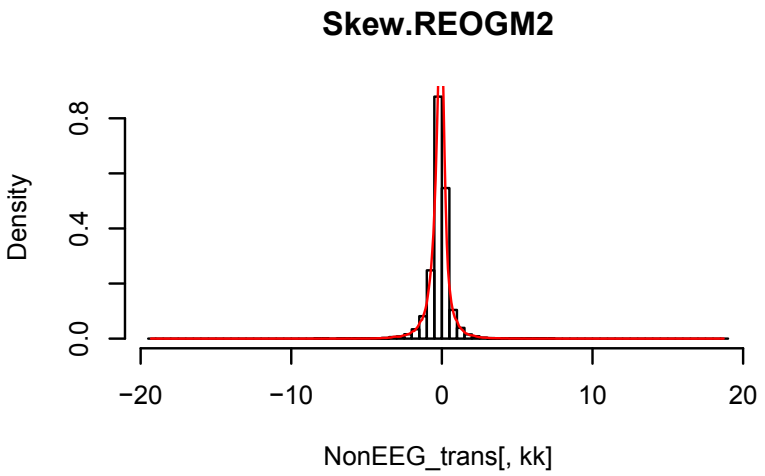

NREM 1 in Skew.LEOGM2
Density
0.0
0.4
0.8
1.2
−15
−10
−5
0
5
10
15
20

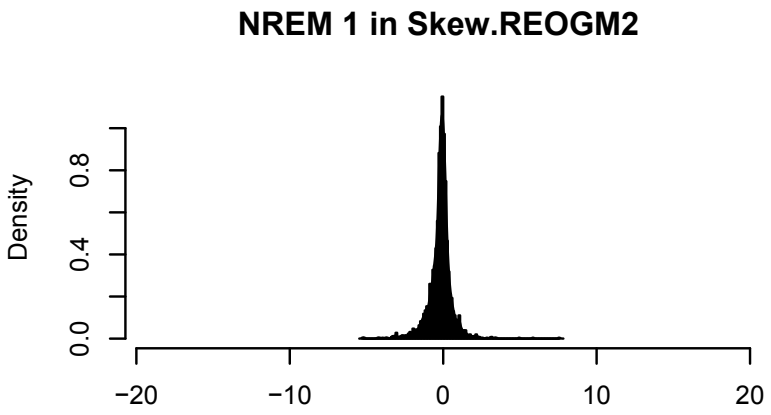

NREM 2 in Skew.LEOGM2
Density
0.0
0.4
0.8
1.2
−15
−10
−5
0
5
10
15
20

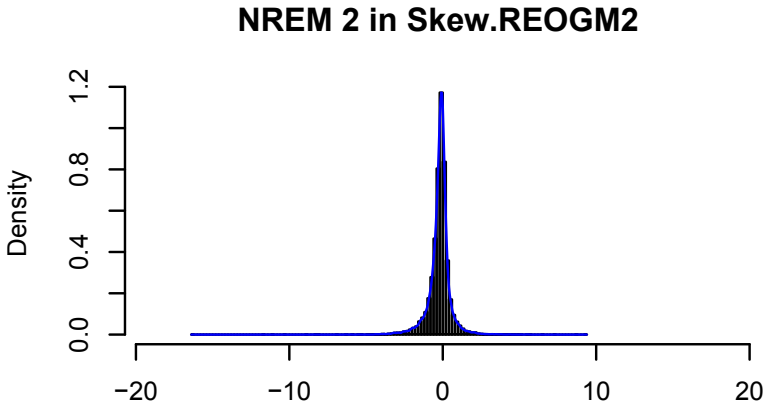

NREM 3 in Skew.LEOGM2
Density
0.0
0.4
0.8
1.2
−15
−10
−5
0
5
10
15
20

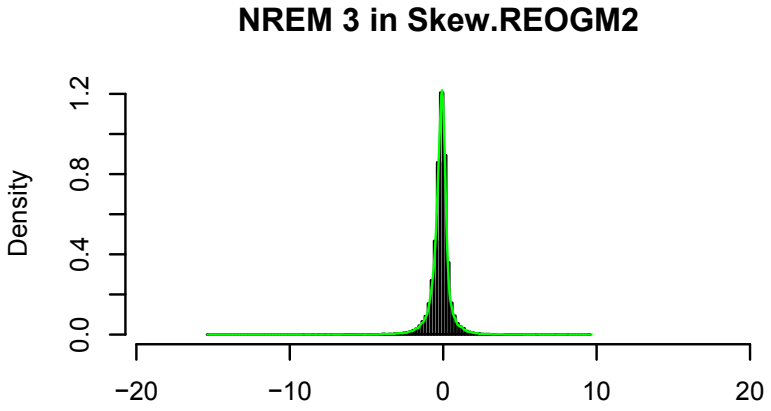

REM in Skew.LEOGM2
Density
0.0
0.4
0.8
1.2
−15
−10
−5
0
5
10
15
20

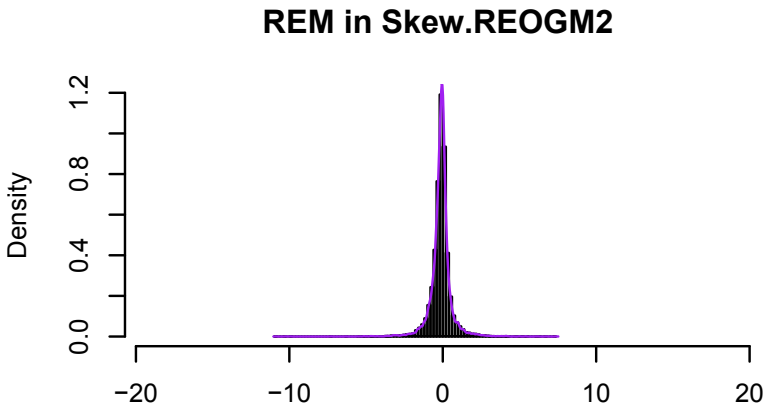

Wake in Skew.LEOGM2
Density
0.0
0.4
0.8
−15
−10
−5
0
5
10
15
20

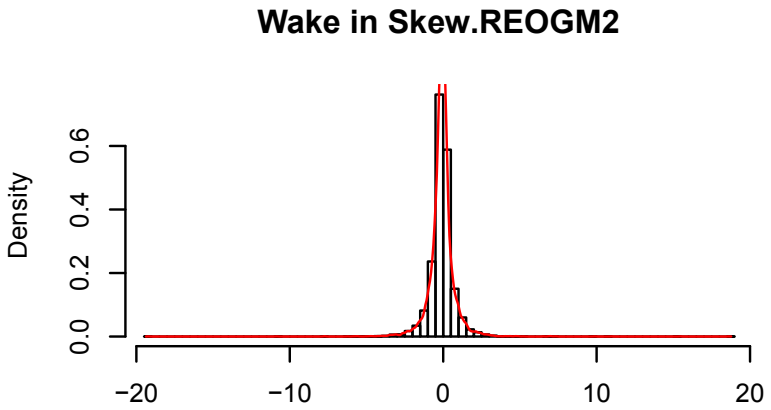

Skew.REOGM2
Density
0.0
0.4
0.8
−20
−10
0
10
20
NonEEG_trans[, kk]

