Strategies for Better Sleep Spindle Detection

Wendy Nie, Chengcheng Fan

1. Introduction

1.1 Sleep Spindles

Sleep spindles, hallmarks of Stage 2 sleep, are bursts of brain activity that may be detected through electroencephalography (EEG) measurements - measurements of voltage across a scalp. Traditionally, they are scored visually; however, this approach is time-consuming. Applying machine learning to sleep spindle detection would reduce identification efforts.

Sleep spindles are comprised of a group of rhythmic waves which progressively increase and then gradually decrease in amplitude [1] . According to AASM standards, spindles are scored according to their frequency (11-15 Hz), duration (≥ 0.5 seconds), and envelope (maximal amplitude should occur at the temporal midpoint of a spindle) [2].

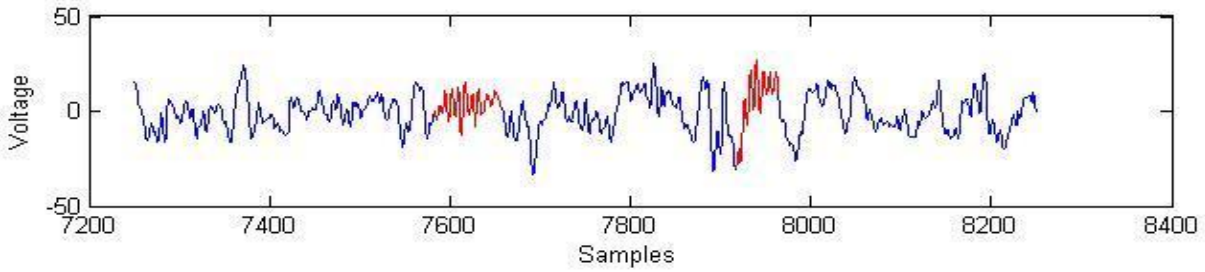


Figure 1: Examples of Spindles (shown in red) from Raw Data Set

1.2 Data

EEG voltage measurements, sampled at 100 Hz, were taken from the central scalp region of 110 sleeping subjects. Experts of sleep research were polled to identify sleep spindles. From these polls, a set of 1987 “Gold Standard” (GS) spindle observations was established and each time series sample was marked spindle-positive or spindle-negative.

Researchers then applied a wavelet-based detector to the raw data with the goal of estimating spindle placement and duration. Estimated spindles are true positives if their overlap with a GS is greater than or equal to 20% of their union with the same GS. The detector yielded 1014 true positive (TP), 877 false positive (FP), and 971 false negative (FN) observations.

From each observation, 44 features were extracted.

2. Objectives

The objective of this project is to improve the performance of the sleep spindle detection system. Two strategies were used to achieve this

1) Cascade the results of the wavelet detector with a classifier that discriminates between true positive (TP) and false positive (FP) observations. To improve classification, we must choose a classifier and select the most effective of the 44 extracted features.

2) Discretize the time series data and generate and test a hidden Markov model (HMM). This strategy enables the incorporation of contextual data, which is difficult to extract in the cascaded system paradigm. To improve detection, a good state model must be selected and accurate estimations of state transition probabilities and observation probabilities must be made.

3. Methods

3.1 Wavelet Detector and Classifier Cascade

A data flow chart of the cascaded wavelet detector and classifier system can be found in Figure 2. The classifier classifies positive observations of the wavelet detector (TP & FP), yielding a different set of overall false positive (FP’), true positive (TP’), and false negative (FN’) observations. Referring to the notation in Figure 2, the total number of cascaded FP’, TP’, and FN’ observations can be computed from FP, TP, and FN observations and classifier sensitivity and specificity estimates using equations 1 through 3. The addition of the classifier increases overall specificity.

