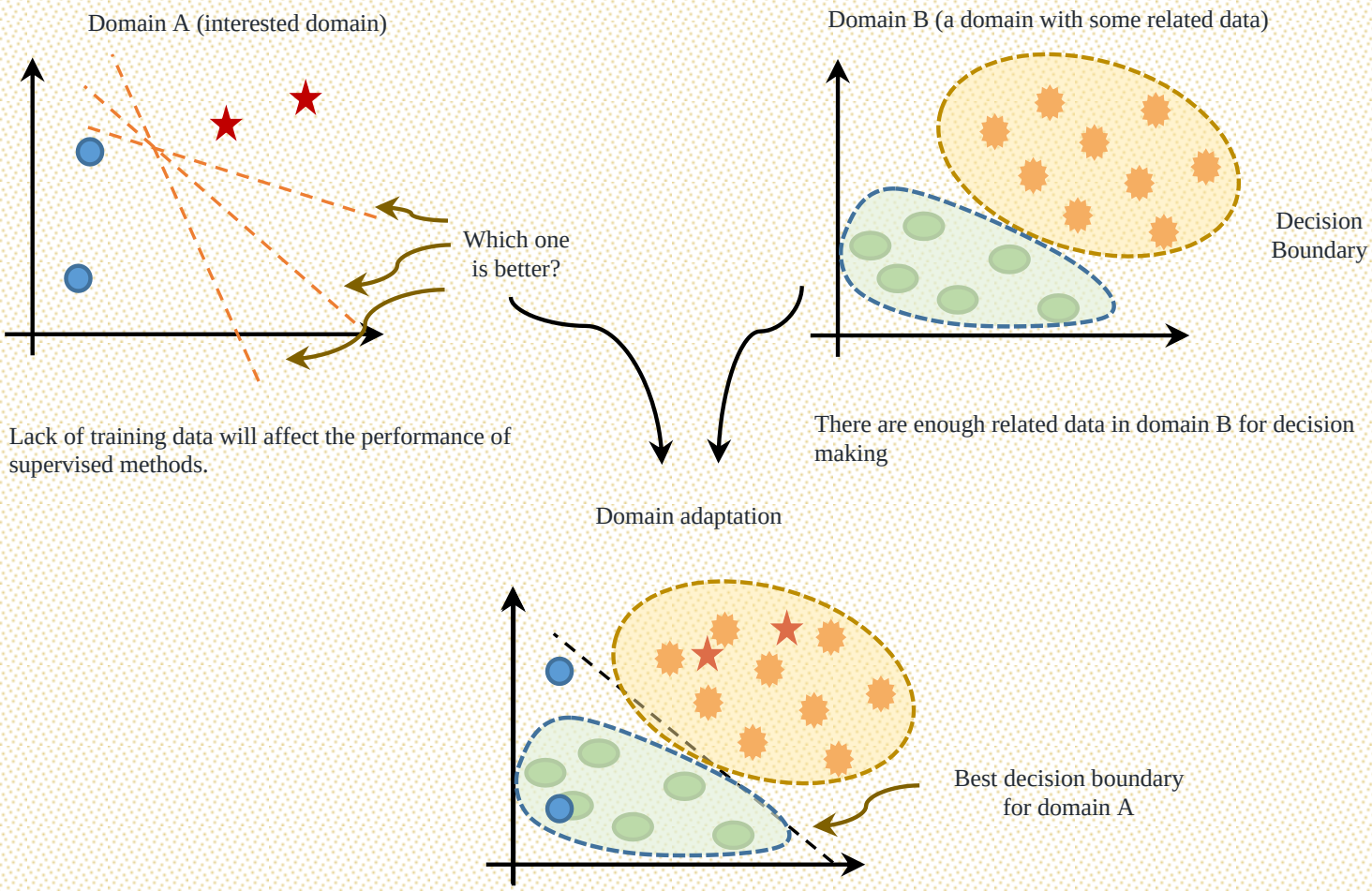

Sirvan Khalighi

Learning Under Distribution Mismatch Applied in Biosignal Processing

Tese de Doutoramento em Engenharia Electrotécnica e Computadores, ramo de especialização em Automação e Robótica, orientada pelo Professor Doutor Urbano José Carreira Nunes e Pela Professora Doutora Bernardete Ribeiro e apresentada à Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Maio de 2016

· U C ·

Universidade de Coimbra

Electrical and Computer Engineering Department
Faculty of Science and Technology
University of Coimbra

Learning Under Distribution Mismatch Applied in Biosignal Processing

Sirvan Khalighi

May, 2016

Electrical and Computer Engineering Department
Faculty of Science and Technology
University of Coimbra

Learning Under Distribution Mismatch Applied in Biosignal Processing

A dissertation submitted to the Department of Electrical and Computer Engineering of the Faculty of Science and Technology of the University of Coimbra in partial fulfillment of the requirements for the Degree of Doctor of Philosophy.

Sirvan Khalighi

Under supervision of
Prof. Dr. Urbano Nunes (advisor)
Profa. Dr. Bernardete Ribeiro (co-advisor)

"Ask yourself only what are the facts and what is the truth that the facts bear out."

Bertrand Russell

To my beloved father who's no longer with me (R.I.P)...

Acknowledgements

I would like to express my sincere gratitude to my adviser, Prof. Dr. Urbano Nunes, for his enormous amount of support, encouragement, and guidance throughout my Ph.D. studies; to my coadviser, Profa. Dra. Bernardete Ribeiro for her favors, helps and review on the statistical machine learning contents; I also wish to express my gratitude for the Institute of Systems and Robotics, University of Coimbra (ISR-UC) for its logistic support.

This research work was supported by a Ph.D. scholarship SFRH/BD/81828/2011, granted by the Portuguese Government's department FCT- Fundação para a Ciência e a Tecnologia, by QREN funded project "SLEEPTIGHT: with reference CENTRO-01-0202-FEDER-011530 ", and FCT/COMPETE projects "AMS-HMI12: Assisted mobility supported by shared-control and advanced human-machine interfaces, with reference RECI/EEI-AUT/0181/2012", and UID/EEA/00048/2013.

Data of ISRUC-Sleep dataset are a part of the data made available by the Laboratory of Sleep from Hospital Centre of Coimbra (CHUC). I am grateful to this Laboratory, mainly to Dr. José Moutinho Santos, to the sleep experts, and my colleague Dulce Oliveira for collecting the records and doing some analysis on the experimental results. I am deeply grateful to all of them. I am very grateful to my all friends, colleagues and professors from ISR, and Special thanks go to Dr. Hadi Aliakbarpour, Fatemeh Pak, Parisa Tirdad, Dr. Gabriel Pires and Teresa Sousa for their support and fruitful discussions during my Ph.D. Finally, I would like to thank my dearest family especially my wife Bahareh for every thing.

Abstract

Bioelectrical signals, which record brain activity, are among the complex dynamic signals due to the strong non-stationarity effects of brain and subject dependency. In biosignal-based classification, training samples and unlabeled test samples are gathered in different recording sessions or from different subjects, yielding two common problems: 1) changes in the probability distributions of training and test instances, which are caused by the non-stationary of brain signals 2) the lack of sufficient labeled training data for each test subjects.

Sleep staging using biosignals is an essential part of the diagnostic process in the assessment of sleep disorders. Several studies have reported the development of automatic sleep stage classification (ASSC) methods using the polysomnographic (PSG) records. The current methods typically assume that, the labeled training data comes from the same probability distribution as the test data. The ASSC is a challenging problem, due to the noisy signals, the subjects' variability, the experts' labeling differences, and the signals complexity, mainly in cases of sleep disorders. Therefore, due to these challenges, the standard learning methods are no longer consistent and they generally yield a drop in performance. In this thesis, aiming to improve the applicability of automatic sleep staging some efficient methods were proposed as follows. First, an efficient subject-independent method is proposed with application in sleep–wake detection and in multiclass sleep staging (awake, non-rapid eye movement (NREM) sleep and rapid eye movement (REM) sleep). To find the best combination of PSG signals for automatic sleep staging, six electroencephalographic (EEG), two electrooculographic (EOG), and one electromyographic (EMG) channels were analyzed. An extensive set of feature extraction techniques were applied, covering temporal, frequency and time–frequency domains. The extracted feature set was transformed and normalized to reduce the effect of extreme values of features. The most discriminative features were selected through a two-step method, composed by a manual selection step based on features' histogram analysis followed by an automatic feature selector.
Second, to overcome the limitations of the subjects and sessions' variability, domain adaptation methods were exploited. In particular, to alleviate the significant mismatch between source and target domains, importance weighting import vector machine (IWIVM), which is an adaptive classifier, was proposed. This adaptive probabilistic classification method, which is sparse and computationally efficient, can be used for unsupervised domain adaptation. Despite the sparseness, the proposed method outperforms the state-of-the-art in both unsupervised and semisupervised domain adaptation scenarios. We also introduce a reliable importance weighted cross validation (RIWCV), which is an improvement of importance weighted cross validation (IWCV), for parameter and model selection. The RIWCV avoids falling down in local minima, by selecting a more reliable combination of the parameters instead of the best parameters.