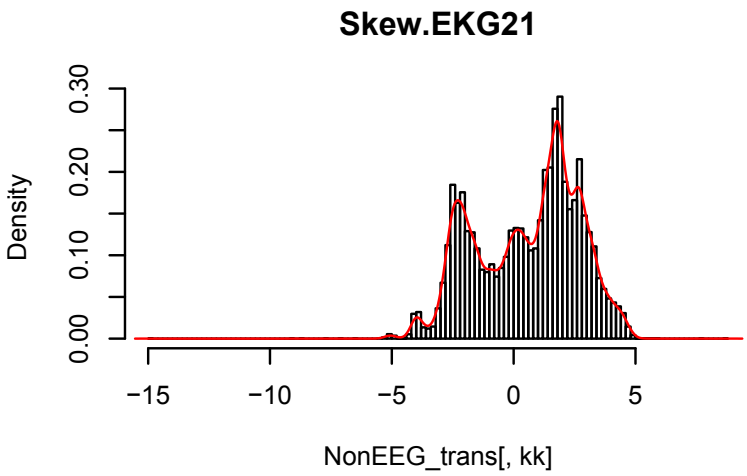

NREM 1 in Skew.REOGM2
Density
0.0
0.4
0.8
−20
−10
0
10
20

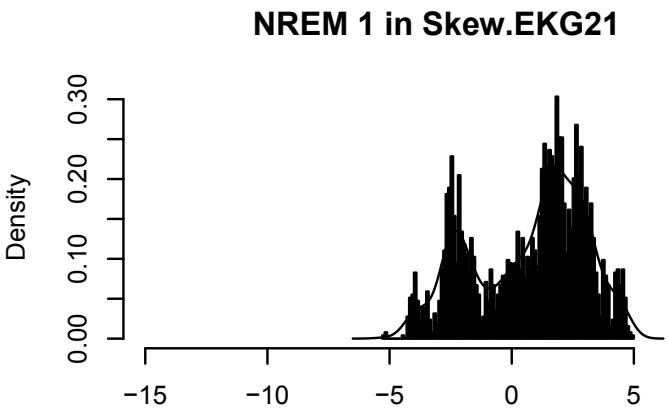

NREM 2 in Skew.REOGM2
Density
0.0
0.4
0.8
1.2
−20
−10
0
10
20

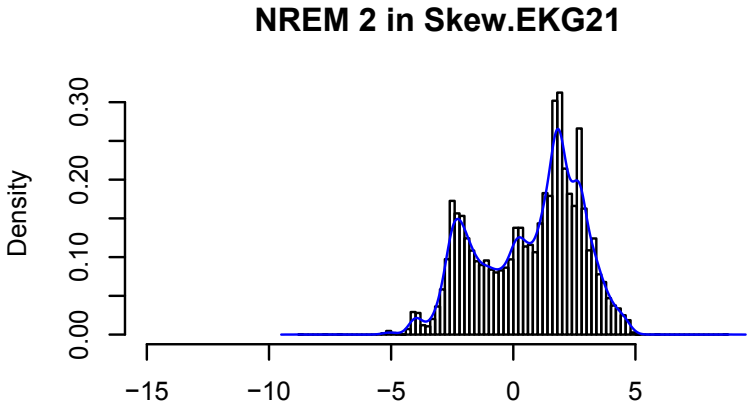

NREM 3 in Skew.REOGM2
Density
0.0
0.4
0.8
1.2
−20
−10
0
10
20

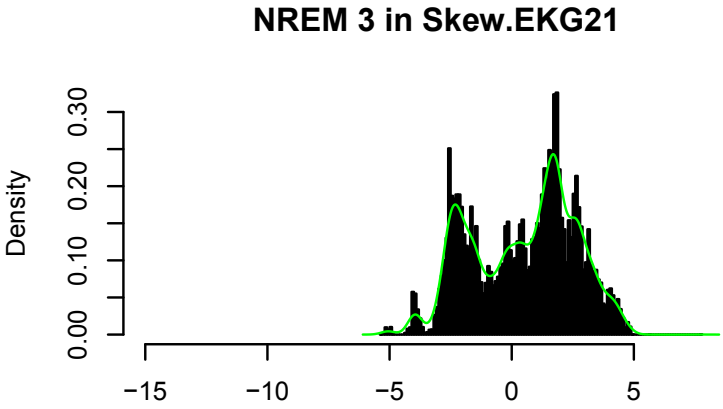

REM in Skew.REOGM2
Density
0.0
0.4
0.8
1.2
−20
−10
0
10
20

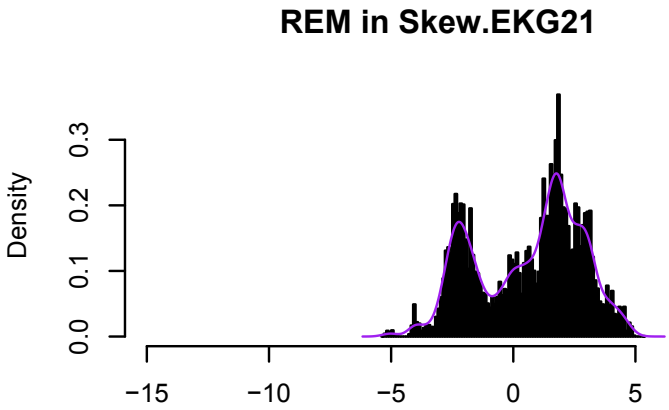

Wake in Skew.REOGM2
Density
0.0
0.2
0.4
0.6
−20
−10
0
10
20

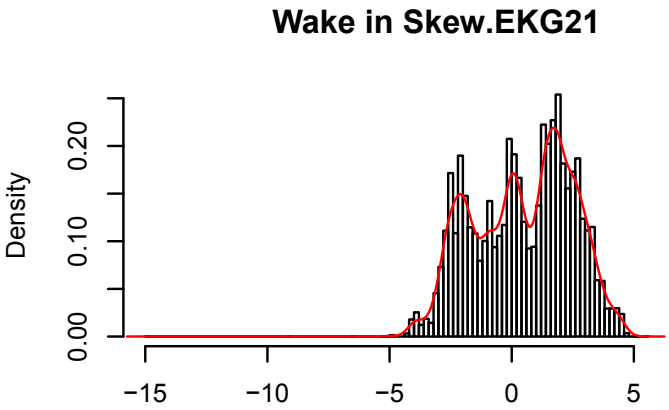

Skew.EKG21
Density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
NonEEG_trans[, kk]

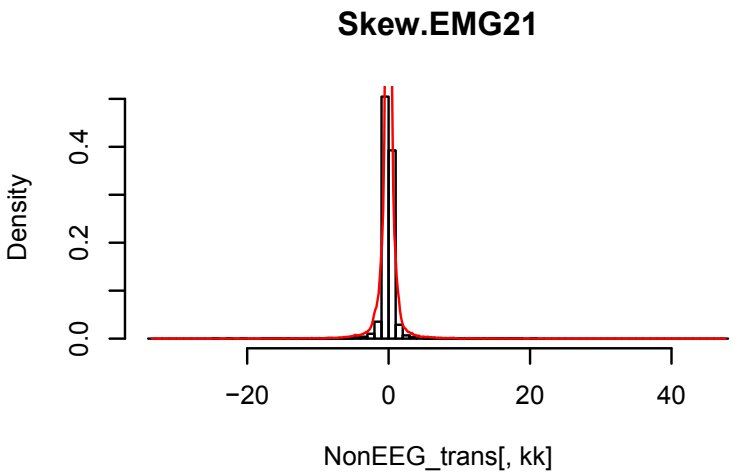

NREM 1 in Skew.EKG21
Density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5

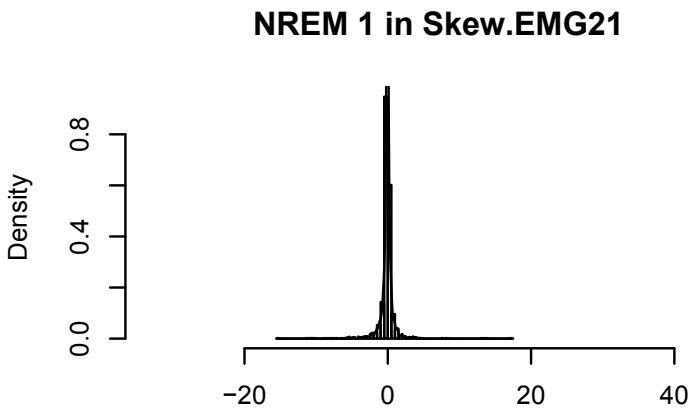

NREM 2 in Skew.EKG21
Density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5

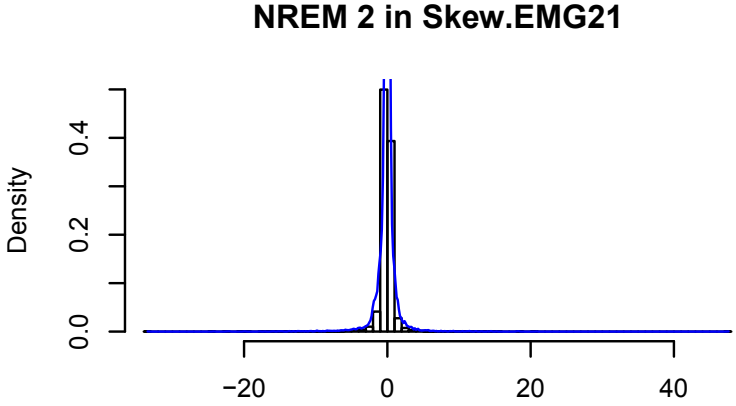

NREM 3 in Skew.EKG21
Density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5

