Automatic Feature Selection for Sleep/Wake Classification with Small Data Sets

J. Foussier[1], P. Fonseca[2], X. Long[2] and S. Leonhardt[1]
[1]*Philips Chair for Medical Information Technology, RWTH Aachen University, Pauwelsstrasse 20, 52074 Aachen, Germany*
[2]*Philips Research Eindhoven, High Tech Campus 34, 5656AE Eindhoven, The Netherlands*

Keywords: Sleep Monitoring, Sleep Staging, Feature Selection, Linear Discriminant Classification, Unobtrusive Monitoring, Cohen's Kappa, Spearman's Ranked-order Correlation.

Abstract: This paper describes an automatic feature selection algorithm integrated into a classification framework developed to discriminate between sleep and wake states during the night. The feature selection algorithm proposed in this paper uses the Mahalanobis distance and the Spearman's ranked-order correlation as selection criteria to restrict search in a large feature space. The algorithm was tested using a leave-one-subject-out cross-validation procedure on 15 single-night PSG recordings of healthy sleepers and then compared to the results of a standard Sequential Forward Search (*SFS*) algorithm. It achieved comparable performance in terms of Cohen's kappa ($\kappa = 0.62$) and the Area under the Precision-Recall curve ($AUC_{PR} = 0.59$), but gave a significant computational time improvement by a factor of nearly 10. The feature selection procedure, applied on each iteration of the cross-validation, was found to be stable, consistently selecting a similar list of features. It selected an average of 10.33 features per iteration, nearly half of the 21 features selected by *SFS*. In addition, learning curves show that the training and testing performances converge faster than for *SFS* and that the final training-testing performance difference is smaller, suggesting that the new algorithm is more adequate for data sets with a small number of subjects.

1 INTRODUCTION

Sleep is an essential process in most animals, including human beings, and although it has been studied for centuries, relatively little is known about it. It is clear, however, that sleep is essential to survive, as sleep deprivation studies on rats have shown (Rechtschaffen and Bergmann, 1995). Computer-aided sleep assessment was introduced to reduce the manpower and costs needed to collect and interpret data during these studies. However, most of these systems still require the subjects to spend one or more nights in a sleep laboratory, which remains a rather expensive and inconvenient procedure. Ambulatory sleep monitoring aims precisely at eliminating this requirement and can effectively be used for diagnosing several sleep disorders. For this, new sensors and algorithms are needed. Significant work has been done to exploit the fact that certain autonomic changes associated with different sleep stages also manifest themselves differently in parameters such as cardiorespiratory activity and body movements. By evaluating how these parameters change, it should be possible, at least to a certain extent, to distinguish some of these stages without resorting to EEG. Several research groups have worked on the extraction of cardiorespiratory and body movement features (e.g., (Devot et al., 2007), (Devot et al., 2010), (Redmond et al., 2007) or (Zoubek et al., 2007)). However, one of the main issues is that many publications address the sleep stage classification problem from a rather limited set of physiological features. Many authors report how successful a certain feature is for the classification task, instead of focusing on methods that aim at selecting the best set of features as we will show later. There is, in fact, a plethora of features described in literature which can be readily used for the task of sleep staging or the extraction of relevant sleep parameters.

Most available PSG data were generated for patients with sleep disorders. As a result, prior work related to sleep staging of healthy subjects with cardiorespiratory signals or actigraphy often relies on very small data sets (often less than a dozen subjects) and were collected by individual research groups for the validation of a new sensor and/or feature.

Foussier J., Fonseca P., Long X. and Leonhardt S..
Automatic Feature Selection for Sleep/Wake Classification with Small Data Sets.
DOI: 10.5220/0004245401780184
In *Proceedings of the International Conference on Bioinformatics Models, Methods and Algorithms* (BIOINFORMATICS-2013), pages 178-184
ISBN: 978-989-8565-35-8

Many authors opt to perform a single feature selection step on the entire data set when applying traditional machine learning approaches, clearly biasing the classification results towards positive performance. Also, each single epoch is subject- and time-dependent. Therefore the data of several subjects cannot be (randomly) mixed and tested in a traditional leave-one-out-cross-validation (LOOCV) but rather with a leave-one-subject-out-cross-validation (LOSOCV) procedure, which reduces the number of possible folds in the cross-validation.