Third, to facilitate the performance comparison of the new methods for sleep patterns analysis, we introduced an open-access comprehensive sleep dataset, called ISRUC-Sleep. The data were obtained from human adults, including healthy subjects, subjects with sleep disorders, and subjects under the effect of sleep medication. Each recording was randomly selected between PSG recordings that were acquired by the Sleep Medicine Centre of the Hospital of Coimbra University (CHUC). The dataset comprises three groups of data: 1) data concerning 100 subjects, with one recording session per subject; 2) data gathered from 8 subjects; two recording sessions were performed per subject, and 3) data collected from one recording session related to 10 healthy subjects. The Polysomnography (PSG) recordings, associated with each subject, were visually scored by two human experts.

This dataset was created aiming to complement existing datasets by providing easy-to-apply data collection with some characteristics not covered yet. ISRUC-Sleep can be useful for analysis of new contributions: (i) in biomedical signal processing; (ii) in development of ASSC methods; and (iii) on sleep physiology.

In addition, due to the similarity of the challenges and the importance of biometric-based recognition, we have also studied the same challenges in the area of iris recognition. The conventional iris recognition methods do not perform well for the datasets where the eye image may contain non-ideal data such as specular reflection, off-angle view, eyelid, eyelashes and other artifacts. We proposed a reliable iris recognition method using a new scale-, shift- and rotation- invariant feature-extraction method in time-frequency and spatial domains. Indeed, a 2-level nonsubsampled contourlet transform (NSCT) was applied on the normalized iris images and a gray level co-occurrence matrix (GLCM) with 3 different orientations was computed on both the spatial image and the NSCT frequency subbands. Moreover, the effect of the occluded parts was reduced by performing an iris localization algorithm followed by a four regions of interest (ROI) selection. The proposed iris identification method was tested on the public iris datasets CASIA Ver.1 and CASIA Ver.4-lamp showing a state-of-the-art performance.

Resumo

Os sinais eléctricos cerebrais são sinais dinâmicos e complexos devido à sua não-estacionariedade e à sua variabilidade inter-sujeito. A classificação automática baseada nestes sinais biológicos inclui amostras para treino e amostras para teste que podem ser adquiridas em sessões diferentes ou de participantes diferentes, levando a que ocorram com frequência dois tipos de problemas: 1) diferenças entre as distribuições de probabilidade das amostras de treino e teste causadas pela não-estacionaridade dos sinais cerebrais; 2) falta de amostras de treino identificadas para cada sujeito de teste.

O estadiamento do sono tem como base as características de alguns sinais biológicos e é essencial no diagnóstico de patologias do sono. Encontram-se descritos na literatura vários estudos que visam a automatização deste processo usando dados de polissonografia (PSG). Os métodos actuais assumem que a distribuição de probabilidade dos dados de treino e dos dados de teste é semelhante. Para além disso, a classificação automática dos estadios do sono enfrenta também dificuldades relacionadas com o ruído dos sinais, com a variabilidade dos sinais entre sujeitos, com as diferenças na classificação feita pelos especialistas e com a complexidade dos sinais, sobretudo em dados adquiridos em pacientes com patologias de sono. Assim, devido a este vasto leque de desafios, os métodos tradicionais de aprendizagem automática apresentam várias limitações que precisam de ser colmatadas para que o seu desempenho global na classificação dos estadios do sono seja melhorado.

Esta tese apresenta um método eficaz para a classificação automática das diferentes etapas do sono que visa aumentar a aplicabilidade deste tipo de algoritmos.

Primeiro, é proposto um método de classificação independente das características individuais de cada sujeito com aplicação na detecção do estar acordado versus estar a dormir e com aplicação na classificação dos diversos estadios do sono (acordado, sono não-REM (NREM) e sono REM). Foram analisados seis canais electroencefalográficos (EEG), dois canais electrooculográficos (EOG) e um canal electromiográfico (EMG) para estudar a combinação de sinais PSG que melhores resultados permite na classificação automática do sono. Para tal, aplicou-se um leque extensivo de técnicas de extracção de características nos domínios de tempo, frequência e tempo-frequência. O grupo de características extraídas foi transformado e normalizado para que efeito dos valores extremos fosse atenuado. As características mais discriminativas foram depois seleccionadas através de dois passos: o primeiro consistiu numa selecção manual baseada no histograma das características extraídas e o segundo consistiu num selector automático.

Em segundo, foram explorados métodos adaptativos para superar as limitações devido à variabilidade do sinal entre sujeitos. Em particular, foi proposto um classificador adaptativo, o importance weighting import vector machine (IWIVM), para atenuar as diferenças entre os

domínios da fonte e do alvo. Este método de classificação com modelo probabilístico adaptativo, esparso e computacionalmente eficiente, pode ser usado de forma não-supervisionada. O método proposto supera o estado da arte quer seja usado de forma supervisionada ou de forma não-supervisionada. Foi também apresentada uma versão melhorada do método de validação cruzada com importância ponderada, o reliable importance weighted cross validation (RIWCV), para a selecção de parâmetros e modelos. Este método evita os mínimos locais seleccionando a combinação mais fiável de parâmetros em vez de seleccionar apenas os melhores parâmetros.

Em terceiro, para facilitar a comparação entre o desempenho dos métodos de análise automática dos padrões do sono, foi reunido e disponibilizado um conjunto de dados PSG ao qual se chamou ISRUC-Sleep Dataset. Foram incluídos dados de adultos saudáveis ou com patologias de sono, que podiam estar ou não sob efeito de medicação. Cada registo PSG incluído foi aleatoriamente seleccionado entre os dados adquiridos no Centro de Medicina do Sono do Centro Hospitalar da Universidade de Coimbra (CHUC). Os dados foram organizados em três grupos: 1) dados de 100 sujeitos com um registo PSG cada; 2) dados de 8 sujeitos com dois registos PSG cada, adquiridos em sessões diferentes; 3) dados de 10 sujeitos saudáveis com um registo PSG cada. Os estadios de sono registados em cada PSG incluído foram classificados por duas vezes, cada uma delas a partir da apreciação visual de um especialista diferente.

Este conjunto de dados foi recolhido com o objectivo de complementar outros já existentes, incluindo para isso informações que até agora não eram disponibilizadas. Espera-se que o ISRUC-Sleep Dataset possa trazer novas contribuições no processamento de sinais biológicos (i), no desenvolvimento de novos métodos de classificação automática do sono (i) e nos estudos da fisiologia do sono.

Para além disso, tendo em conta as similaridades de desafios/dificuldades no reconhecimento automático da iris e a importância que este pode ter no reconhecimento biométrico, foram também exploradas formas de melhorar os métodos de classificação nesta área. Os métodos convencionais apresentam várias limitações quando os dados disponibilizados não estão nas condições ideais e contêm alguns artefactos como reflexão especular, pálpebras ou cílios, por exemplo. Assim, para um reconhecimento automático e eficiente da iris, é proposto um novo método que extrai características no domínio do tempo-frequência e no domínio espacial com escala, rotação e deslocamento invariáveis. Foi aplicada uma transformada de Contourlet sem subamostragem de dois níveis nas imagens normalizadas da iris e estimada uma matriz de co-ocorrência do nível de cinzento com três direcções diferentes para as bandas de frequências resultantes da transformada e para as imagens espaciais. Para além disso, o efeito das partes oclusas foi minimizado através da aplicação de um algoritmo de localização da iris seguido da selecção de quatro regiões de interesse. O método proposto para a identificação da iris foi testado usando os conjuntos de dados CASIA Ver.1 e CASIA Ver.4-lamp publicamente disponíveis, demonstrando bom desempenho.

Contents

List of Figures

List of Tables

Acronyms

AASM	The American Academy of Sleep Medicine
ACC	Accuracy
ANN	Artificial Neural Networks
AR	Auto Regressive
ASSC	Automatic Sleep Stage Classification
ASVM	Adaptive SVM
BCR	Balanced Correction Rate
BER	Balanced Error Rate
CAP	The Cyclic Alternating Pattern
CCA	Canonical Correlation Analysis
CT	Contosurlet Transform
CV	Cross Validation
DA	Domain Adaptation
DEFS	Differential Evolution Feature Selection
DWT	Discrete Wavelet Transform
ECG	Electrocardiogram
EEG	Electroencephalogram
EER	Equal Error Rate
EM	Expectation Maximization
EMG	Electromyogram
EOG	Electrooculogram
ERM	Empirical Risk Minimization
FCBF	Fast Correlation Based Feature Selection
FFT	Fast Fourier Transform
FN	False Negatives