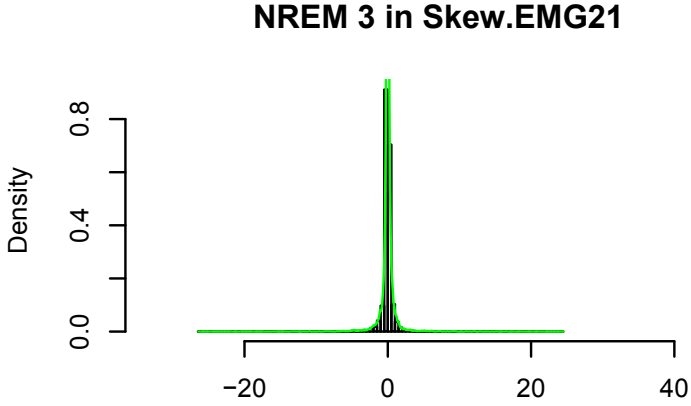

REM in Skew.EKG21
Density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5

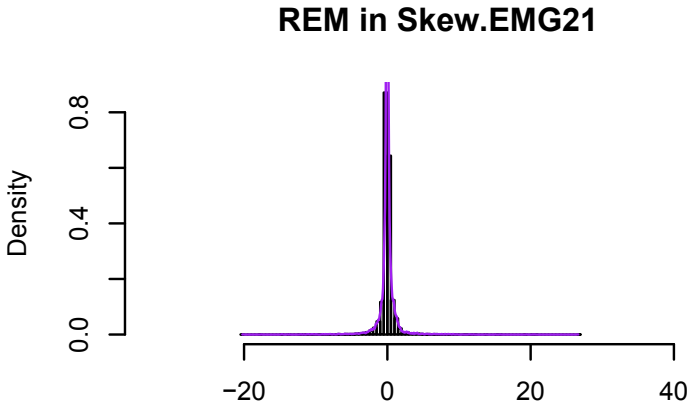

Wake in Skew.EKG21
Density
0.00
0.10
0.20
−15
−10
−5
0
5

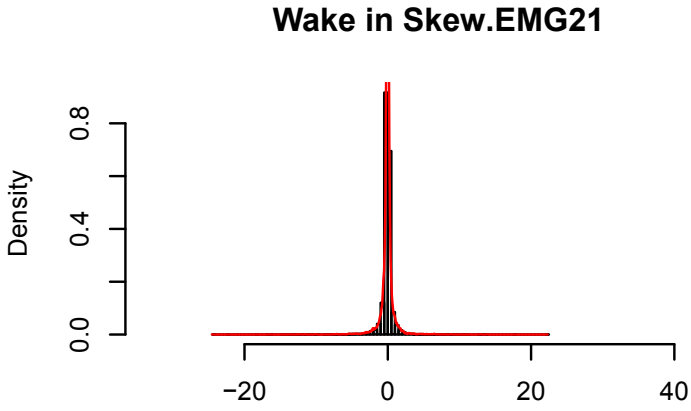

Skew.EMG21
Density
0.0
0.2
0.4
−20
0
20
40
NonEEG_trans[, kk]

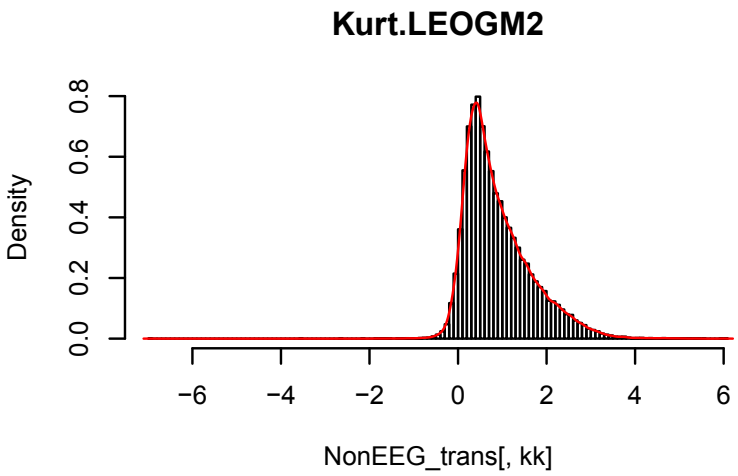

NREM 1 in Skew.EMG21
Density
0.0
0.4
0.8
−20
0
20
40

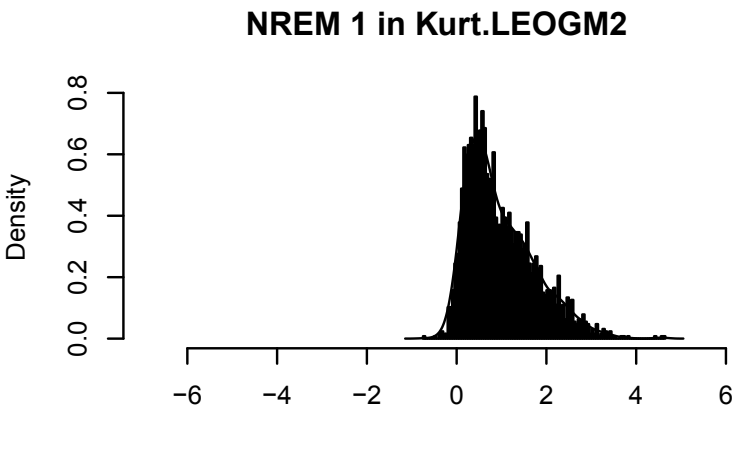

NREM 2 in Skew.EMG21
Density
0.0
0.2
0.4
−20
0
20
40

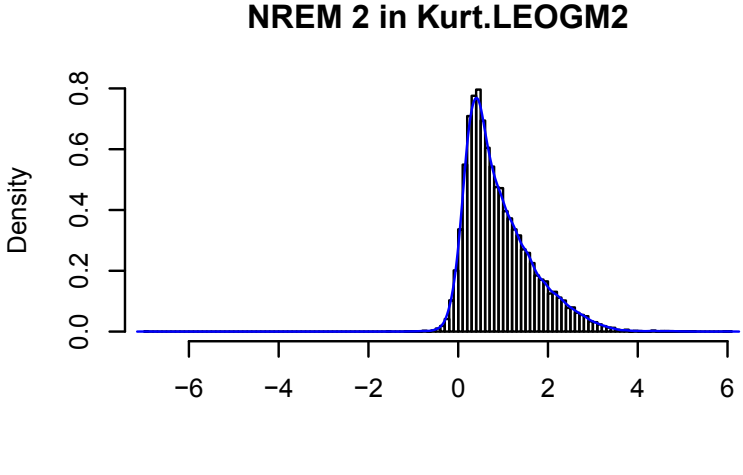

NREM 3 in Skew.EMG21
Density
0.0
0.4
0.8
−20
0
20
40

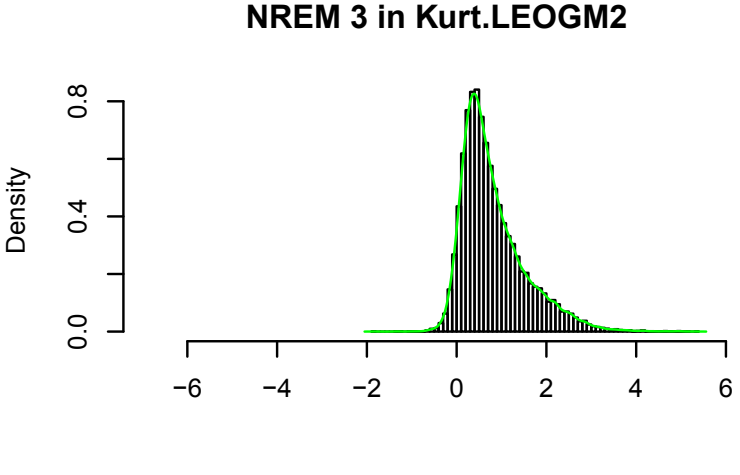

REM in Skew.EMG21
Density
0.0
0.4
0.8
−20
0
20
40

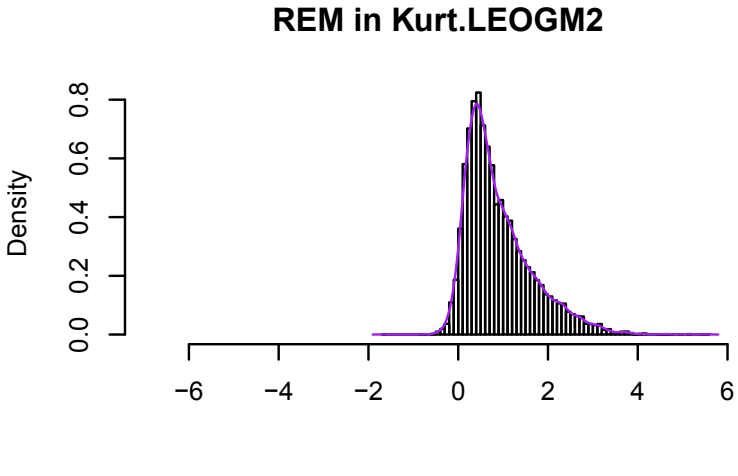

Wake in Skew.EMG21
Density
0.0
0.4
0.8
−20
0
20
40

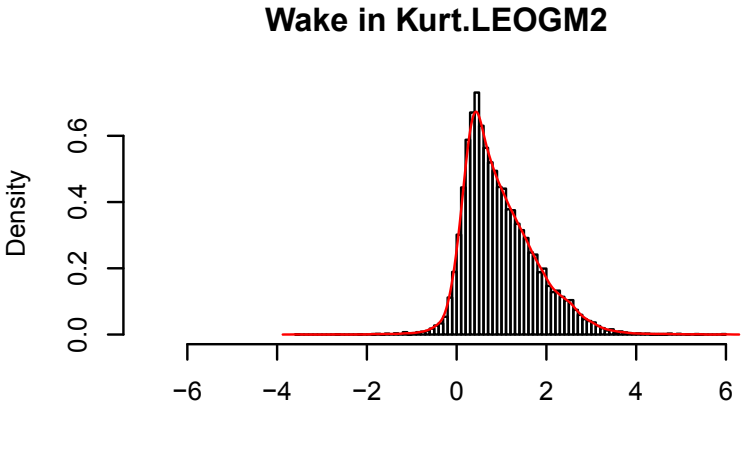

Kurt.LEOGM2
Density
0.0
0.2
0.4
0.6
0.8
−6
−4
−2
0
2
4
6
NonEEG_trans[, kk]