Finding the ideal set of features for *sleep/wake* classification, especially for small data sets, is a complex and challenging task especially when the number of features is large. An exhaustive search, although leading to the optimal feature set, is impractical in terms of computational time as soon as the dimension of the feature space becomes larger. Sequential search, backward (*SBS*) and forward (*SFS*), tries to address this issue by following a single search path during the process (Whitney, 1971). However, it often delivers sub-optimal solutions especially in problems with small data sets. In the work of (Zoubek et al., 2007) an example of the employment of the *SFS* algorithm can be found.

Building upon previous research published by (Devot et al., 2007; Devot et al., 2010), we will describe a new feature selection method that is particularly adequate for use in each single training step of the LOSOCV and for linear discriminant classifiers. Linear discriminant classifiers, like most other classifiers, are sensitive to the dimension of the feature space. A large number of features can also cause over-fitting and prevent the classifier from generalizing well to new data when assuming a certain degree of independence between the features. On the other hand, if the dimension is too small the classifier will often be too sensitive to noise (Duda et al., 2001). Computational time also plays a role, especially when the number of available features increases. All these constraints have been taken into account during the design process of the feature selection algorithm. In addition, as we will show, this feature selection method is also well suited for data sets with small number of subjects. Finally, by integrating feature selection in the training step of a cross-validation procedure, we will guarantee that the training (including feature selection) and testing steps are performed on mutually exclusive data sets, and at the same time on the largest possible data set. We will then apply and evaluate the proposed feature selection method within a classification framework used for *sleep/wake* detection in healthy sleepers. In order to highlight the properties of the proposed feature selection algorithm, all classification results, including total computational time, stability of the selected features and generalization capabilities, are compared to a standard Sequential Forward Search (*SFS*) algorithm.

2 METHODS AND MATERIALS

2.1 Data Set

The data set consists of 15 single-night PSG recordings of healthy sleepers – ten female (age 31±12.4 yrs, BMI 24.76±3.7 kg/m^2) and five male subjects (age 31±5.5 yrs, BMI 24.38±2.72 kg/m^2). Each PSG recording includes at least the EEG channels recommended by the American Academy of Sleep Medicine (AASM), a 2-lead ECG and the thoracic respiratory effort. In addition, actigraphy was acquired with a Philips Actiwatch and synchronized with the PSG. Nine subjects were measured in Boston (USA), at the Sleep Health Center, and six subjects in Eindhoven (The Netherlands), at the sleep laboratory of the High Tech Campus. The study protocol was approved by the Ethics Committees of the respective center and all subjects signed an informed consent form. Sleep stages were scored by professional sleep technicians according to the guidelines of the AASM as *wake*, *non-REM sleep 1-3* (N1-N3) and *REM sleep* using 30-second epochs. In order to train and test our classifier for the *sleep* and *wake* classes, we merged the *N1-N3* and *REM* classes into a single *sleep* class.

Since the data were recorded in two different sleep laboratories, with two differently configured PSG systems, the data were first resampled to a common sampling rate (512 Hz for ECG, 10 Hz for respiratory effort, and 30-second period for actigraphy).

2.2 Classification Framework

The classification framework, illustrated in Figure 1, is divided in two main parts: training and classification. Before training, the data set is first split into independent training and testing sets. Each contains manually annotated sleep scores, indicating the sleep stage for every epoch. The predictions of the classifier are compared with the ground-truth annotations and the performance of the classifier on the testing set is computed.