TP’ = (TP)(sensitivity’) (eq. 1)

FP’ = (FP)(1 - specificity’) (eq. 2)

FN’ = FN + TP (1 - sensitivity’) (eq. 3)

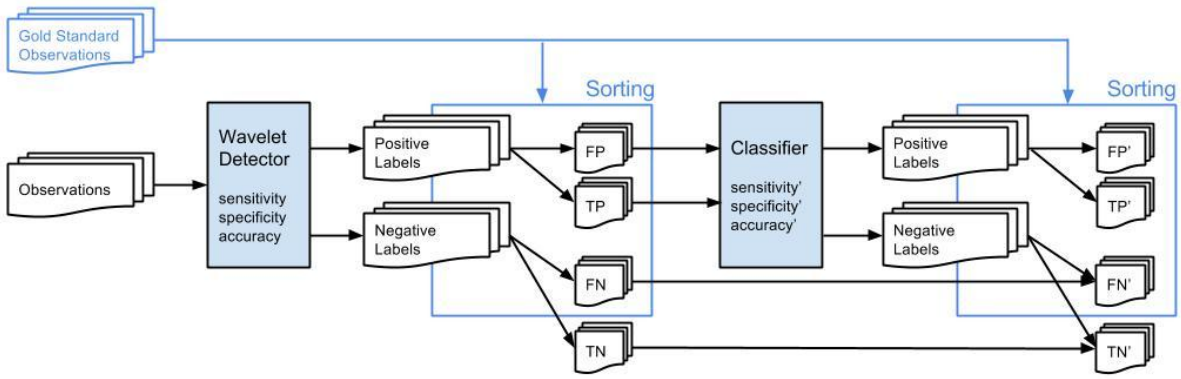


Figure 2: Structure of the Cascaded System

3.1.1 Feature Selection

To reduce redundancy and processing time, subsets of features were selected and tested. Blind selection schemes include forward selection and minimization of correlation between features within a set. Schemes requiring more human involvement include PCA for dimension reduction and subgroup feature selection.

Forward search terminates when the algorithm accuracy exceeds 90%. After repeated trials, features are ranked by how often they were selected.

We assume that the stronger the correlation between two features, the less information they provide as a group. In correlation minimization, the squared-cross-correlation feature pairs are arranged in a matrix. The value of feature set, {i,j,k}, is the sum of all the squared-cross-correlation values in rows and columns, i, j, and k. The set with the smallest value is assumed to be optimal.

In subgroup feature selection, features were grouped in terms of similarity and one feature from each group is added to the final set; for example, RawFrequency and Huupp frequency are likely to be highly-correlated so either one or the other is added to the final set. To assess features within a subgroup, the performance of a single feature from the subgroup with all features from every other subgroup is evaluated.

PCA was predominantly used to visualize features within the subspace; however, assuming the principal component direction has a high variance because FP and TP are nearly separable within that direction, then the principal component vector also hints at the importance of each feature for classification. The features corresponding to the elements with the largest absolute values in the principal component vector might be a part of the optimal set of features.

3.1.2 Classifier Selection

Two linear classifiers and SVM were considered.

Logistic regression and Gaussian discriminant analysis are probability-based. Gaussian discriminant analysis is asymptotically efficient if the assumption that features follow a Gaussian distribution a feature space is true.

For SVM classification, we followed the recommendations given by Hsu et al. [2] for tuning parameters of SVM classifier. We started with Gaussian kernels and set the regularization parameter to 1. Then we tried different parameters for training, and selected the best parameters via cross validation. The important parameters recommended by Hsu et al. [2] are C for soft margin and scaling factor (sigma) in the radial basis function kernel. We repeated this process for polynomial and multilayer perceptron kernels.

3.2 Time-Series Windows & the Hidden Markov Model

As an alternative, the time-series data was divided into windows that are labeled either spindle-positive or spindle-negative based on GS scoring.

First, a moving average of the time domain data, found through convolution of the data with a 20-sample Gaussian blur filter, was subtracted from the original data. Windows of 64 samples that advanced at a hop rate of 10 samples were used. Features were extracted from each window. Windows were spindle-positive if over 90% of its samples were spindle-positive.

It was assumed that the data would adhere to a hidden Markov model. Within this model, each window has a hidden state, Z_n, and an observation, X_n.

Window state was chosen to be the number of past spindle-positive or spindle-negative windows adjacent to the current state. If a series of spindle-positive windows are noted, the state is positive. If a series of spindle-negative windows are noted, the state is negative. If the window switches from spindle-negative to spindle-positive or vice versa, its state becomes 0. States were capped within the range of -10 to 20, representing 1 and 2 seconds of data, respectively. These values were chosen because of typical spindle duration and time-between-spindle values.