FP	False Positives
GAD	Generalized Anxiety Disorders
gfk	Geodesic Flow Kernel
GLCM	Gray Level Co-occurrence Matrix
HDA	Heterogeneous Domain Adaptation
HHT	Hilbert Huang Transform
i.i.d	Independent and Identical Distribution
ISRUC	Institute of Systems and Robotics, University of Coimbra
IWCV	Importance Weighted Cross Validation
IWERM	Importance Weighted ERM
IWIVM	Importance Weighted Import Vector Machine
KLIEP	Kullback-Leibler Importance Estimation Procedure
KLR	Kernel Logistic Regression
KMM	Kernel Mean Matching
LDA	Linear Discriminant Analysis
LOOCV	Leave-One subject-Out Cross-Validation
LSIF	Least-Squares Importance Fitting
MDP	Markov Decision Process
MLLR	Maximum Likelihood Linear Regression
MMD	Maximum Mean Discrepancies
MODWT	Maximal Overlap Discrete Wavelet Transform
mRMR	minimal-Redundancy and Maximal-Relevance
NB	Naïve Bayes
NLL	Negative Logarithmic Likelihood
NLP	Natural Language Processing
NRI	Noradrenaline Reuptake Inhibitor
NREM	Non-Rapid Eye Movement
NSCT	Nonsubsampled Contourlet Transform
NSDFB	Nonsubsampled Directional Filter Bank
NSLP	Nonsubsampled Laplacian Pyramids
OSA	Obstructive Sleep Apnea
PCA	Principal Component Analysis
PSG	Polysomnography

REM	Rapid Eye Movement
RIWCV	Reliable Importance Weighting Cross Validation
R&K	Rechtschaffen and Kales Standard
ROC	Receiver Operating Characteristic
ROI	Regions of Interest
RSP	Relative Spectral Power
RuLSIF	Relative uLSIF
RVM	Representative Vector Machine
SAS	Sleep Apnea Syndrome
SBFS	Sequential Backward Floating Selection
SBS	Sequential Backward Selection
SBSP	Sub-band Spectral Power
SCL	Structural Correspondence Learning
SENS	Sensitivity
SFFS	Sequential Forward Floating Selection
SFS	Sequential Forward Selection
SNR	Signal-to-Noise Ratio
SNRI	Selective Noradrenaline Reuptake Inhibitors
SOFM	Self Organizing Feature Map
SPEC	Specificity
SRM	Structural Risk Minimization
SSM4S	Sirvan Supervised Method for Sleep Staging
SSRI	Selective Serotonin Reuptake Inhibitors
SVM	Support Vector Machine
TN	True Negatives
TP	True Positives
uLSIF	unconstrained-LSIF

Chapter 1

Introduction

This chapter provides an overview of the thesis. It introduces biosignal processing, sleep staging and domain adaptation. Section 1.2 describes the current challenges of biosignal processing and the thesis motivations. Section 1.3 gives an overview of how this thesis proposes to overcome these challenges. It also outlines the methods developed in the context of this thesis.

1.1 Domain Adaptation for Biosignal Processing

Biosignals, consisting of electrical and non-electrical signals, are refereed to all kinds of signals that provide a description or information regarding biological beings. Bioelectrical signals are usually related in electric currents produced by the sum of electrical potential differences across a specialized tissue, organ or cell system like the nervous system. The Electroencephalogram (EEG), Electrocardiogram (ECG), Electromyogram (EMG) and Electrooculogram (EOG) are the most common and essential biosignals used in real-time telehealth-monitoring systems, advanced clinical decision-support systems, mobile biosignal analyzers, seizure epileptic detectors, and sleep disorders detectors.

The non-stationarity and complexity of biosignals, the patterns variability of the signals for different subjects specially for the patient subjects, and different types of noises are prominent challenges of biosignal processing. It is commonly accepted that the existing supervised machine learning techniques are not accurate and reliable enough to be routinely used [4]. Therefore, the development of reliable methods that are robust to noises and adaptive to new subjects are highly desirable.

The performance of the classical supervised learning methods are highly affected by the number of training data. Moreover, these methods typically assume that test instances come from the same probability distribution as the labeled training instances, which created the learning model. However, in real world problems, such as health monitoring and diagnosis applications, due to the aforementioned factors, the assumption of the same probability distribution of training and test data does not hold and there are some differences between training and testing instances. Therefore, many common theoretical guarantees and bounds on the error of predictions are no longer valid and the standard methods do not work well in practice. Domain adaptation methods attempt to alleviate difference between train and test by identifying and transferring the relevant useful knowledge, and by generalizing the created models from training (source) domain to testing (target) domain. Therefore, it is highly desirable to develop methods for biosignal analysis that are robust to the variations and adaptable to the new subjects or to new recordings.

Sleep and Sleep Staging: Sleep is an active and regulated process with an essential restorative function for physical and mental health [5]. Sleep occupies approximately one-third of our lifetime, and in addition to nutrition, fitness and emotional states, it plays an important role in human wellness. Quality of sleep has an important effect on the health and quality of life. Sleep staging is an essential part of the diagnostic process in the assessment of sleep disorders such as Sleep Apnea Syndrome (SAS) [6]. Therefore, study of individual behaviors during sleep, such as monitoring, scoring and detecting abnormal changes of sleep pattern through all-night polysomnogram (PSG) recordings have consistently been an important research topic. The manual sleep stage classification is a labor intensive task that involves the interpretation, by an expert, of numerous signals captured in all-night PSG session. Accordingly, an efficient automatic sleep stage classification (ASSC) method may save time and provide an objective assessment of sleep, independent of subjective interpretation of experts. Several studies have reported the development of ASSC methods using analysis of PSG records, including EEG records in combination with EOG and EMG records collected from human individuals. The PSG signals used for ASSC can be obtained either through invasive or non-invasive recording methods. The implanted electrodes in invasive methods, allow for a very accurate reading of the electrical activity over a small portion of the subject. However, non-invasive methods are attractive and practical since, they do not require any surgical procedures, and the performance has shown to be comparable to implanted electrodes when a more sophisticate

adaptive algorithm is used [7]. Since in non-invasive methods, the electrical potentials must pass through the skull, the EEG signals are inherently very noisy. The ASSC is a challenging biosignal analysis problem, due to the subjects and sessions' variability, the experts' labeling differences, and the signals complexity, mainly in cases of sleep disorders. Giving the aforementioned challenges and characteristics, the ASSC problem was investigated as a biosignal analysis application.

1.2 Challenges and Motivations

This thesis deals with several challenges in the areas of 1) biosignal processing, specially in ASSC, 2) domain adaptation, and 3) IRIS recognition. The main goal is to overcome the challenges, which summarized as follows:

- State-of-the-art of ASSC shows that sleep segments (epochs) of healthy subjects, free of sleep stage transitions without disagreement between experts classification, can be automatically classified using a low number of features. However, on ambiguous sleep epochs (e.g., sleep transitions) the agreement level between the scoring by automatic classification and the human expert is only around 60% [8]. The ambiguity and similarity of some the PSG signals is one of the main challenges in automatic sleep staging. Therefore, development of automatic sleep stage classification methods, which are robust to noise and ambiguous pattern, is highly desirable.

- In ASSC, several PSG channels are analyzed, simultaneously, thus extraction and selection of the best spatiotemporal synchronization patterns, which reliably represent the subject's conditions, are one of the main issues in ASSC. Moreover, selection of the best channels for applications of sleep-wake and multiclass sleep staging is another challenging issue in ASSC.

- Numerous methods have been developed for automatic detection of arousals, apnea, and sleep stages [9–12]. These methods often use PSG recordings, including electrophysiological signals (electrocardiographic activity, brain-wave patterns, eye movements, and activation signal of muscles), pneumological signals (airflow, blood oxygen level, and movement of respiratory muscles), and other contextual information (body position, lights, snore recording, etc.)[13]. These signals have been collected from human individuals using noninvasive surface electrodes. To evaluate the efficiency of automatic sleep

pattern analysis methods, non-public and few existing public datasets have been used. Rigorous comparisons between the developed methods cannot be done since the used datasets differ in recording conditions, physiological conditions of subjects and number of assessed subjects. To facilitate the performance comparison of the new methods for sleep patterns analysis, datasets with quality content, publicly-available, are necessarily, very important and useful.

- The PSG signals are among the complex dynamic signals due to the strong non-stationarity effects of brain and subject dependency [14]. The features extracted from biosignals may vary due to the diseases and health problems, recording environment changes, subjects' physical conditions, and the electrode displacement. In the PSG-based classification, training and test samples are gathered from different subjects or from different recording sessions. Due to the non-stationary and other effective factors, PSG signals may change, which causes a difference in the distributions between training and test data. To alleviate the difference, a few studies have been conducted on adaptive ASSC systems with positive results [15]. However, these adaptive methods, which are based on supervised learning techniques, utilized labeled test samples, so they are costly and impractical for real world applications [16].

  In ASSC, the distribution of training and test instances are different but related to each other in some sense, thus it is possible to estimate the test probability distribution via the training set. As concerns, it is highly desirable to develop the ASSC methods that are robust to the variations and adaptable to the new subjects or to the new recordings. One of the assumptions is to alleviate the difference between train and test domains based on covariate shift adaptation [17]. Covariate shift adaptation is a method which overcome this shortcoming, assuming that the input distributions of training and testing sessions are different while the conditional distribution of output given input remains unchanged [17].