As mentioned in the introduction, autonomic changes associated with different sleep stages will manifest themselves differently in certain physiological parameters. In order to exploit these changes we

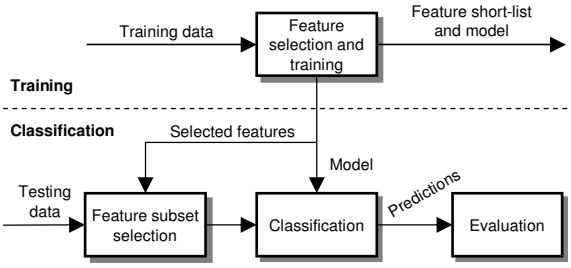

Figure 1: Block diagram illustrating the classification framework.

extracted a total of 60 features from the ECG, the respiratory (thoracic) effort and the actigraphy signals on 30-second epochs. The cardiac features are based on heart-rate-variability (HRV) evaluated in time and frequency domain. Non-linear properties were also examined based on Detrended Fluctuation Analysis (DFA) and Sample Entropy. The respiratory features were defined in the time domain - including statistical measures derived from both the signal waveform and respiratory period and non-linear measures of "similarity" - as well as in the frequency domain. For actigraphy, we used so-called activity counts, directly acquired with the Actiwatch. Since we did not put any additional effort on the task of feature extraction, we will not mention it further in this paper and refer to previous work (Devot et al., 2010; Long et al., 2012).

The training step comprises an iterative feature selection procedure whereby a short-list of features of the original 60 features is chosen. This short-list should comprise the set of features that best characterizes the different sleep stages accordingly to the annotations of the training set. On each iteration of feature selection, the input feature vectors are reduced to a subset of feature vectors. This subset is then used to train a model which is in turn used to classify the same input data. The training classification performance is fed back to the feature selection procedure.

Assuming that all features are normally distributed and the covariance matrices for all classes are identical, i.e., $\mathbf{\Sigma_a} = \mathbf{\Sigma}$, we have a "linear discriminant" function given by

$$g_a(\mathbf{f}) = -\frac{1}{2}(\mathbf{f}-\boldsymbol{\mu_a})'\mathbf{\Sigma}^{-1}(\mathbf{f}-\boldsymbol{\mu_a}) + \ln(P(\omega_a)) \quad (1)$$

where $\boldsymbol{\mu_a}$ and $\mathbf{\Sigma}$ are the mean vector for class ω_a and the pooled covariance matrix (Duda et al., 2001; Redmond et al., 2007). To use this function in the training step of our classification framework, we need to compute the sample mean and the prior probabilities of each class and the inverse pooled covariance matrix $\mathbf{\Sigma}$. We chose the linear discriminant instead of a quadratic discriminant, because quadratic discriminants are known to require larger sample sizes than linear discriminants and they also seem to be more sensitive to possible violations of the basic assumptions of normality (Friedman, 2012). This is particularly important for classification of features derived from physiological data, which very often do not follow a normal distribution. Furthermore, for problems with small sample sizes it is also common to use the pooled covariance estimate as a replacement of the population class covariance matrices (Friedman, 2012).

Regarding the prior probabilities $P(\omega_a)$ of each class, we used the observation that the different classes have different probabilities throughout the night (Redmond et al., 2007). The time-dependent prior probabilities for a given class can be obtained by counting, for each epoch relative to the beginning (i.e., when lights were turned off) of each recording, the number of times that epoch was annotated with that class. The prior probability term in the linear discriminant (1) can be used to bias the classification to a certain class.