3.2.1 Forward-Backward Algorithm

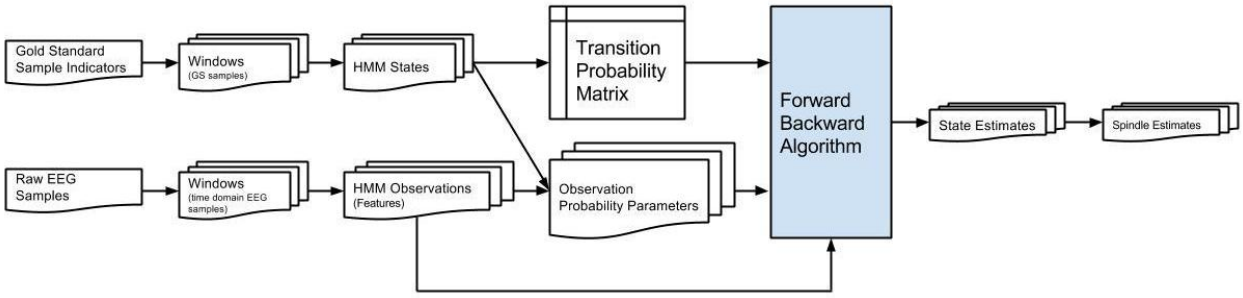


Figure 3: Sliding Windows & Forward Backward Algorithm

Taking advantage of conditional independence in the HMM model, a recursive means of computing the probabilities of Z_n given all X values exists [3]. In the equations below, i and j represent state numbers, Z_n represents the state of sample window n, and X_n represents the observation at sample window n. $X_{n:m}$ represents all the observations spanning windows n through m. N is the total number of windows.

The probability that a Z_n is state i is the product of $\alpha_i(n)$ and $\beta_i(n)$.

$$p(Z_n = i|X_{1:N}) = p(Z_n = i|X_{1:n})p(X_{n+1:N}|Z_n = i, X_{1:n}) = p(X_{n+1:N}|Z_n = i)p(Z_n = i|X_{1:n}) = \alpha_i(n)\beta_i(n)$$

(eq. 4)

α and β may be computed recursively.

$$p(Z_n = i, X_{1:n}) = \sum_j p(Z_{n-1} = j, X_{1:n-1})p(Z_n = i|Z_{n-1} = j, X_{1:n-1})p(X_n|Z_{n-1} = j, X_{1:n-1}, Z_n = i)$$

$$= p(X_n|Z_n = i)\sum_j p(Z_{n-1} = j, X_{1:n-1})p(Z_n = i|Z_{n-1} = j)$$

(eq. 5)

$$\alpha_i(n) = p(X_n|Z_n = i)\sum_j \alpha_j(n-1)p(Z_n = i|Z_{n-1} = j)$$

(eq. 6)

$$p(X_{n+1:N}|Z_n = i) = \sum_j p(Z_{n+1} = j|Z_n = i)p(X_{n+2:N}|Z_{n+1} = j, Z_n = i)p(X_{n+1}|X_{n+2:N}, Z_{n+1} = j, Z_n = i)$$
$$= \sum_j p(X_{n+2:N}|Z_{n+1} = j)p(X_{n+1}|Z_{n+1} = j)p(Z_{n+1} = j|Z_n = i) \quad \text{(eq. 7)}$$

$$\beta_i(n) = \sum_j \beta_j(n+1)p(X_{k+1}|Z_{k+1} = j)p(Z_{k+1} = j|Z_k = i) \quad \text{(eq. 8)}$$

α and β are initialized as follows.

$$\alpha_i(1) = p(Z_1 = i, X_1) = p(Z_1 = i)p(X_1|Z_1 = i) \quad \text{(eq. 9)}$$

$$\beta_i(N) = p(X_N|Z_N = i) \quad \text{(eq. 10)}$$

To perform the recursion described, the transition probabilities between states, the probability of a specific observation, and the probability of the initial state must be approximated using training samples. To maintain tractable values, $\alpha_i(n)$ and $\beta_i(n)$ were scaled to sum to 1 for each n.