In addition, due to the similarity of the challenges and the importance of biometric-based recognition, we have also studied the same challenges in the area of iris recognition. The main iris recognition challenges are as follows:

- Iris recognition is a reliable and accurate biometric identification technology due to the uniqueness, aging invariant and noninvasive characteristics of iris. Moreover, this is a noncontact data acquisition technology. The conventional iris recognition methods do

not perform well for the datasets where the eye image may contain nonideal data such as specular reflection, off-angle view, eyelid, eyelashes and other artifacts. Therefore, development of reliable iris recognition method that are scale-, shift- and rotation-invariant is highly desirable.

1.3 Contributions

The thesis provides a set of methods to deal with the challenges described in the previous section. In summary, the main contributions of this thesis are as following:

- A new ASSC method has been developed, aiming to improve sleep stage classification accuracy in two applications: sleep-wake detection and multiclass sleep staging classification. The effectiveness of our approach is demonstrated through a series of experiments involving PSG data from our extensive dataset of 128 different subjects with confirmed or only suspicious sleep disorders which was collected in central hospital of Coimbra [18–20].

- A new time-frequency based feature extraction method, for ASSC was proposed. To decompose the PSG signals in different resolutions, the maximum overlap discrete wavelet transform (MODWT), which is shift invariant transform, is employed. Moreover, due to importance and usefulness, several temporal and frequency feature extraction methods, which represent other aspects of the signals, have been researched [18–20].

- Some works such as [12, 21] used just one or more EEG channels, whereas others [5, 22, 23] used EEG channels in combination with EOG and EMG channels. Therefore, to reduce the computational cost and improve classification performance, a systematic analysis for finding the best combination of EEG, EOG and EMG channels, for both application sleep-wake detection and multiclass sleep staging, is performed [20, 24].

- Current automatic feature selectors are classifiers' dependent; moreover, they are affected by extreme values in feature vectors. Therefore, to find the most discriminative features for sleep-wake detection and multiclass sleep staging a two-step feature selector is applied on the transformed and normalized feature vectors. This two-step algorithm is composed by a manual selection followed by an automatic selector. For the second part of the algorithm, six different feature selectors are exploited [20, 24].

- Importance weighted import vector machine (IWIVM), an adaptive probabilistic classification method, based on direct importance estimation, is proposed for unsupervised domain adaptation [25]. This instance-weighting adaptation method, which is sparse and computationally efficient, can be used for asymptotically canceling the bias caused by covariate shift [26].

- Aiming to improve the performance of model selection under covariate shift, reliable importance weighting cross validation (RIWCV), a modified version of importance weighting cross validation, is introduced. To find the optimal parameters under covariate shift, this modified cross validation strategy, attempts to select the most reliable parameters in source domain instead of the best parameters [25].

- Datasets with quality content, publicly available, are an important vehicle for accelerating research, since they facilitate the performance comparison of new approaches and methods. We introduce a publicly-available comprehensive sleep dataset, called ISRUC-Sleep, which comprises three subgroups. The subgroups of the dataset contain PSG signals of different adult individuals, including healthy subjects (subgroup-III), subjects with sleep disorders, and subjects under the effect of sleep medication (subgroup-I). Sleep stages were labeled by two sleep experts. Furthermore, for eight subjects (subgroup-II), two sets of PSG data that have been recorded at different dates, are provided [27].

- Aiming to evaluate and compare new contributions, which will use this dataset as a benchmark, results of applying a subject-independent ASSC method on ISRUC-Sleep dataset are presented. This supervised-learning based method, detailed in Khalighi et al. [20], is henceforth named *SSM4S*[1] [27]

- An adaptive ASSC method has been developed, aiming to improve the applicability of automatic sleep staging with two applications: sleep-wake detection and multiclass sleep staging. The main goal of this method, which is based on unsupervised covariate shift adaptation, is to alleviate the subject's variability of the training and test subjects [28].

Moreover, to address the highlighted issues in the context of iris recognition, the following contributions are presented:

- An overall contribution: A new scale-, shift- and rotation-invariant iris recognition method, in time-frequency and spatial domains is proposed. The effectiveness of our

[1] Sirvan Supervised Method for Sleep Staging

approach is validated, through a set of experiments using CASIA dataset Ver.1 and Ver.4-lamp [29].

- Iris segmentation: to determine the pupil region, among labeled regions in binary eye image, a pupillary boundary detection method is proposed. The contribution here is a new way to select the pupil region that consists on selection of a region with the largest area and the smallest eccentricity in the binary image. Moreover, in some of the state-of-the-art methods only the upper and/or lower part of the iris image (texture) is used to remove the occluded regions by the eyelid and eyelashes, which results in loss of significant "information". Therefore, to mitigate this problem after detection of limbic boundary, a four-ROIs selection method is proposed [29, 30].

- Feature extraction: after normalizing the selected regions of interest some textural features are extracted from the gray level co-occurrence matrix (GLCM). The GLCM is calculated on both spatial image and frequency subbands of NSCT decomposition. Moreover, numerical features are calculated directly on NSCT frequency and spatial iris image [29, 30].

- Feature selection: to reduce the influence of extreme values, the extracted features are transformed, normalized and then fed in to our feature selector. Selection of features using well known automatic feature selectors is not accurate enough to get the best results; most of these feature selectors, select feature elements from all the feature-types which yield inaccurate selection. In order to obtain a more accurate selection and further reduce the number of extracted features, a new two-step feature selection process, which consists of filtering and a wrapper phases, is proposed. In the first step, some of the feature-types are removed using a simple filtering algorithm and then, in the second step, the minimal-redundancy and maximal-relevance (mRMR), which is a wrapper based feature selector, is applied [29, 30].

1.4 Organization of the Thesis

This thesis covers several methods related to biosignal processing, automatic sleep stage classification, domain adaptation and iris recognition. As summarized in Fig. 1.1, this thesis is organized in the following chapters. Chapter 2 provides the necessary backgrounds of the statistical learning methods with a focus on distribution mismatch of data. In Chapter 3

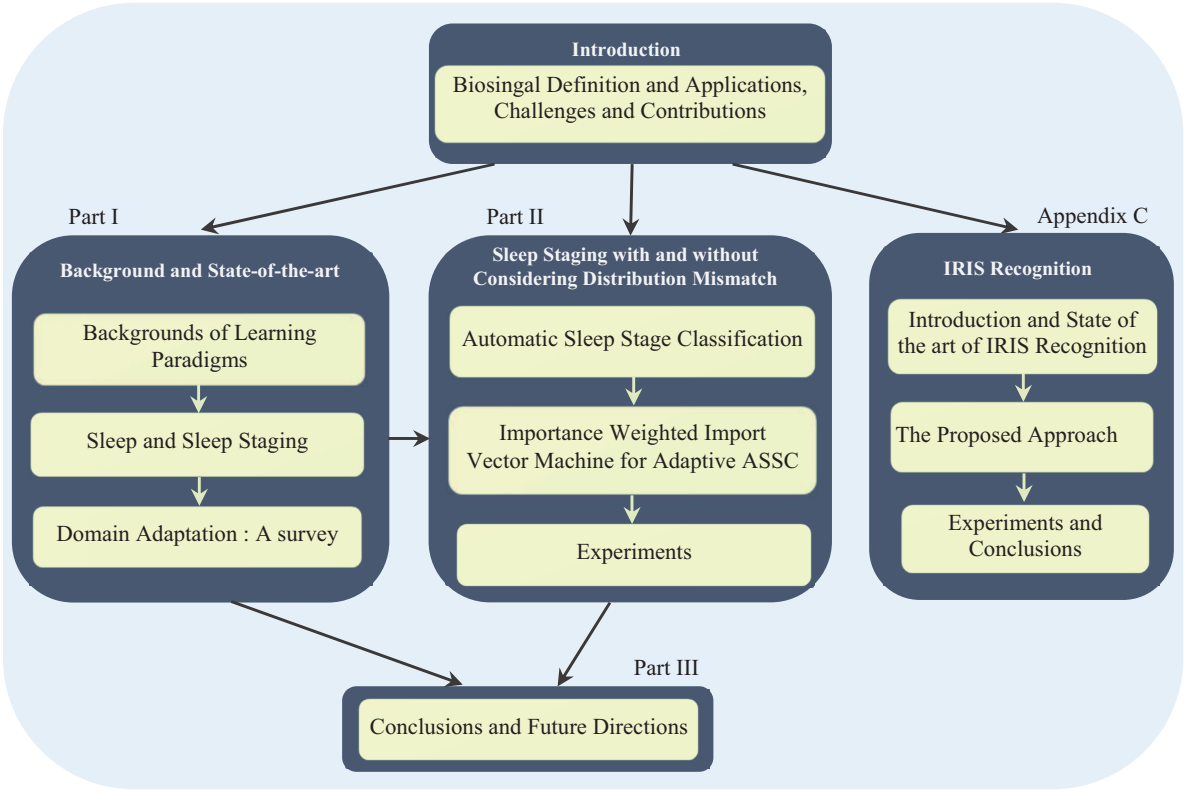

FIGURE 1.1: Structure of the Thesis. Part-II and Appendix C consist on the contributions on the areas of sleep staging, domain adaptation and iris recognition, respectively.

the basic terminology for the understanding of sleep staging is introduced. Next, a state-of-the-art of ASSC methods, is presented. We provide a survey on domain adaptation in Chapter 4, and focus on the methodologies developed in the literature, regardless of their specific application domains. The Second part of this document, details each of the stated contributions. An efficient subject-independent method, based on supervised learning, with applications in sleep-wake detection and in multiclass sleep staging is proposed in Chapter 5. Importance weighting import vector machine (IWIVM), which is an unsupervised domain adaptation method, is proposed in Chapter 6. Moreover, this chapter presents an adaptive ASSC method, based on unsupervised covariate shift adaptation. The experimental results and analysis are reported in Chapter 7. During the experiments, the developed methods have been evaluated on the benchmark datasets. A conclusion of this thesis and the remarks to the future directions are summarized in Chapter 8. In addition to these chapters, a brief report of our contributions on the iris recognition problem is provided in Appendix C.