The feature selection procedure described in this paper aims at selecting features that simultaneously offer a high discrimination power between classes, yet are uncorrelated with each other. It makes use of the classifier structure and theory. It can be shown that when using the linear discriminant function in (1) in a two-class problem where the classes are equiprobable the error probability of the classifier depends on the following metric

$$\delta^2 = (\boldsymbol{\mu}_a - \boldsymbol{\mu}_b)'\mathbf{\Sigma}^{-1}(\boldsymbol{\mu}_a - \boldsymbol{\mu}_b). \quad (2)$$

This metric, also called the Mahalanobis distance, reflects the "class separability" for a given feature set. As such, it seems appropriate to measure the discriminating power of each individual feature. When evaluating a single feature k, the inter-class distance between the classes ω_a and ω_b can be rewritten as

$$d_k = \frac{|\mu_{k_a} - \mu_{k_b}|}{\sigma_k} \quad (3)$$

where μ_{k_a} and μ_{k_b} are the population means of each class and σ_k is the standard deviation of feature k. The top-discriminating features are those with the highest inter-class distance d_k. As a measure of correlation between features, the algorithm uses the Spearman's ranked-order correlation (Abdullah, 1990). This correlation measure is particularly robust in the presence of outliers, very common when measuring physiological signs. Unlike the Pearson's correlation, it does not require a linear relation between the features to express the correlation between them. As an example of signals with high Spearman's ranked-order and low Pearson's correlation, consider the inter-beat interval

(*IBI*) and the derived instantaneous heart rate (*HR*): $HR = IBI^{-1}$. It is clear that *HR* and *IBI* correlate monotonically, but not linearly.

Maximum discrimination power and minimum correlation are combined in the feature selection algorithm described in the box on the right hand side ("mahal"). The algorithm assigns a score to each feature (steps 1 to 4). The higher the score, the better a feature is for our classification task. An iterative procedure will then search a variety of score thresholds, and determine the classification performance obtained with the corresponding feature short-list (step 5). The highest performance will correspond to the optimal short-list of features for our training set. Note that when cross-validation is used to evaluate the performance of a classifier, this procedure can be used with the training set defined on each iteration. Each short-list can then be used for classification with the testing set of the same iteration.

Both the *mahal* and *SFS* feature selection methods will be evaluated by comparing the performance in terms of κ and AUC_{PR} on the training and the testing set. Performance curves during the feature selection procedure, learning curves and AUC_{PR} values of the classification results using the selected features of each cross-validation step, will all help giving us a good insight of the overall performance of each feature selection method. In addition, the number and diversity of the selected features and the total computation time are analyzed. It can be shown that the fraction of misclassified epochs during LOSOCV corresponds to the maximum likelihood estimate for the (unknown) error rate of a classifier (Duda et al., 2001). Although this procedure has also been used to evaluate the performance of similar classifiers in earlier work, the feature selection was applied on the complete data set, and therefore, also on the testing set. The feature selection described in this paper is applied in each iteration of the LOSOCV, guaranteeing that the testing data used to validate the classifier were not exposed to the tuning and training steps. That means that for each iteration a separate short-list of features is determined.

First, the performance of the classifier was evaluated using the traditional metrics of accuracy, precision, specificity and sensitivity (considering *wake* as the positive class) for each iteration of the LOSOCV and for the pooled results (Fawcett, 2004). However, because the *wake* and *sleep* classes are very imbalanced (the wake epochs represent less than 10% of all epochs) these metrics can fail to give an accurate overview of the performance for both classes (Haibo and Garcia, 2009). For that reason we do not present those metrics in this paper, but compute and

Algorithm 1: (*mahal*).

For the feature values and associated ground-truth $\{\mathbf{f}_i, y_i\}$ of each epoch in a given training set:

Step 1

Compute the inter-class distance for each feature k as

$$d_k = \frac{\left|\overline{f}_{k_a} - \overline{f}_{k_b}\right|}{\sigma_k} \tag{4a}$$

where the sample mean for a given class z and the standard deviation are given by

$$\overline{f}_{k_z} = \frac{\sum\limits_{i \in Z} f_{k_i}}{\#Z}, \text{ for } Z = \{i | y_i = z\} \tag{4b}$$

$$\sigma_k = \sqrt{\frac{\sum\limits_{i=1}^{N} \left(f_{k_i} - \overline{f}_k\right)^2}{N-1}}, \text{ with } \overline{f}_k = \frac{\sum\limits_{i=1}^{N} f_{k_i}}{N} \tag{4c}$$

Collect all unique inter-class distances in an array $\mathbf{d}$.