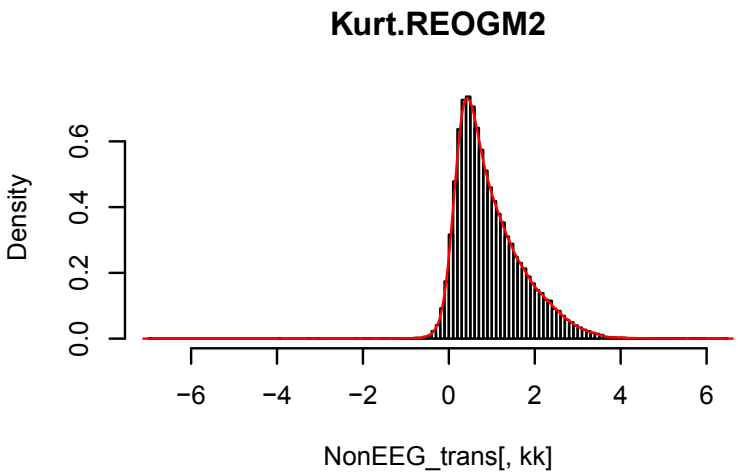

NREM 1 in Kurt.LEOGM2
Density
0.0
0.2
0.4
0.6
0.8
−6
−4
−2
0
2
4
6

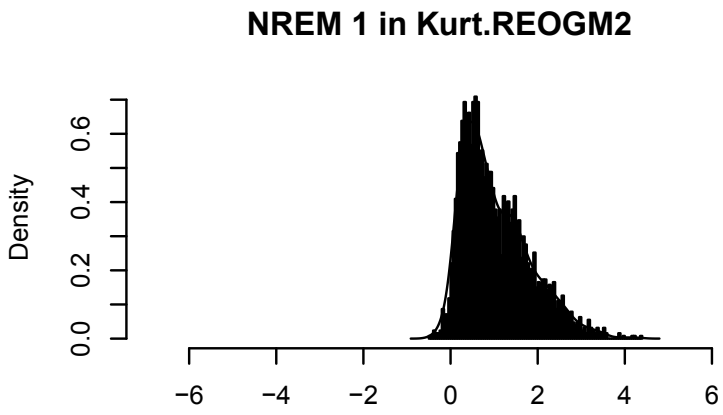

NREM 2 in Kurt.LEOGM2
Density
0.0
0.2
0.4
0.6
0.8
−6
−4
−2
0
2
4
6

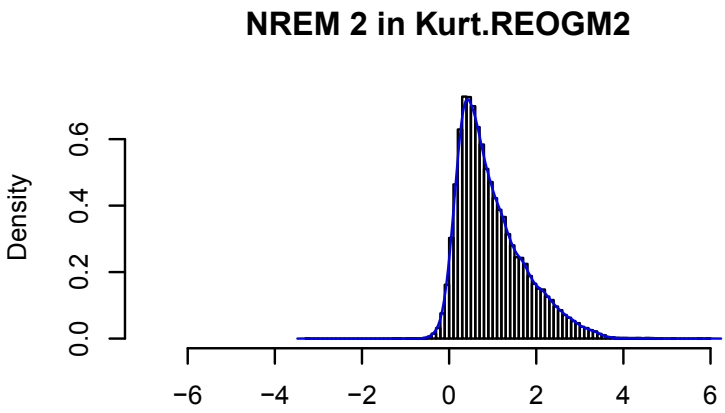

NREM 3 in Kurt.LEOGM2
Density
0.0
0.4
0.8
−6
−4
−2
0
2
4
6

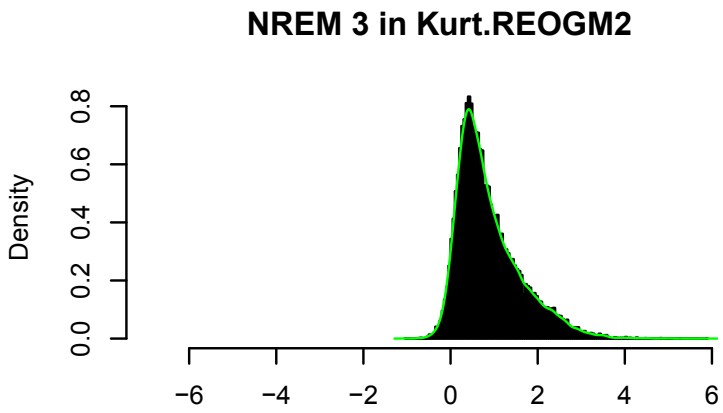

REM in Kurt.LEOGM2
Density
0.0
0.2
0.4
0.6
0.8
−6
−4
−2
0
2
4
6

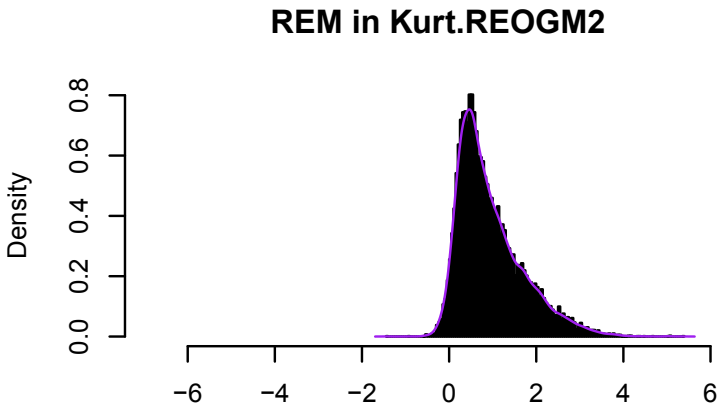

Wake in Kurt.LEOGM2
Density
0.0
0.2
0.4
0.6
−6
−4
−2
0
2
4
6

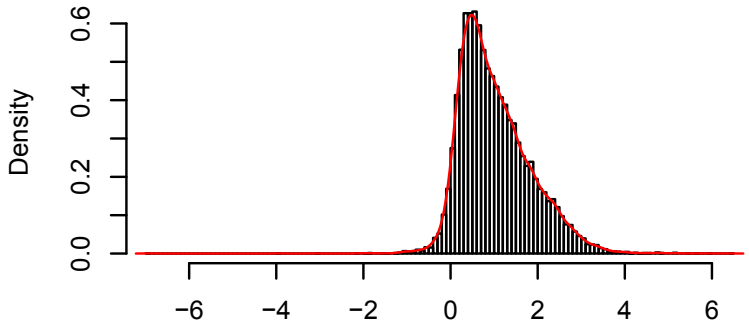

Kurt.REOGM2
Density
0.0
0.2
0.4
0.6
−6
−4
−2
0
2
4
6
NonEEG_trans[, kk]