1.5 Related Published Works

Some sections of the next chapters as well as some of the results in this thesis, have also been presented in the following publications.

- **Sirvan Khalighi**, Bernardete Ribeiro, Urbano Nunes, *Importance Weighted Import Vector Machine for Unsupervised Domain Adaptation*, IEEE Transactions on Cybernetics, IEEE, 2016.
- **Sirvan Khalighi**, Teresa Sousa, Jose Moutinho Santos, Urbano Nunes, *ISRUC-Sleep: A comprehensive public dataset for sleep researchers*, Computer Methods and Programs in biomedicine, Elsevier, 2015.
- Teresa Sousa, Aniana Brito, **Sirvan Khalighi**, Gabriel Pires, Urbano Nunes, *A two-step automatic sleep stage classification method with dubious range detection*, Computers in Biology and Medicine, Elsevier, 2015.
- **Sirvan Khalighi**, Fatemeh Pak, Parisa Tirdad, Urbano Nunes, *Iris Recognition using Robust Localization and Nonsubsampled Contourlet-Based Features*, The Journal of Signal Processing Systems, Springer Verlag, 2014.
- **Sirvan Khalighi**, Teresa Sousa, Gabriel Pires, Urbano Nunes, *Automatic sleep staging: a computer assisted approach for optimal combination of features and polysomnographic channels*, The Journal of Expert Systems with Applications, Elsevier, 2013.
- **Sirvan Khalighi**, Teresa Sousa, Urbano Nunes, *Adaptive Sleep Stage Classification under Covariate Shift*, In 34rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 12), San diego, Sept. 2012.
- **Sirvan Khalighi**, Parisa Tirdad, Fatemeh Pak, Urbano Nunes, *Shift and Rotation Invariant Iris Feature Extraction Based on Non-Subsampled Contourlet Transform and GLCM*, In the International Conference on pattern recognition applications and methods, (ICPRAM), Feb. 2012.
- Teresa Sousa, Dulce Oliveira, **Sirvan Khalighi**, Gabriel Pires, Urbano Nunes, *Neurophysiological and Statistical Analysis of Failures in Automatic Sleep Stage Classification*, In the International Conference on bio-inspired systems and signal processing (Biosignals 2012), 2012.

- **Sirvan Khalighi**, Teresa Sousa, Dulce Oliveira, Gabriel Pires, Urbano Nunes, *Efficient Feature Selection for Sleep Staging Based on Maximal Overlap Discrete Wavelet Transform and SVM*, in the 33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 11), Boston, Sept. 2011.

Part I

Background and State-of-the-art

Chapter 2

Backgrounds of Learning Paradigms

Statistical machine learning methods are now very popular in a wide area of science. This chapter provides the necessary backgrounds of the statistical learning methods with a focus on distribution mismatch of data. First, in Section 2.1 a review on classical machine learning methods is provided. These methods are developed based on the essential assumption of independent and identical distribution (i.i.d). Next, a brief review of the distribution mismatch problem on real world applications is summarized in Section 2.2. Finally, the characteristics of learning methods and their differences to deal with the distribution mismatch problem are provided.

2.1 Learning Without Considering Distribution Mismatch

Machine learning is a branch of artificial intelligence, which aims to study mathematical foundations and practical applications of systems that act based on past experience instead of being explicitly programmed [31]. Classical machine learning methods estimate the models from data to predict on data. The predictions (of test data) can only be successful if the estimated models from the training data can be generalized on testing data.

The *independent and identical distribution* (*i.i.d.*) is the main assumption in theoretical models of the typical learning systems. Indeed, based on *i.i.d.*, the training and test data follow the same probability distributions.

Depending on the objectives and the types of label information, machine learning algorithms without considering distribution mismatch are conventionally categorized into following types:

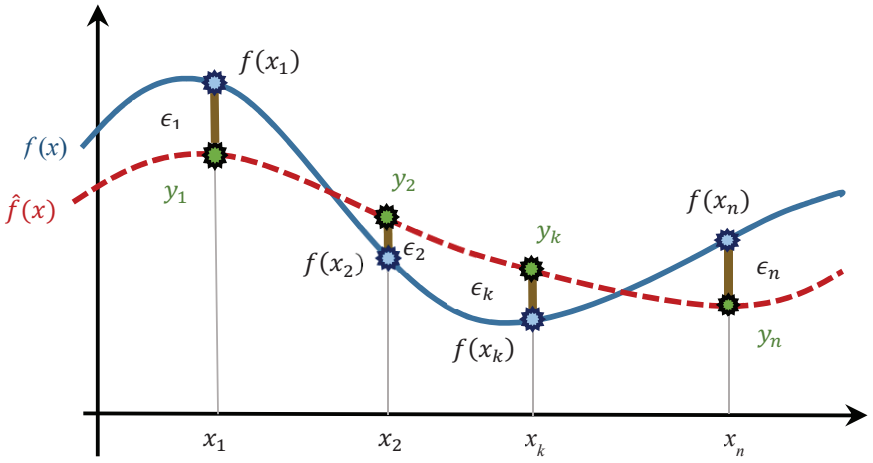

FIGURE 2.1: Supervised learning methods learn a function $f(x)$ from the samples; $f(x)$ and $\hat{f}(x)$ denote the target function and the learned function, respectively.

Supervised learning, Unsupervised learning, Semisupervised learning and Reinforcement learning, which will be described in next.

2.1.1 Supervised Learning

Supervised learning methods predict a functional relation $f: \mathbf{X} \rightarrow Y$ between a set of input samples $\boldsymbol{x} \in \mathbf{X} \subset \mathbb{R}^d$ and output variables $y \in Y$ so that it performs well on unseen samples. In fact, it infers an underlying input-output relation based on input-output of training samples (Fig. 2.1). This relation can be inferred using learning function f as follows:

$$f^* = \underset{f}{argmin} E_{(\boldsymbol{x},y)\sim D} L\left(f\left(\boldsymbol{x}\right), y\right), \tag{2.1}$$

where y and L denote the label of a sample $\boldsymbol{x}$, and a loss function, respectively. Moreover, $f(\boldsymbol{x})$ denotes the prediction function f on sample $\boldsymbol{x}$. The distribution D, typically is unknown, thus we cannot directly minimize f according to 2.1. However, based on *empirical risk minimization* (ERM) principle, the loss function L is obtained by a manually prepared set of $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ as follows:

$$f^* = \underset{f}{argmin} \sum_{i=1}^{N} L\left(f\left(\boldsymbol{x}_i\right), y_i\right). \tag{2.2}$$

Selection from a finite training set by ERM often leads to the problem of overfitting. *Structural risk minimization* (SRM) (2.3) attempts to prevent overfitting by incorporating a *regularization term* $\lambda \|f\|^2$ that penalizes extremely large values in f(2.3). The SRM principle selects

an f that performs well by balancing the model's complexity against its success at fitting the training data [32],

$$f^* = \underset{f}{argmin} \sum_{i=1}^{N} L\left(f\left(\boldsymbol{x}_i\right), y_i\right) + \lambda \|f\|^2 \tag{2.3}$$

The main objective of supervised methods is to approximate a model with the minimal generalization error, which depends on the joint distribution of input-output pairs and the model's complexity. Supervised learning can further be divided, based on the label types, into the following categories:

- Classification: includes decision trees [33], Naive Bayes, K-nearest neighbors [34] and support vector machines [35, 36];
- Regression: includes linear regression [37], kernel regression [38] and Gaussian process [39].

2.1.2 Unsupervised Learning

In unsupervised learning the output labels $y \in Y$ are not available for training. The goal of unsupervised learning methods is usually to discover some hidden underlying structures in the input data $\boldsymbol{x} \in \mathbf{X}$ (Fig. 2.2). In unsupervised learning, the training data is still assumed to be drawn *i.i.d.* from some underlying distribution D. It helps to understand the intrinsic properties of the data. Unsupervised learning problems can be divided on:

- Visualization or dimensionality reduction: includes, principal component analysis [40], locally linear embedding [38], and maximum variance unfolding [41];
- Outlier detection and density estimation mixture models [42];

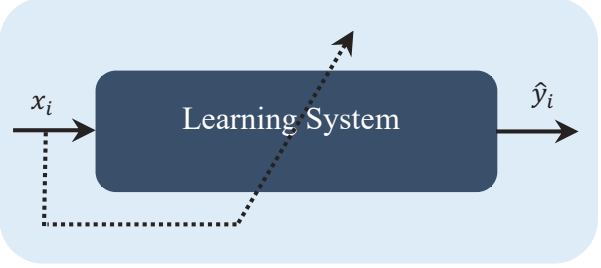

FIGURE 2.2: Unsupervised learning.