Step 2

Compute the Spearman's ranked-order correlation $c_{k,l}$ between each feature and the remaining features

$$c_{k,l} = \text{corr}\left(\mathbf{f}'_k, \mathbf{f}'_l\right) \tag{4d}$$

where $\mathbf{f}'_k$ and $\mathbf{f}'_l$ are the feature k and l respectively, for each epoch in the training set.

Step 3

Assign a "score" s_k of zero to each feature

$$s_k := 0, \text{ for } k \in \{1, ..., N_F\} \tag{4e}$$

Step 4

for each M in $\mathbf{d}$ and for $C = 0...1$, step size $\Delta_C = 0.01$

for each feature k

if $d_k > M$ and if the feature is uncorrelated with the others,

$$c_{k,l} < C, \forall l \in \{1, ..., N_F\} \tag{4f}$$

or has a higher distance than the feature it is correlated with,

$$m_k > m_l, \forall l \in \left\{l | l = 1, ..., N_F, c_{k,l} \geq C|\right\} \tag{4g}$$

increase its score

$$s_k := s_k + 1/N_S \tag{4h}$$

where N_S is the number of loop steps,

$$N_S = \frac{\#\mathbf{d}}{\Delta_C} \tag{4i}$$

Step 5

for $S = 0...1$, step size $\Delta_S = 1/N_S$

compile a short-list of features $\mathbf{l}_S$ with score higher than the threshold $\mathbf{l}_S = \{k | s_k > S\}$, compute the performance κ_S of the classifier in the training set using $\mathbf{l}_S$.

Step 6

Return as the final short-list the list of features which gave the highest performance $\mathbf{l} = \mathbf{l}_{S_{MAX}}$, with

$$S_{MAX} = \{S | \kappa_S = \max\left(\{\kappa_1, \kappa_2, ...\}\right)\}. \tag{4j}$$

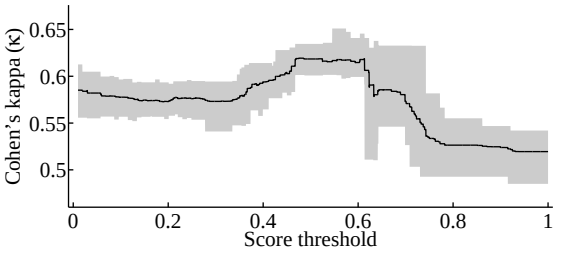

Figure 2: Performance κ on the training set for different score thresholds.

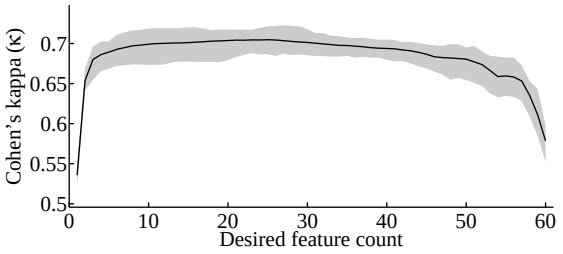

Figure 3: Performance κ on the training set for different desired number of features in the SFS algorithm.

analyze the Cohen's Kappa coefficient of agreement (κ) instead. This metric is directly interpretable as the proportion of joint judgments for which there is agreement, after chance agreement is excluded (Cohen, 1960). Despite their widespread use, these metrics only assess the classifier's performance on a single point in the entire solution space, namely that obtained by directly comparing the output of the discriminants defined for both classes by (1). To compare a classifier with other classifiers this single point might not be sufficient. By doing so, we assume equal misclassification costs and fully known class distributions (Provost et al., 1998). When there is a large imbalance between the classes - as it is the case for *sleep/wake* classification - Precision-Recall (PR) curves should be used instead of Receiver Operating Characteristic (ROC) curves (Davis and Goadrich, 2006). In order to assess the performance of a classifier across the entire solution space, it is customary to compute the area under the curve, in this case, under the PR curve (AUC_{PR}). Unlike the computation of the AUC for the ROC curve, computing the AUC for the PR curve requires a more complex procedure, the composite trapezoidal method proposed by (Davis and Goadrich, 2006).