4. Results

4.1 Cascaded System Results

4.1.1 Initial Classifier Cascade Test

An initial performance check was used to verify the effectiveness of logistic regression, GDA, and SVM, and to establish baseline sensitivity, specificity, and accuracy values when all 44 features are used. The results in Table 1 were amassed through 50 trials of 10-fold cross-validation.

Table 1: Initial Cascade Test Results

	Logistic	GDA	SVM (Gaussian kernel & C=1)	SVM (Polynomial kernel & C =1)	SVM (MLP kernel & C = 1)
sensitivity	80.0 ± 3.5 %	58.1 ± 4.4 %	88.7 ± 6.4 %	71.3 ± 0.96 %	61.0 ± 3.5 %
specificity	80.6 ± 5.5 %	85.2 ± 3.4 %	21.0 ± 4.8 %	66.4 ± 0.81 %	66.6 ± 2.33 %
overall accuracy	80.3 ± 3.4 %	70.8 ± 3.3 %	57.2 ± 1.3 %	69.0 ± 0.41 %	63.6 ± 1.06 %

1014 TP, 877 FP, and 971 FN observations were previously made using the wavelet-based detector. Cascading the wavelet based detector with a logistic regression for separating TP from FP results in 814 TP, 170 FP, and 1174 FN observations.

4.1.2 Feature and Classifier Selection Results

4.1.2.1 Forward Search Results

20 trials of forward search were performed using 90% accuracy as the threshold for terminating the run. Figure 5 shows the frequencies of the 44 features being selected by 20 forward search trials. The features that are most frequently selected are: 15, 32, 1, 17, 31,14 and 16.

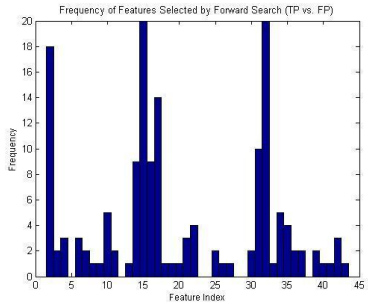


Figure 5: Frequency of Features Selected by Forward Search

	Features (15, 32, 1, 17)	Features (15, 32, 1, 17, 31, 14, 16)	All 44 Features
sensitivity	79.3 ± 4.0 %	80.9 ± 4.7 %	80.0 ± 3.5 %
specificity	78.6 ± 3.3 %	76.5 ± 5.9 %	80.6 ± 5.5 %
overall accuracy	79.0 ± 2.6 %	78.7 ± 4.2 %	80.3 ± 3.4 %

Table 2: Logistic Regression Using Forward Search Results

4.1.2.2 Minimum Correlation

Because the minimum correlation set computes a value for every possible combination, it can only be reasonably used for smaller sets. The set with the lowest value for each number of features are listed in Table 3 along with its corresponding sensitivity, specificity, and overall accuracy. Performance was approximated through 20 trials of 10-fold cross validation with logistic regression (Table 3).

Table 3: Minimum Correlation Sets for FP and logistic regression result

Number of Features	Best Set	Sensitivity	Specificity	Accuracy
3	3 9 24	76.4 ± 7.6 %	45.3 ± 11.3 %	61.6 ± 6.1 %
4	2 5 12 24	81.0 ± 4.7 %	40.2 ± 5.0 %	61.9 ± 4.4 %
5	2 5 12 20 24	74.1 ± 5.5 %	71.1 ± 6.5 %	72.5 ± 3.8 %
6	1 2 5 9 24 33	77.4 ± 7.9 %	69.9 ± 6.8 %	73.6 ± 6.2 %
7	1 2 5 9 15 24 33	79.7 ± 4.3%	77.3 ± 5.2 %	78.6 ± 2.8 %

4.1.2.3 Best within Each Group

The best feature sets found through the minimum correlation technique contain features that would have been intuitively sorted in different feature categories. Following this logic, one last test involved selecting the best features from each category heuristically. The categories and their feature numbers are listed in Table 4.