- Clustering: includes, K-means [43], spectral clustering [44], hidden Markov models [45], mixture models.

2.1.3 Semi-Supervised Learning

Semi-supervised learning includes a class of algorithms that is somewhat between supervised and unsupervised learning. The goal of semi-supervised learning is to learn from both labeled and unlabeled data, *i.e.*, the training data consists of labeled data instances $(\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_l) \in \mathbf{X}$, with corresponding labels $(y_1, y_2, ..., y_l) \in Y$, and unlabeled data instances $(\boldsymbol{x}_{l+1}, \boldsymbol{x}_{l+2}, ..., \boldsymbol{x}_{l+u}) \in \mathbf{X}$. Semi-supervised learning methods make use of a large amount of unlabeled data in conjunction with a small amount of labeled data [46]. This results in a considerable improvement in learning accuracy. Indeed, the semi-supervised learning methods, learn a classifier that maximize the separation across the underlying clusters of the data. Different frameworks for semi-supervised learning have been proposed. Semi-supervised methods, based on a set on assumptions, can be divided into:

- Generative models: include expectation maximization (EM) algorithms on mixture models [47];
- Manifold-based methods: include Manifold alignment and Sammon's mapping, project high dimensional data onto an intrinsic nonliner low dimensional manifolds [48, 49];
- Low density separation: includes Transductive SVM [50, 51], entropy or information regularization models [52];
- Graph-based models: include graph min cut [53], graph kernels [54], Gaussian Markov random fields and harmonic function method [55].
- The interactive mixture model based approaches: include self-training [56], co-training (multi-view training) [57] and Active learning [58].

2.1.3.1 Self-training

Self-training [56] or *self-teaching* is a general single-view bootstrapping algorithm. This group of algorithms uses its own predictions to teach itself. As shown in Figure 2.3, first, an initial model is created using a small amount of labeled data. Next, the created model is used

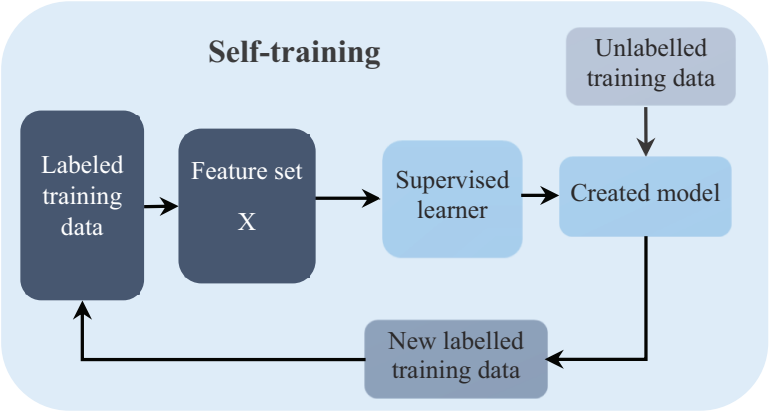

FIGURE 2.3: Self-training.

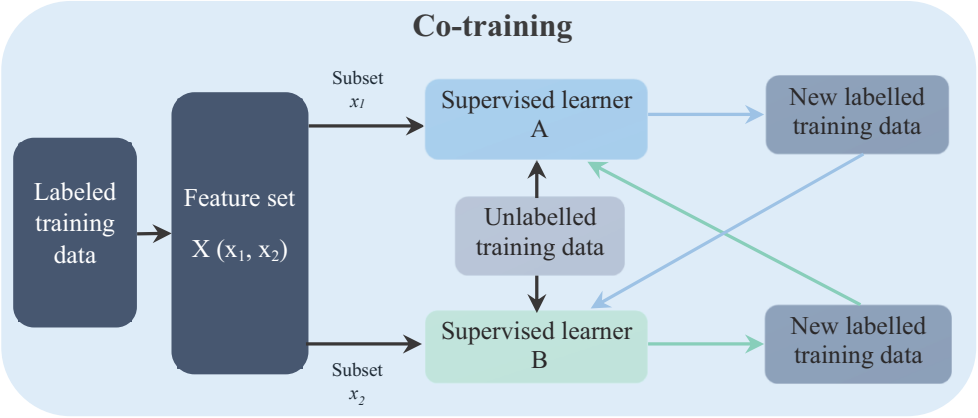

FIGURE 2.4: Co-training.

to label a pool of unlabeled data. Then, to update the created model, the most confident unlabeled data with their corresponding labels are combined with the original labeled data. The algorithm run for a fixed number of iterations or until a convergence criterion is satisfied.

2.1.3.2 Co-training

Co-training [57], a semi-supervised bootstrapping approach, attempts to increase the amount of annotated data by using a large amount of unlabeled data. Under co-training assumption, two different weak classifiers are trained on different *views* of data (*i.e.*, two different feature representations). In co-training, each classifier alternately assigns labels to the new unlabeled data. Then, the most confident labeled data resulted from each classifier are added to the initial set of labeled data. This set is used to iteratively estimate additional labeled training data using the other classifier (Fig. 2.4). To effectively leverage the unlabeled data, co-training

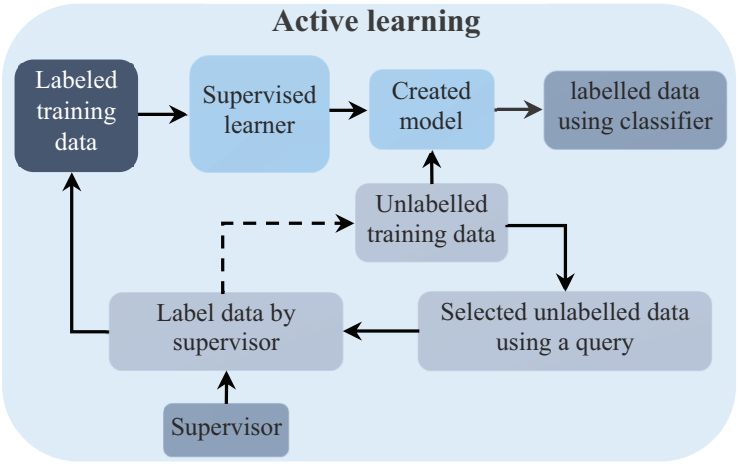

FIGURE 2.5: Active learning.

assumes consistency between two views. They are class conditionally independent and each view is sufficient to train a low-error classifier.

2.1.3.3 Active learning

Active learning [58] or iterative semi-supervised learning, is a learning framework to deal with the situations where unlabeled data is abundant but manually labeling is expensive. According to active learning, a learning algorithm can perform better with less training if it is allowed to choose the data from where it learns. In such scenario, learning algorithms can actively query human experts for labels (Fig. 2.5). In Active learning, the learner chooses the number of samples for training. Thus the required samples are much lower than the number required in normal supervised learning. Moreover, the input training points are generated following a user-defined distribution. Therefore, the probability distribution of the training and test data can be different.

2.1.4 Reinforcement Learning

Reinforcement learning is a framework to learn a policy function for computer agents that map the situations to actions via interaction with an environment. The goal is to select optimal actions in an environment in order to maximize a cumulative reward [59]. Even though the goal is to learn a policy function as a result of its experience, it is different from supervised learning. In reinforcement learning, the correct outputs cannot be obtained directly. Unlike unsupervised learning, an implicit supervision in the form of reward information is available

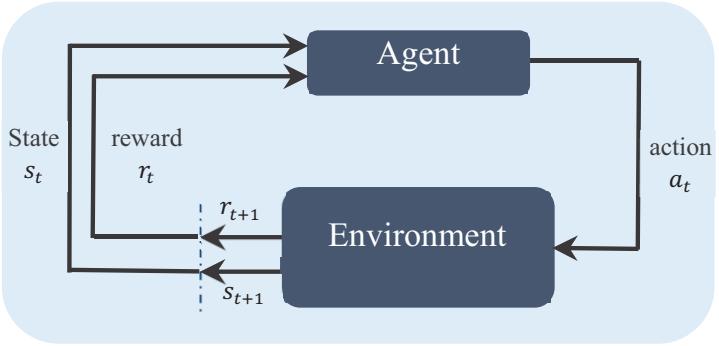

FIGURE 2.6: Reinforcement learning.

(to the policy learner) during interactions with the environment.