Finally, we also computed so-called "learning curves" (Duda et al., 2001) to gain insight into the generalization capabilities of the classifier. By varying the number of subjects in the data set, these curves can help predict what the performance of the classifier would be when using more training data. They can be obtained by computing the testing and training error or alternatively, the testing and the training performance (e.g., κ) for data subsets of different size n randomly selected from the whole data set. As n increases, the testing and training performance should approach the same asymptotic value. The convergence speed indicates how well a classifier is suited for small data sets and has to be taken into account when classifying small data sets.

3 RESULTS

First, the performance κ obtained on the training set for each score threshold S (step 5 of the feature selection algorithm) is illustrated in Figure 2. To further show the stability of the feature selection process, we plotted with shaded bands the range between the minimum and maximum κ obtained for each threshold across all iterations of the LOSOCV. The performance peaks around $S = 0.5$, with a relatively narrow shaded band. Intuitively, a narrow performance band means that the performance obtained for a given score threshold is similar across all iterations of the LOSOCV, suggesting that the procedure is stable. Note that with thresholds beyond 0.6, and therefore with smaller short-lists, the performance drops. There seems to be an optimal number of features which on the one hand prevents overfitting while on the other maximizes the generalization capabilities of our classifier. The use of a score threshold in this procedure is advantageous since unlike many other feature selection algorithms we do not need to specify the "desired" number of features, letting that depend on the actual properties of the training set. In a similar manner, by sweeping through a desired number of features, we can observe how the performance of *SFS* evolves (Figure 3). The average performance is maximal when using 26 features. The width of the shaded band is comparable to the one in Figure 2. Note that the performance on the training set is higher when compared with *mahal*, where from the beginning more features correlated with each other, even with high discriminative power, are excluded. This typically leads to a higher performance on the training data set, but, as we will show, not necessarily on the testing data set.

Table 1 lists the performance of the classifier on the testing set for each iteration of the LOSOCV and also the overall performance. The first column indicates the iteration number of the LOSOCV. The three columns with the headers **mahal** and **SFS** indicate the

Table 1: Results on the testing set during cross-validation, considering wake class as positive.

	mahal (total time = 984 s)			SFS (total time = 9205 s)		
it	**κ**	**AUC PR**	**# feat.**	**κ**	**AUC PR**	**# feat.**
1	0.77	0.83	8	0.77	0.85	20
2	0.58	0.73	7	0.70	0.75	15
3	0.66	0.61	7	0.71	0.77	21
4	0.55	0.82	10	0.65	0.89	31
5	0.61	0.80	15	0.67	0.68	31
6	0.94	0.94	7	0.73	0.92	17
7	0.76	0.85	16	0.69	0.78	24
8	0.89	0.93	10	0.81	0.86	22
9	0.76	0.89	8	0.63	0.88	27
10	0.28	0.28	11	0.32	0.32	17
11	0.70	0.79	10	0.71	0.77	21
12	0.60	0.83	9	0.64	0.74	11
13	0.56	0.68	14	0.68	0.76	10
14	0.24	0.35	13	0.49	0.59	30
15	0.53	0.80	10	0.58	0.80	18
pooled	**0.62**	**0.59**	**-**	**0.64**	**0.60**	**-**
mean	**0.63**	**0.74**	**10.33**	**0.65**	**0.76**	**21**
std	**0.19**	**0.19**	**2.94**	**0.12**	**0.15**	**6.69**

κ performance, the AUC_{PR} and the number of selected features for each iteration of the LOSOCV for the *mahal* and the *SFS* algorithm, respectively. The row **pooled** indicates the overall performance obtained after pooling all classification epochs. Note that this is different from the average results, which are indicated in the row **mean** with the standard deviation **std**. In addition, the total computation time for all 15 feature selection steps is included in the first row of the table.