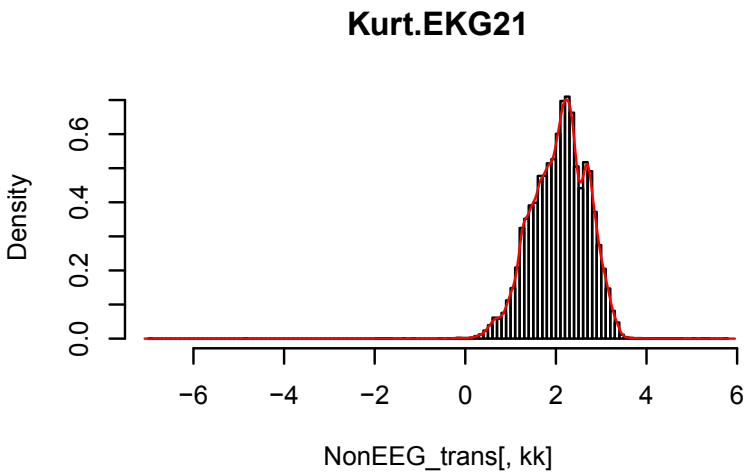

NREM 1 in Kurt.REOGM2
Density
0.0
0.2
0.4
0.6
−6
−4
−2
0
2
4
6

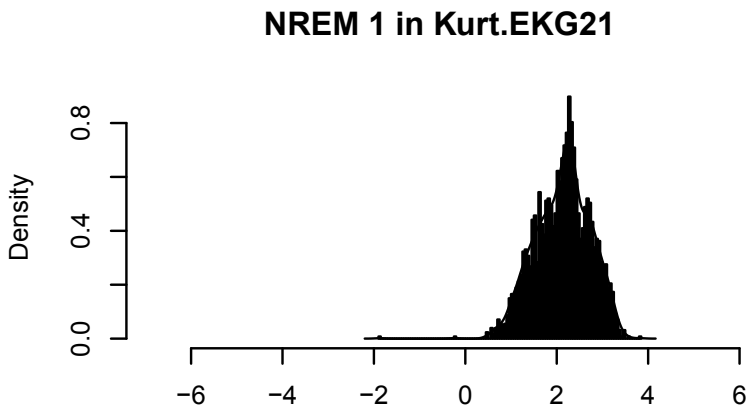

NREM 2 in Kurt.REOGM2
Density
0.0
0.2
0.4
0.6
−6
−4
−2
0
2
4
6

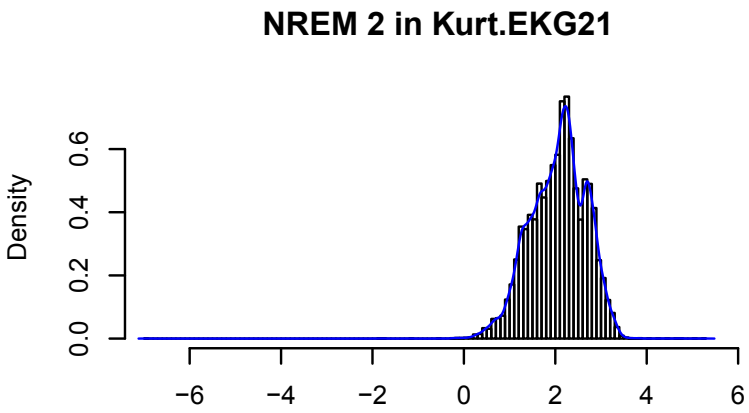

NREM 3 in Kurt.REOGM2
Density
0.0
0.2
0.4
0.6
0.8
−6
−4
−2
0
2
4
6

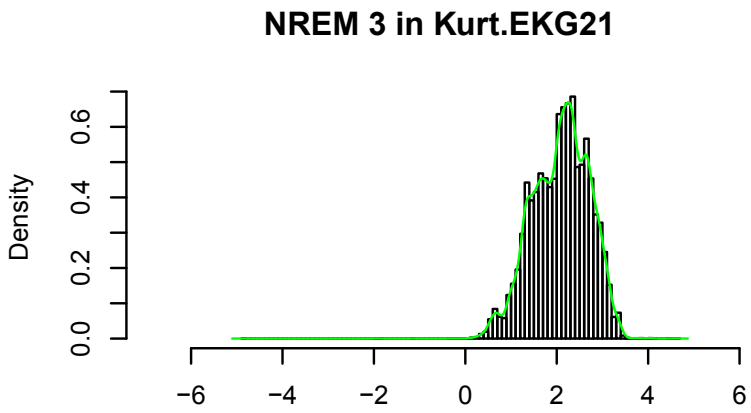

REM in Kurt.REOGM2
Density
0.0
0.2
0.4
0.6
0.8
−6
−4
−2
0
2
4
6

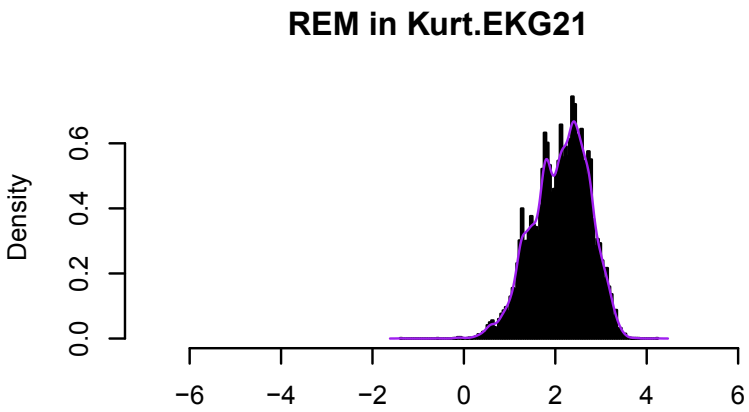

Wake in Kurt.REOGM2
Density
0.0
0.2
0.4
0.6
−6
−4
−2
0
2
4
6

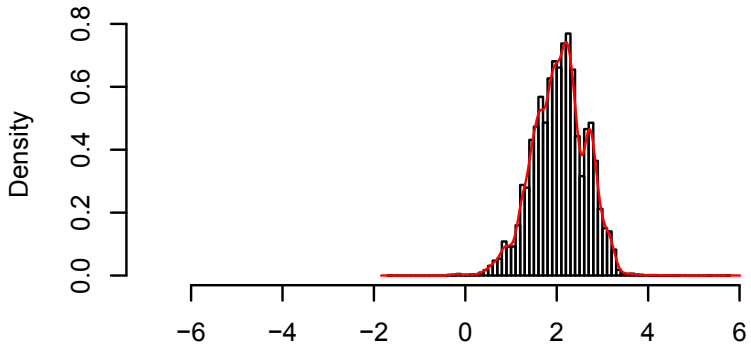

Kurt.EKG21
Density
0.0
0.2
0.4
0.6
−6
−4
−2
0
2
4
6
NonEEG_trans[, kk]