For each category, feature sets were created using a single feature from the category and every feature from the other categories. Overall accuracy, assessed through 20 logistic regression trials with 10-fold cross validation, was used to rank the category features.

Table 4: Subgroup Feature Selection

Group	Frequency	Amplitude	Time-to-Amplitude	SIR	SPR	ContextBP	ContextWA	Duration	Power	Activity	Sigma Index	Sigma Power
Features, ranked from best to worst	6 4 5 3	10 7	12 11 9 8	20 22 16 21 18 17 19	26 27 24 28 29 30 25	34 36 31 32 35 37 33	41 44 42 39 40 38 43	1	2	14 13	15	23

The best features from each category were combined into the feature set, [1 2 5 10 11 14 15 16 23 26 31 39]. Sensitivity, specificity, and accuracy, estimated through 20 trials of 10-fold cross validation of logistic regression, are 79.4 ± 3.4 %, 79.9 ± 5.4 %, and 79.5 ± 2.4 %, respectively.

4.1.2.4 PCA

We performed PCA on the data and obtained 44 principal components and scores. The scree plot below only shows the first 3 (instead of the total 44) components that explain almost 100% of the total variance; moreover, the first component by itself explains about 80% of the variance, so that it might be a reasonable way to reduce the dimensions. However, when we plot the principal component scores of TP and FP observations, we find that none of the first 3 principal components discriminates the two groups very well (Figure 6).

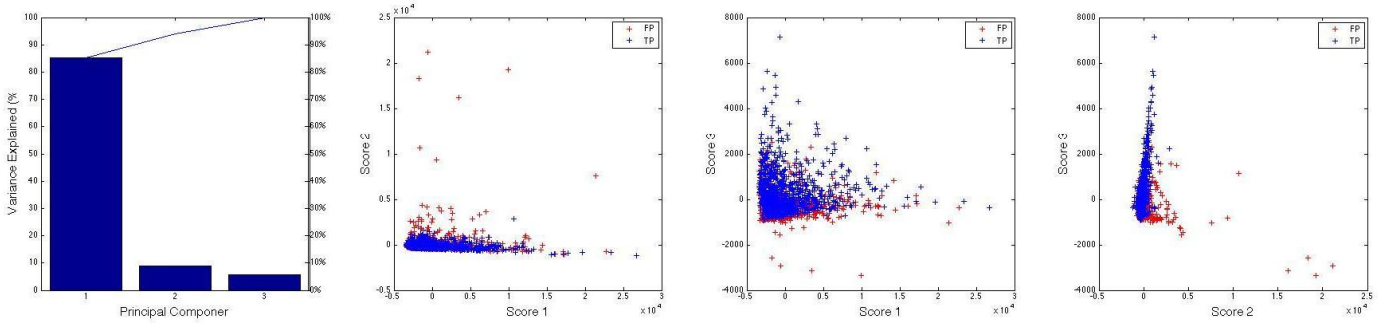


Figure 6: PCA results

4.1.4 SVM Classification

After some initial tests, we decided to use Gaussian kernels for our SVM classifier, because the other two kernels (polynomial and multilayer perceptron) don't converge very well. We tuned parameters by trying a geometric sequence of the regularization parameter C from 1e-2 to 1e5 by a factor of 10; and a geometric sequence of the scaling factor from 1e-5 to 1e5 by a factor 10. The 10-fold cross validation results of this tuning process are shown in Figure 7 below. We found that C = 10, and sigma = 10 give the best accuracy result. We then used SVM and the optimal to select the best 2 features for discriminating FP and TP observations. Feature 15 and feature 32 are the best pairs for discriminating TP and FP using SVM.