Considering the classification performances compared to the ground-truth, similar kappa values of 0.62 and 0.64 have been computed for *mahal* and *SFS*, respectively. As it can also be seen, κ ranges from 0.24 to 0.94 for *mahal* and from 0.32 to 0.81 for *SFS*, respectively. This shows that between-subject variation is quite high, reflecting important physiological differences between individuals, regardless of the employed feature selection algorithm. Also the AUC_{PR} of both algorithms are comparable with 0.59 (*mahal*) and 0.60 (*SFS*). A Wilcoxon signed rank test confirmed that the results are not significantly different ($p = 0.39$).

Figure 4 displays the learning curves obtained by varying the number of subjects in the data set. The overall performances (pooled LOSOCV results) achieved on the training and testing data sets with *mahal* converge rapidly from 5 subjects and stabilize around a κ of 0.65. *SFS* converges much slower. The performance gap of κ between the training and the testing remains higher than for *mahal*, even for 15 subjects. The performance on the training set remains slightly above 0.7, whereas the performance for both *mahal* and *SFS* on the testing set are at around 0.65.

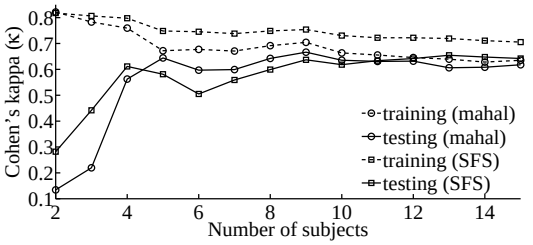

Figure 4: Learning curves obtained by varying the number of subjects in the training and testing sets.

4 DISCUSSION AND CONCLUSIONS

The feature selection step, essential to the proper design of a good classifier, usually suffers from an important methodological issue: in the presence of little available data, researchers often opt to perform feature selection on the complete data set. The feature selection algorithm proposed in this paper addresses this issue by offering the possibility of being integrated in a cross-validation procedure. The algorithm is fully automatic and, more importantly, does not require the choice of a desired number of features.

Figure 2 and Figure 3 do not reflect how many (different) features were chosen. Inspecting Table 1, a big difference is noticeable in the number of selected features. Where *mahal* selects about 10 features in average, with standard deviation (std) of less than 3 features, *SFS* selects 21 features with a standard deviation of nearly 7. The diversity of selected features after selection with *SFS* is very high whereas *mahal* consistently selects similar set of features during the different iterations of the cross-validation procedure, also with one small data set. In order to compare how consistently the feature selection algorithms were across the different iterations of the cross-validation procedure, we computed the mean number (and standard deviation) of iterations each feature was selected. The *mahal* algorithm chose 9.12 (5.42) and *SFS* 6.43 (4.59) number of iterations per selected feature in average. Only 17 different features for *mahal*, compared to 46 for *SFS*, were selected by the feature selection process. Furthermore, each feature is selected, in average, more times than with *SFS* which further illustrates how stable the selection procedure is to changes in the training set. A higher diversity of features mainly has two drawbacks. First, it is more difficult to choose a final set of features when design-

ing a classifier, since this seems to vary with every small change in the training set. Second, more features means higher feature extraction time. Despite the smaller number of selected features, the classification performance was not significantly affected.