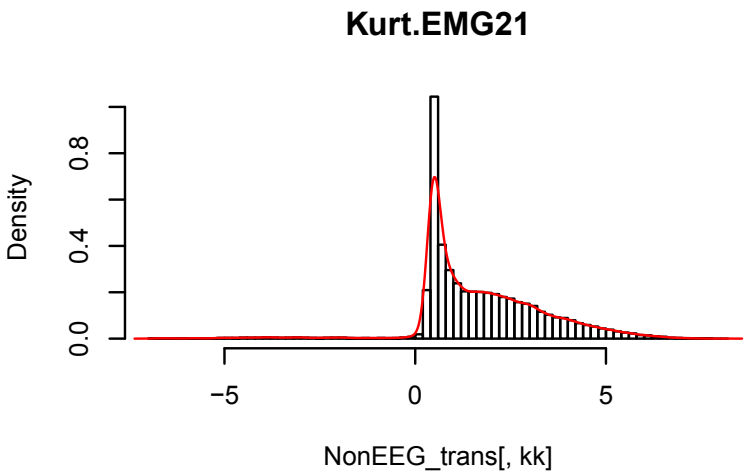

NREM 1 in Kurt.EKG21
Density
0.0
0.4
0.8
−6
−4
−2
0
2
4
6

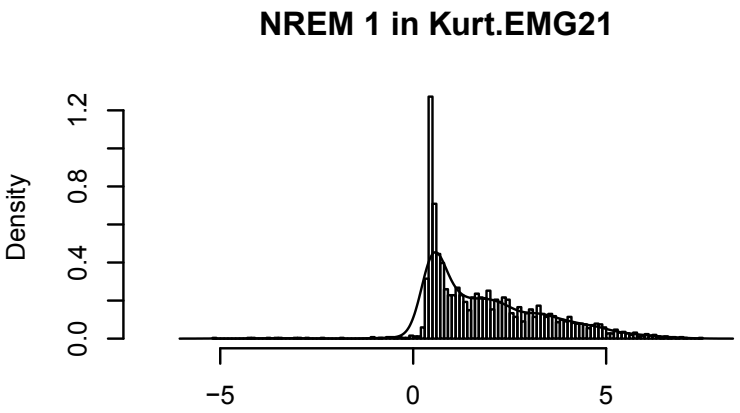

NREM 2 in Kurt.EKG21
Density
0.0
0.2
0.4
0.6
−6
−4
−2
0
2
4
6

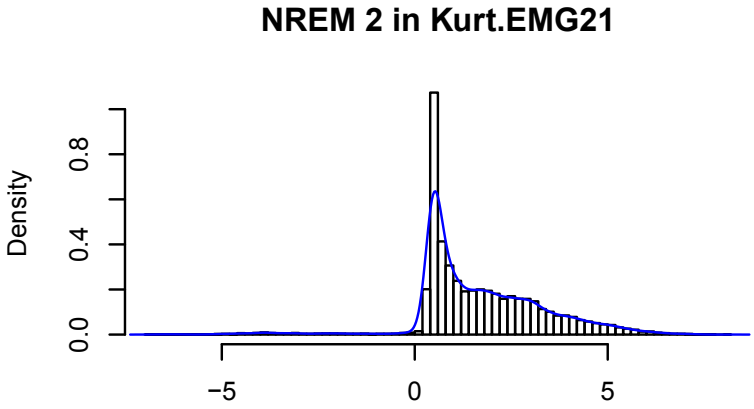

NREM 3 in Kurt.EKG21
Density
0.0
0.2
0.4
0.6
−6
−4
−2
0
2
4
6

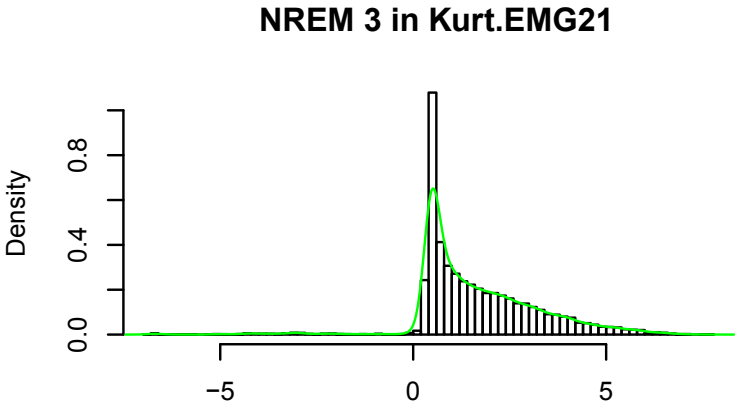

REM in Kurt.EKG21
Density
0.0
0.2
0.4
0.6
−6
−4
−2
0
2
4
6

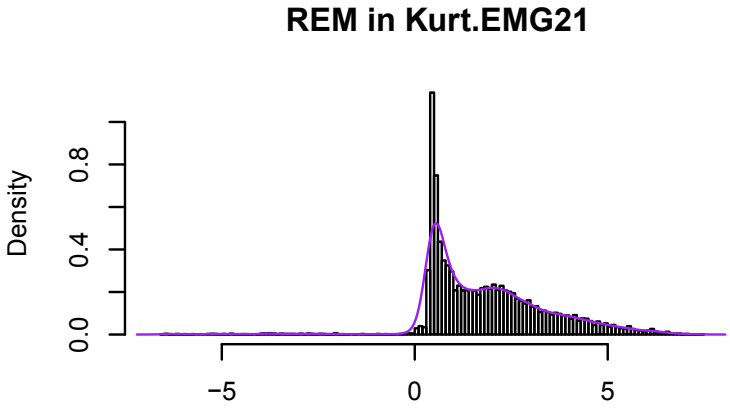

Wake in Kurt.EKG21
Density
0.0
0.2
0.4
0.6
0.8
−6
−4
−2
0
2
4
6

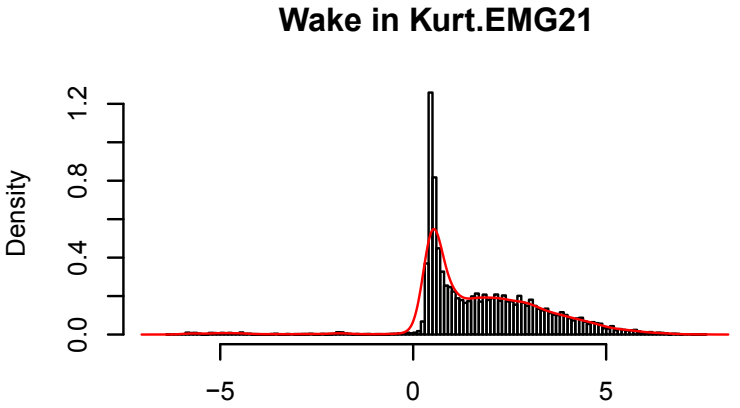

Kurt.EMG21
Density
0.0
0.4
0.8
−5
0
5
NonEEG_trans[, kk]