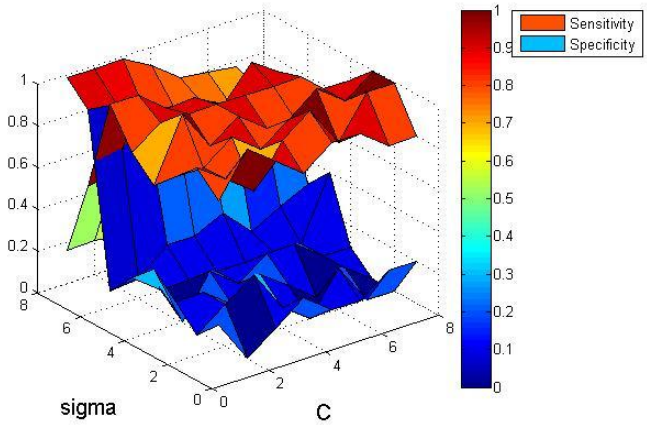


Figure 7: Turning SVM parameters

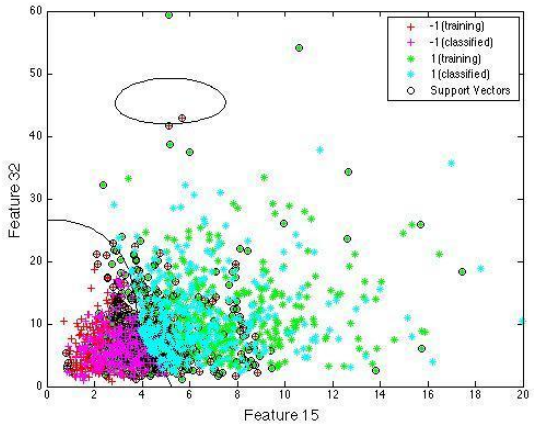


Figure 8: SVM classifier on best 2 feature pairs

	SVM (C=1, sigma = 0.01), 44 features	SVM (C=10, sigma=10) , 44 features	SVM(C=10, sigma=10) , feature 15 & 32
sensitivity	88.7 ± 6.4 %	73.64 ± 0.65%	71.2 ± 0.52 %
specificity	21.0 ± 4.8 %	78.58±0.83%,	82.39 ± 0.34%
Overall accuracy	57.2 ± 1.3 %	75.93±0.28 %	76.38 ± 0.21 %

4.1.5 Forward-Backward Recursion

The subset of features, [1 2 6 10 12 14 15 20 23 26 34 41], was found to be relatively effective at separating TP from FP observations. Some of them, such feature 1 - duration - cannot be computed within the HMM windows paradigm. The ones that could (2, 6, 14, 15, 20, 23, 26, 34, 41) were extracted from each window and used to assess the performance of the forward-backward recursion algorithm.

A single 10-fold cross-validation trial of the forward-backward algorithm resulted in TP, FP, and FN counts of 1415, 6109, and 509, respectively.

5. Discussion & Future Work

5.1 Classifier Cascade

The addition of the classifier to the wavelet detector increases overall specificity and decreases overall sensitivity. If the wavelet detector is configured to have a positive bias, overall performance could be within the desired range.

5.2 Classifier Selection

Logistic regression performed better than Gaussian discriminant analysis, suggesting features don't follow a Gaussian distribution. SVM with Gaussian kernels perform better than linear, polynomial or mlp kernels. After tuning, SVM and logistic regression have similar results.

5.3 Feature Selection

In our forward selection scheme, the most frequently selected features found are likely consistently selected near the beginning, while the ones that appear less often are likely selected later and inconsistently. By adding one feature at a time to the set, we might miss sets that are strong together but weaker alone initially, which is why its results might be different from the minimum correlation set strategy.

The minimum correlation set strategy identifies sets that provide the most information; however, the information is not necessarily useful. For example, if a feature value is always 3, its correlation with other features is always 0, but it's useless in classification. Since brute force computation of correlation values for every combination is not much faster than directly testing the classifier using every set, in this case, a direct classification test might be more useful. Interestingly, the minimum correlation set returns features that would have been manually placed in different subgroups, suggesting the subgroups chosen do, as desired, have low correlation with each other.

Manually dividing the features into subgroups is faster than the minimum correlation set strategy, but assumptions were made about which features are correlated enough to place into a group.