The performance of a feature selection algorithm can also be described in terms of the total computational time that an algorithm needs to find the optimal feature set. Here, we only analyze the time needed by the feature selection itself. The feature extraction step is not taken into account. *mahal* is nearly 10 times faster than *SFS*, with 984 s and 9205 s, respectively. By design, on each iteration *SFS* must redo the entire classification step in the training set for each feature before choosing which feature to add to the feature set. The time increases approximately exponentially with each new feature added to the total feature set. In contrast, the computational time of the *mahal* algorithm increases approximately linearly as the performance calculations are only performed on the selected feature subsets (step 5 of the algorithm). In addition, the algorithm automatically restricts the list of features that have to be tested during selection by evaluating their statistical power in advance, i.e., the Mahalanobis distance and the Spearman's ranked-order correlation.

Our classifier achieves a performance of $\kappa = 0.62$ in distinguishing *sleep/wake*, which is at least as high as most work published so far, with fewer features used during classification. However, the differences in performance obtained for different subjects are too large to be ignored. It seems from the learning curves that this classifier is approaching its maximum performance with the currently extracted features. In order to further improve it, new approaches seem to be needed. These could take into account, or better yet, compensate for subject-specific differences in the physiological expressions of different sleep stages. Nevertheless, the feature selection algorithm *mahal* described in this paper seems well-suited for this problem since it is stable enough to be integrated in a cross-validation procedure, also in the presence of small data sets.

ACKNOWLEDGEMENTS

The authors thank Dr. Reinder Haakma and Sandrine Devot, as well as Prof. Ronald Aarts for their comments and careful reading of the manuscript.

REFERENCES

Abdullah, M. (1990). On a robust correlation coefficient. *The Statistician*, 39(4):455–460.

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46.

Davis, J. and Goadrich, M. (2006). The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd international conference on Machine learning ICML 06*, volume 10 of *ICML '06*, pages 233–240, Pittsburgh (USA). ACM Press.

Devot, S., Bianchi, A. M., Naujokat, E., Mendez, M., Brauers, A., and Cerutti, S. (2007). Sleep monitoring through a textile recording system. In *IEEE Engineering in Medicine and Biology Society*, volume 2007, pages 2560–2563.

Devot, S., Dratwa, R., and Naujokat, E. (2010). Sleep/wake detection based on cardiorespiratory signals and actigraphy. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS)*, pages 5089–5092. IEEE.

Duda, R. O., Hart, P. E., and Stork, D. G. (2001). *Pattern Classification*. Wiley, 2nd edition.

Fawcett, T. (2004). ROC Graphs: Notes and Practical Considerations for Researchers. *ReCALL*, 31(HPL-2003-4):1–38.

Friedman, J. H. (2012). Regularized Discriminant Analysis. *Journal of the American Statistical Association*, 84(405):165–175.

Haibo, H. and Garcia, E. (2009). Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284.

Long, X., Fonseca, P., Foussier, J., Haakma, R., and Aarts, R. (2012). Using Dynamic Time Warping for Sleep and Wake Discrimination. In *IEEE Engineering in Medicine and Biology Society - International Conference on Biomedical and Health Informatics (BHI)*, volume 25, pages 886–889, Hong Kong/Shenzhen (China).

Provost, F., Fawcett, T., and Kohavi, R. (1998). The case against accuracy estimation for comparing induction algorithms. In *Proceedings of the 15th International Conference on Machine Learning*, volume 445. JSTOR.

Rechtschaffen, A. and Bergmann, B. (1995). Sleep deprivation in the rat by the disk-over-water method. *Behavioural Brain Research*, 69(1-2):55–63.

Redmond, S. J., de Chazal, P., O'Brien, C., Ryan, S., McNicholas, W. T., and Heneghan, C. (2007). Sleep staging using cardiorespiratory signals. *Somnologie*, 11(4):245–256.

Whitney, A. (1971). A Direct Method of Nonparametric Measurement Selection. *IEEE Transactions on Computers*, C-20(9):1100–1103.

Zoubek, L., Charbonnier, S., Lesecq, S., Buguet, A., and Chapotot, F. (2007). Feature selection for sleep/wake stages classification using data driven methods. *Biomedical Signal Processing and Control*, 2(3):171–179.