NREM 1 in Kurt.EMG21
Density
0.0
0.4
0.8
1.2
−5
0
5


NREM 2 in Kurt.EMG21
Density
0.0
0.4
0.8
−5
0
5

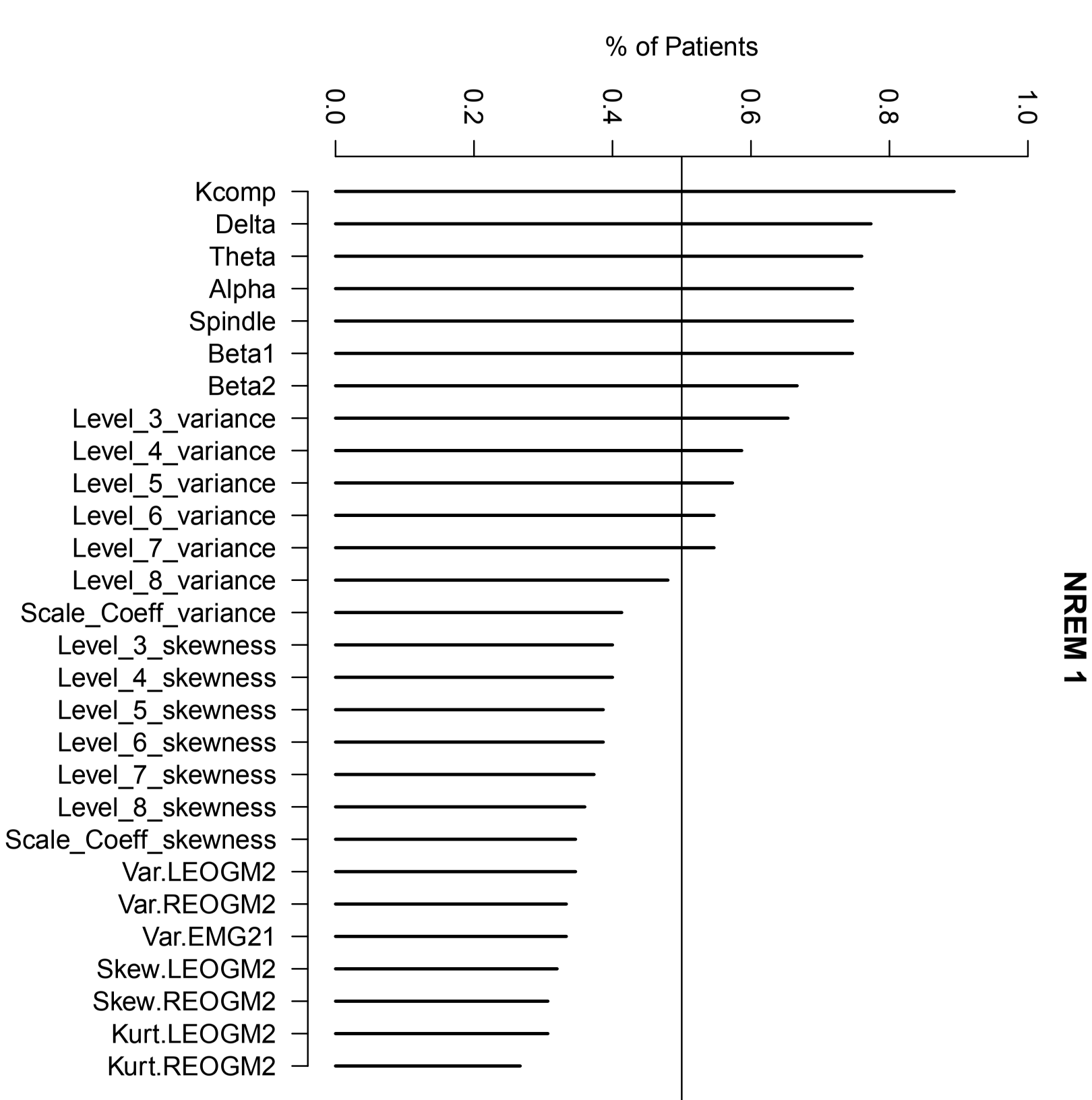

NREM 3 in Kurt.EMG21
Density
0.0
0.4
0.8
−5
0
5

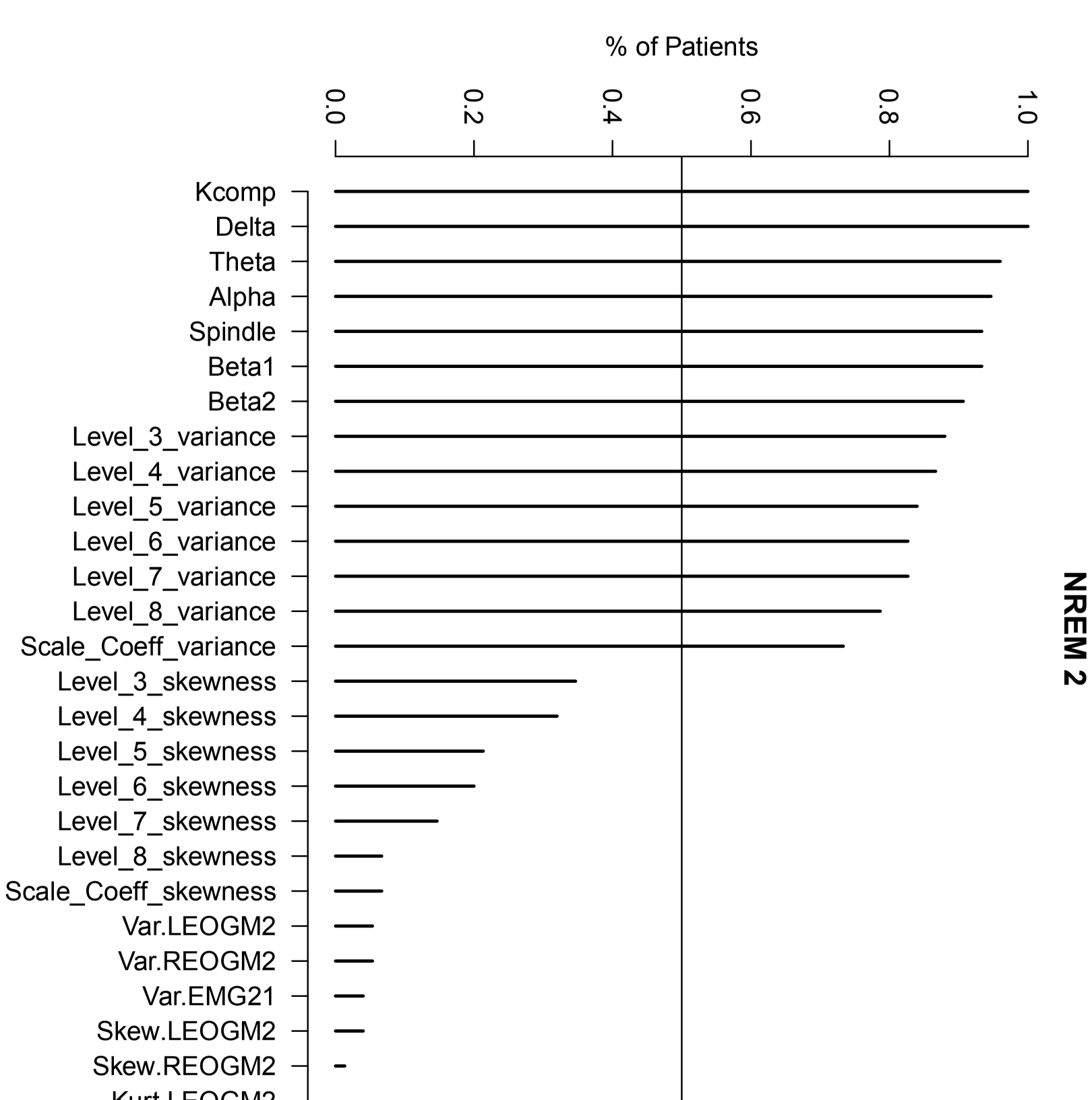

REM in Kurt.EMG21
Density
0.0
0.4
0.8
−5
0
5

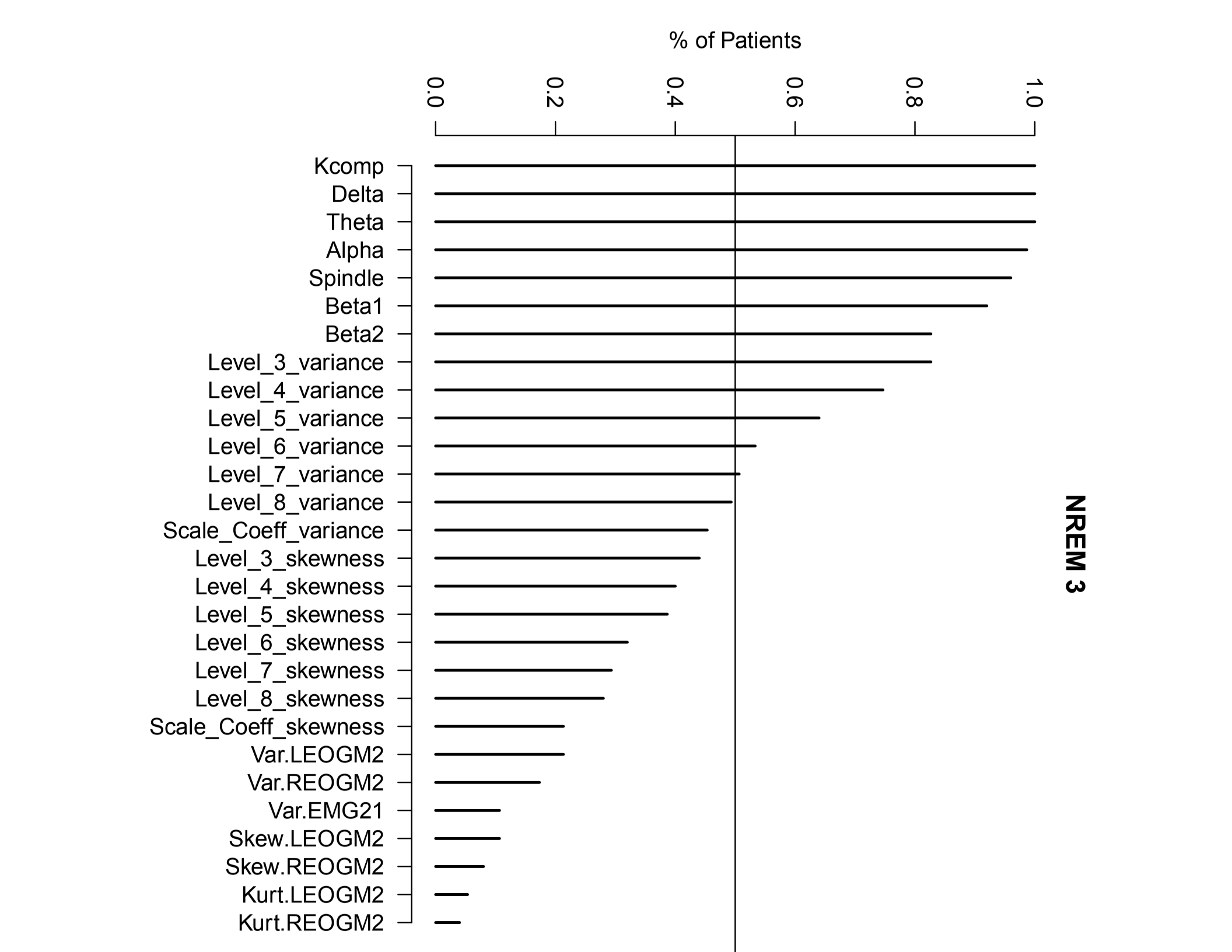

Wake in Kurt.EMG21
Density
0.0
0.4
0.8
1.2
−5
0
5

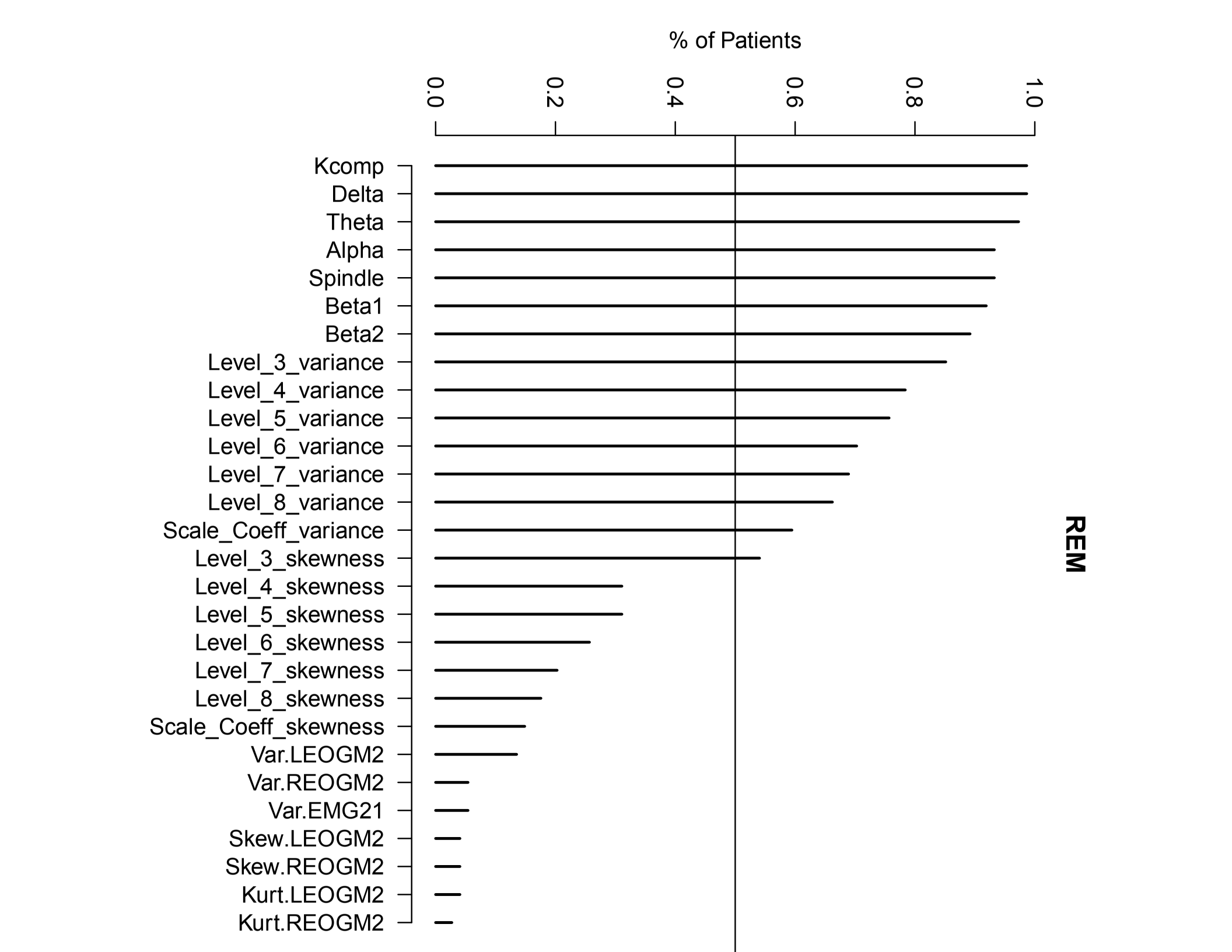
Appendix A.4

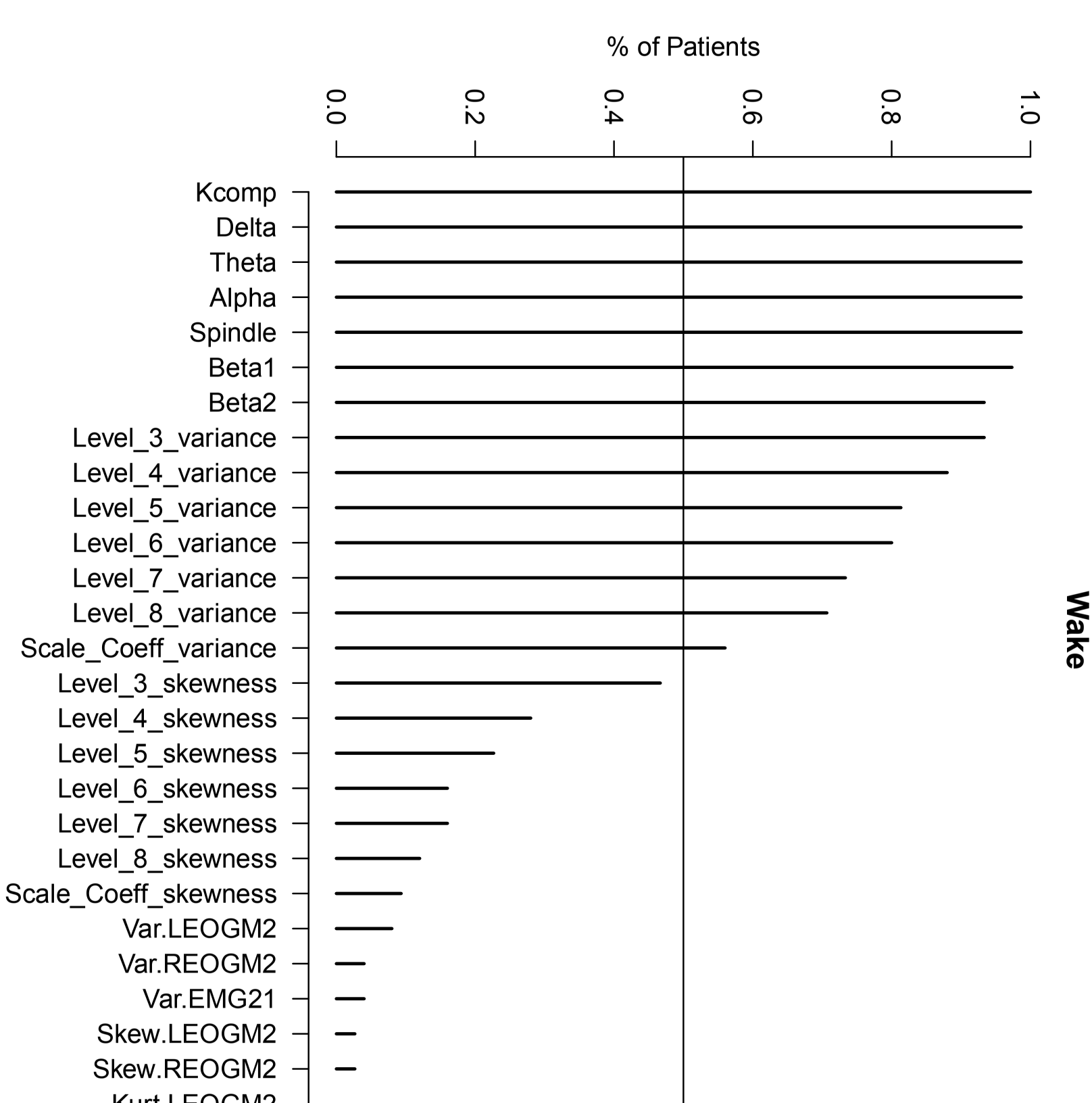
Number of Features Important to Majority of Patients in Each State.


% of Patients
0.0
0.2
0.4
0.6
0.8
1.0
Kcomp
Delta
Theta
Alpha
Spindle
Beta1
Beta2
Level_3_variance
Level_4_variance
Level_5_variance
Level_6_variance
Level_7_variance
Level_8_variance
Scale_Coeff_variance
Level_3_skewness
Level_4_skewness
Level_5_skewness
Level_6_skewness
Level_7_skewness
Level_8_skewness
Scale_Coeff_skewness
Var.LEOGM2
Var.REOGM2
Var.EMG21
Skew.LEOGM2
Skew.REOGM2
Kurt.LEOGM2
Kurt.REOGM2
NREM 1


% of Patients
0.0
0.2
0.4
0.6
0.8
1.0
Kcomp
Delta
Theta
Alpha
Spindle
Beta1
Beta2
Level_3_variance
Level_4_variance
Level_5_variance
Level_6_variance
Level_7_variance
Level_8_variance
Scale_Coeff_variance
Level_3_skewness
Level_4_skewness
Level_5_skewness
Level_6_skewness
Level_7_skewness
Level_8_skewness
Scale_Coeff_skewness