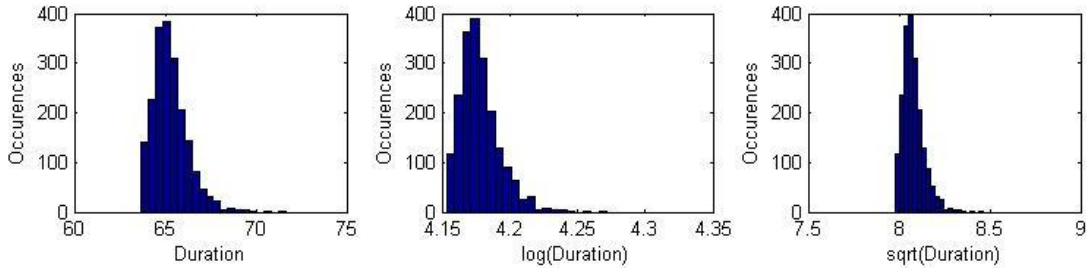
PCA results showed classes aren't separable in the dimensions with the largest variances.

5.4 HMM

A series of HMM tests could be a project in itself. The fact that values didn't pan out could be due to the Markov model assumptions (observations are predictable by a state, observations might not be normally-distributed) or state discretization schemes (window and hop sizes selected).

The hidden state scheme selected might not be a good predictor of observations because the durations of spindles can vary by 4-fold (.5 to 2 seconds) and maximal amplitude always occurs at the center. A better scheme might involve assigning a ratio to each state: the number of spindle-positive windows in a row so far divided by the total number of spindle-positive windows in a row, discretized.

The normal distribution assumption was likely also problematic. Examining histograms of feature values, most features are positive and positively-skewed. If features adhere to the log-normal distribution or the chi-squared distribution with a single degree of freedom, the issue would be simplified because respectively, their logs and square roots would be normally-distributed; however, that was not the case.



Inaccurate conditional observation probability estimates ruin hidden state probability estimates. One way to fix this is to find an established distribution to which the observations can fit. Another way would be to discretize the observations and estimate the conditional probability heuristically, which would require either a lot of training data, or very coarse discretization.

A window size of .64 seconds was selected because spindles last at least .5 seconds, and it might be difficult to discriminate between spindle-positive and spindle-negative states using features extracted from a window that is much larger than .5 seconds; however if window size is too small, the data will be noisy. A hop size that is too large might miss spindles altogether, but if hop size is too small, the number of states becomes too large.

The HMM model is flexible but configurations must be further explored.

6. Conclusions

The best features within each subgroup are [6 10 7 12 20 26 34 41 1 2 14 15 23]. With logistic regression, the set yields sensitivity, specificity, and accuracy values of 79.4 ± 3.4 %, 79.9 ± 5.4 %, and 79.5 ± 2.4 %, respectively. With SVM, this yields sensitivity, specificity, and accuracy values of 74.32±0.49% , 80.0±0.42%, and 76.9 ±0.24 %. Based on this, we recommend both classifiers.

For the wavelet detector and classifier cascade, this results in 805 TP, 176 FP, and 1180 FN observations. The total number of false observations of the cascaded system is 1356. The total number of false observations for the wavelet detector is 1848. Thus, the classifier increases accuracy, but it should be noted that it decreases sensitivity and increases specificity.

The HMM model made 1415 TP, 6109 FP, and 509 FN observations – its sensitivity is high, but its specificity is too low.

Although there is an opportunity to capture more contextual information using the HMM model, important features, such as an observation duration estimate, are also lost. The HMM model parameters must be tested more thoroughly.

Because true negatives are so numerous, and because true positive data is required for research, in spindle detection, it would make sense to maximize specificity given a sensitivity approaching 100% (i.e. yielding false positives is okay as long as there are very few false negatives).

References

[1] Silber, Michael H., et al. "The visual scoring of sleep in adults." *J Clin Sleep Med* 3.2 (2007): 121-131.

[2] http://www.aasmnet.org/scoringmanual/v2.0.2/html/index.html?GScoringStageN2.html

[3] Hsu, Chih-Wei, Chih-Chung Chang, and Chih-Jen Lin. *A Practical Guide to Support Vector Classification*.

[4] T. Mailund, C. Storm, "Hidden Markov Models" slide show for PATTERN RECOGNITION IN BIOINFORMATICS, Department of Computer Engineering, Aarhuss, 2012