Usually, the environment is formulated as a Markov decision process (MDP), and the agent and environment interact in a sequence of discrete time steps $t = 0, 1, 2, 3, ...$ In each time step t, the agent receives a representation of the environment's state, $s_t \in S$, where S is the set of possible states. Depending on s_t, agent selects the action $a_t \in A(s_t)$, where $A(s_t)$ is the set of actions available in state s_t. In the next time step, as a result of the agent action, the agent receives a numerical reward, $r_{t+1} \in R$, and finds itself in a new state, s_{t+1}. Reinforcement learning methods learn an action-selection policy $\pi_t(s, a)$, which is a mapping from states $s_t = s$ to probability of selecting a possible action $a_t = a$, in order to maximize long-term cumulative rewards. Figure 2.6 depicts such agent-environment interaction. Reinforcement learning is particularly suitable to solve problems which include a long-term versus short-term reward trade-off, and have been applied successfully to various real world problems, including robot control, game theory and economics.

2.2 Learning Under Distribution Mismatch

As mentioned above, based on the main assumption of classical learning methods (*i.e.* independent and identical distribution (*i.i.d.*)), test data instances follow both the same probability distribution and the same feature space as the training instances [60]. However, this assumption is not satisfied in real-world applications. For example, in wifi localization the signal strength depends on many fast evolving parameters, yielding different distributions. In face recognition, training images are often obtained under some set of lighting or occlusion conditions that may change in the test phase (Fig. 2.7). In speech recognition, acoustic models

FIGURE 2.7: Face images with different lighting and occlusion conditions from one subject of the AR-face dataset [1].

trained on one speaker may be used by another. In natural language processing (NLP) including, parsers, documents classification and sentiment analysis, the carefully annotated training data are different from test data. The data evolve over time and change from one domain to another, thus, the training data may be outdated and not sufficiently representative for the distribution of the test data. The differences between the training (s) and test (ta) data may come from [60]:

- Differences between the feature spaces $\mathbf{X}_s \neq \mathbf{X}_{ta}$ or the marginal probability distributions $P_s(\boldsymbol{x}) \neq P_{ta}(\boldsymbol{x})$.
- Differences between the label spaces $Y_s \neq Y_{ta}$ or the predictive distributions (the conditional probability distributions) $P_s(y|\boldsymbol{x}) \neq P_{ta}(y|\boldsymbol{x})$.

The assumption of independent and identical distribution (*i.i.d.*) of training and test data is often violated in practice, and consequently the standard learning methods are no longer consistent leading a drop in the general performance. Indeed, when the distribution changes, the statistical models based on training data need to be rebuilt. The distribution changes between the training (source) and the test (target) domains may lead to a difference in joint probability distributions $P_{ta}(\boldsymbol{x}, y)$ and $P_s(\boldsymbol{x}, y)$ between the two domains, which may result in a failure of the *empirical risk minimization* (ERM) method. The ratio of joint distributional difference, $(P_{ta}(\boldsymbol{x}, y)/P_s(\boldsymbol{x}, y))$ indicates how different the two domains are at $(\boldsymbol{x}, y)$, where $\boldsymbol{x} \in \mathbf{X}$ denotes an instance from the observations set $\mathbf{X}$, $y \in Y$ denotes the class label y in class labels set Y, and $P_{ta}(\boldsymbol{x}, y)$ and $P_s(\boldsymbol{x}, y)$ denote the true underlying joint distributions in the target and source domains, respectively. Since estimating the probability distribution $P(\boldsymbol{x}, y)$ is a challenging task, especially in the target domain, where the labels are not available, the probability ratio cannot be computed directly. In order to simplify the analysis, the joint probability distributions $P_{ta}(\boldsymbol{x}, y)$ and $P_s(\boldsymbol{x}, y)$ can be decomposed. Bayes' rule is a basis to describe the relationship between different probability functions over the input variable $\boldsymbol{x}$ and

the target variable y:

$$P(\boldsymbol{x}, y) = P(y|\boldsymbol{x})P(\boldsymbol{x}) = P(\boldsymbol{x}|y)P(y), \tag{2.4}$$

Therefore, $P_{ta}(\boldsymbol{x}, y)/P_s(\boldsymbol{x}, y)$ can be decomposed as follow:

$$\frac{P_{ta}(\boldsymbol{x}, y)}{P_s(\boldsymbol{x}, y)} = \frac{P_{ta}(\boldsymbol{x})}{P_s(\boldsymbol{x})}\frac{P_{ta}(y|\boldsymbol{x})}{P_s(y|\boldsymbol{x})} = \frac{P_{ta}(y)}{P_s(y)}\frac{P_{ta}(\boldsymbol{x}|y)}{P_s(\boldsymbol{x}|y)}, \tag{2.5}$$

where $P_{ta}(\boldsymbol{x})$ and $P_s(\boldsymbol{x})$ denote the true marginal distributions of $\boldsymbol{x}$, $P_{ta}(y|\boldsymbol{x})$ and $P_s(y|\boldsymbol{x})$ denote the true conditional distributions, $P_{ta}(\boldsymbol{x}|y)$ and $P_s(\boldsymbol{x}|y)$ denote the class conditional distributions, and $P_{ta}(y)$ and $P_s(y)$ denote the class priors, in the target and source domains, respectively. Based on (2.5) differences between joint distributions of the source and the target domains can be categorized as follows:

1. *Labeling difference*: If the difference of $P_{ta}(\boldsymbol{x}, y)$ and $P_s(\boldsymbol{x}, y)$ comes from the difference between $P_{ta}(y|\boldsymbol{x})$ and $P_s(y|\boldsymbol{x})$ for a considerable number of $\boldsymbol{x} \in \mathbf{X}$, it means that the distribution of labels are different in the two domains at $\boldsymbol{x}$. Labeling difference [61] is also known as *model shift* [62] or *concept drift* [63], which refers to 1) the uncertainty about the future and the changes happening in target concept definition, 2) changes in statistical properties of streaming data over time [64], and 3) the training using the corrupted instances, which results in noisy labeled data [65]. The existing methods to deal with labeling difference can be categorized into *instance selection* [63], *instance weighting* [63] and *ensemble learning*[66].

2. *Priors difference*: (a) If the difference of $P_{ta}(\boldsymbol{x}, y)$ and $P_s(\boldsymbol{x}, y)$ comes from the difference between $P_{ta}(y)$ and $P_s(y)$, while $P_{ta}(\boldsymbol{x}|y) = P_s(\boldsymbol{x}|y)$, it means that, the class priors are shifted while the class conditional distributions remain the same. This type of distribution change is also referred as *target shift* [67] or *class imbalance* [68], in which some classes have much more instances than others. (b) However, if there is a difference between $P_{ta}(\boldsymbol{x}|y)$ and $P_s(\boldsymbol{x}|y)$, while $P_{ta}(y) = P_s(y)$ this category is known as *conditional shift* [67]. Re-sampling of training instances is the most effective approach to deal with *Priors difference*.

3. *Instance difference*: Another possible difference between the two domains may be due to the differences of $P_{ta}(\boldsymbol{x})$ and $P_s(\boldsymbol{x})$. It means that source and target domains may have different dense regions of $\boldsymbol{x}$. In this case, the marginal distributions $P(\boldsymbol{x})$ of the features are different in the source and the target domains while the posterior distributions of

the labels is the same in the two domains. This distributional difference is also referred to *covariate shift*. *Sample selection bias*, a particular case of covariate shift, refers to nonrandom selection of training samples, which happens due to a variety of practical limitations such as the cost of data labeling or acquisition [69, 70].

In real world applications, there are cases in which several types of distribution mismatch exists, simultaneously. However, the aforementioned types of differences, based on Bayesian probability functions, are simplifications of the realistic problems. The machine learning community considers different aspects of the problem depending on the type of:

(a) the distribution mismatch in terms of domain and tasks;

(b) the learning process of the tasks (*i.e.* symmetric or asymmetric task).

In [60], some solutions based on knowledge transfer between tasks/domains are proposed.

2.2.1 Transfer Learning

Transfer learning, a general sub-field of machine learning, aims to identify, extract and transfer the relevant useful knowledge from one (or more) source domains/task for learning in a target domain/task (Fig. 2.8). This allows the domains, tasks, and distributions of training and test data to be different. There are three main issues on transfer learning [60]:

- What to transfer: it identifies and extracts the relevant and common knowledge from source domains that can be beneficial for learning in the target domain.
- How to transfer: it asks how to develop learning algorithms for transferring the extracted knowledge to the target domain.
- When to transfer: it identifies the situation for which transfer learning can be applied. A negative transfer will happen, where the domains are not related to each other. The negative transfer decreases the performance of the target learner compared to the normal learning algorithms.