Var.LEOGM2
Var.REOGM2
Var.EMG21
Skew.LEOGM2
Skew.REOGM2
NREM 2


% of Patients
0.0
0.2
0.4
0.6
0.8
1.0
Kcomp
Delta
Theta
Alpha
Spindle
Beta1
Beta2
Level_3_variance
Level_4_variance
Level_5_variance
Level_6_variance
Level_7_variance
Level_8_variance
Scale_Coeff_variance
Level_3_skewness
Level_4_skewness
Level_5_skewness
Level_6_skewness
Level_7_skewness
Level_8_skewness
Scale_Coeff_skewness
Var.LEOGM2
Var.REOGM2
Var.EMG21
Skew.LEOGM2
Skew.REOGM2
Kurt.LEOGM2
Kurt.REOGM2
NREM 3


% of Patients
0.0
0.2
0.4
0.6
0.8
1.0
Kcomp
Delta
Theta
Alpha
Spindle
Beta1
Beta2
Level_3_variance
Level_4_variance
Level_5_variance
Level_6_variance
Level_7_variance
Level_8_variance
Scale_Coeff_variance
Level_3_skewness
Level_4_skewness
Level_5_skewness
Level_6_skewness
Level_7_skewness
Level_8_skewness
Scale_Coeff_skewness
Var.LEOGM2
Var.REOGM2
Var.EMG21
Skew.LEOGM2
Skew.REOGM2
Kurt.LEOGM2
Kurt.REOGM2
REM


% of Patients
0.0
0.2
0.4
0.6
0.8
1.0
Kcomp
Delta
Theta
Alpha
Spindle
Beta1
Beta2
Level_3_variance
Level_4_variance
Level_5_variance
Level_6_variance
Level_7_variance
Level_8_variance
Scale_Coeff_variance
Level_3_skewness
Level_4_skewness
Level_5_skewness
Level_6_skewness
Level_7_skewness
Level_8_skewness
Scale_Coeff_skewness
Var.LEOGM2
Var.REOGM2
Var.EMG21
Skew.LEOGM2
Skew.REOGM2
Wake