To deal with the aforementioned issues, several transfer-learning based methods with different names are proposed. Based on the relationships between tasks, as well as the availability of

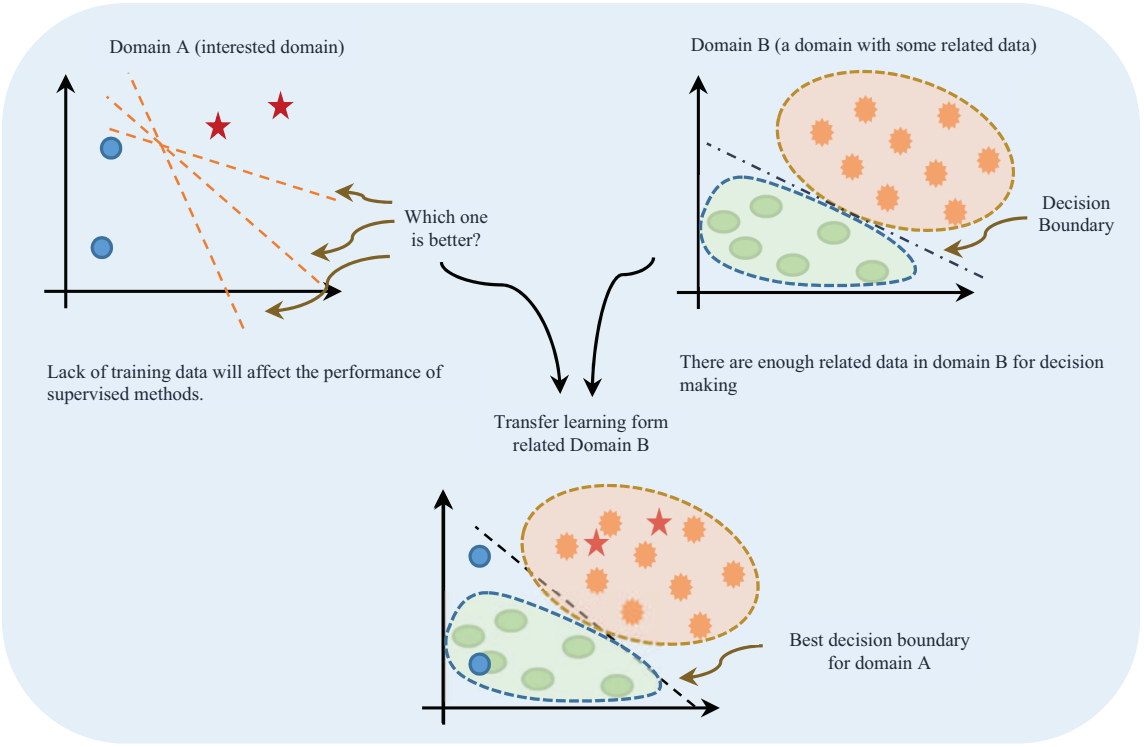

FIGURE 2.8: Knowledge transfer from domain B to domain A.

labeled training samples in source and target domains, the methods can be divided into four groups:

1. *Inductive transfer learning:* A setting of transfer learning, where a few labeled instances of the target domain and much more labeled instances from source domain exist. The source domain could be different or the same as the target domain, however, the target task is different but related to the source task. Data labeling in many real-world problems is too expensive. Therefore, the predictive model of the target domain can be induced using both labeled samples of the target and source domains. In this case, knowledge transfer from source domain D_s and source task T_s results in a better target model $f_{ta}(.)$, higher performance, and in particular prevents overfitting in target domain [71].

2. *Transductive transfer learning:* A case where, many labeled instances in source domain and no labeled instances in the target domain are available [72, 73]. The source and target tasks are the same while the target domain is different but related with the source domain. It aims to leverage not only the source knowledge but also some extra information about the relation between the source and the target. The feature space of the

domains can be different or the same. Using transductive transfer learning, identification of object categories, never seen before, can be performed through transfer of their description [74, 75].

3. *Self-taught learning:* A few labeled instances of the target domain and no labeled instances from source domains are available. The source and target domains are the same, however, the labels as well as the tasks of source and target domains may be different but related. Self-taught learning aims to extract some useful information from the source, even if some label sets are unknown. Source knowledge can be formalized as a high level representation through unsupervised feature construction [76].

4. *Unsupervised transfer learning:* A case when no labeled instances in the target and source domains are observed in the training phase. The source and target domains may be the same, however, the target task is different but related to the source task. The methods proposed in this setting include self-taught clustering, transfer dimensionality reduction and density estimation [77, 78].

2.2.1.1 Heterogeneous vs. Homogeneous Transfer

Depending on the similarity or differences between the features that, represented source and target domains, transfer learning methods can be divided into Homogeneous or Heterogeneous, respectively. The methods cover both the conditions of fixed and changing label sets across the tasks. Several heterogeneous transfer methods have been presented in cross-language classification [78, 79] and text-to-images classification [80, 81] (Fig. 2.9).

2.2.2 Multitask Learning

Multitask learning (MTL), a particular case of inductive transfer learning, aims to simultaneously learn from multiple related tasks using some knowledge transfer between them, which can significantly improve the learning performance (Fig. 2.10). In MTL setting, there are no source and target domains. In order to obtain more robust solutions, the methods use the relationship between the tasks. In this framework, the feature space of the tasks are the same, however, the tasks are different but related. Each task is considered a bias for the others and has a positive generalization effect. Depending on the structures, there are two types of MTL methods:

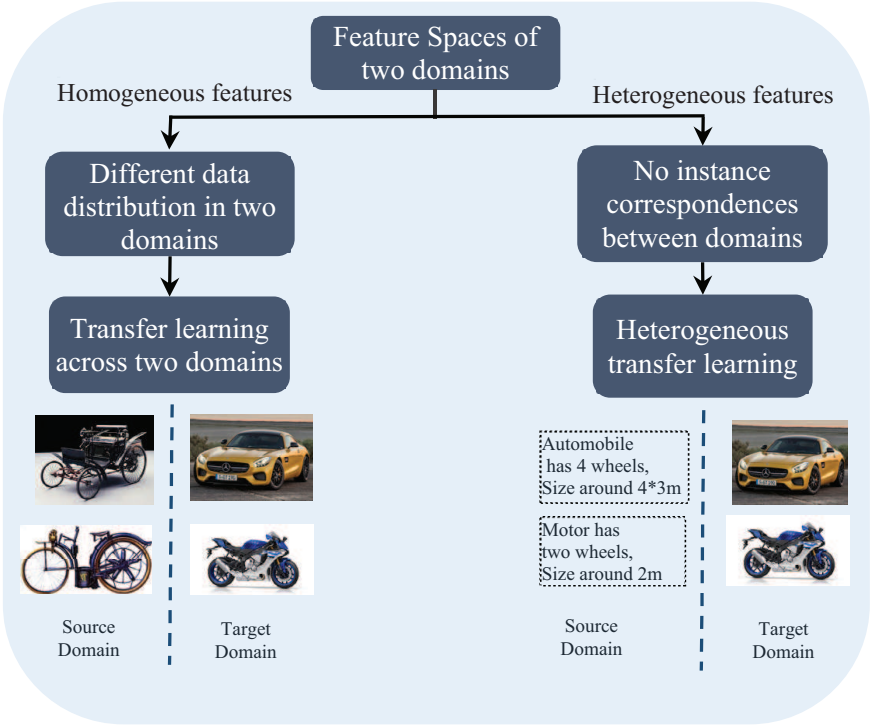

FIGURE 2.9: An intuitive illustration of Homogeneous vs. Heterogeneous transfer.

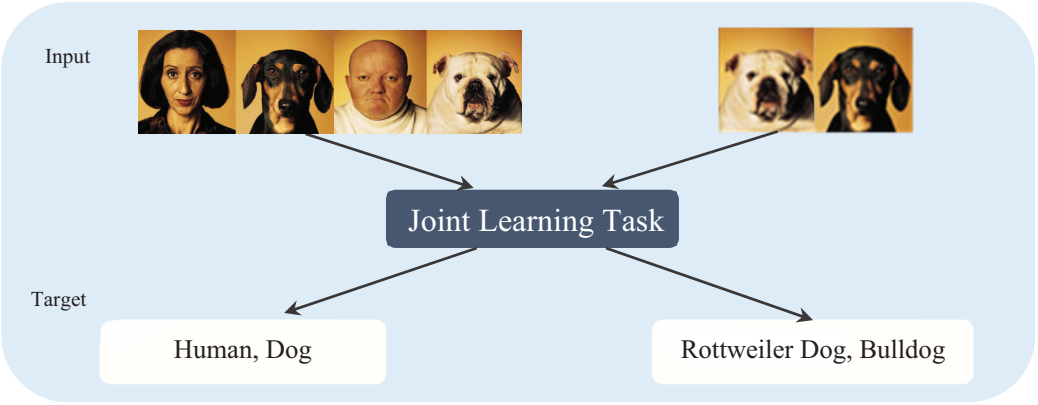

FIGURE 2.10: Multitask learning.

- *Sequential MTL:* The tasks are learned iteratively in a sequence, one after another. Indeed, in each iteration, a new task is added to the system where the previous knowledge obtained from the other tasks influences the learning of this task. However, adding a new task will affect the predictors of the previous tasks. By applying the transferred knowledge in MTL, the method needs less training samples to learn a new task.

- *Parallel MTL:* The available data are learned by the tasks at the same time. The knowledge transfer between the tasks improves the performance of each task predictor, while less training samples are needed comparing to a single task learning strategy.

The effectiveness of MTL methods including, discriminative learning [82], latent features [83] and Bayesian [84] has been shown in a wide range of areas such as natural language processing [85] and machine vision [86–88].

2.2.3 Domain Adaptation

Domain adaptation (DA) aims to build a model using the labeled training data in a source domain that will perform well on the test data of the target domain. The problem of domain adaptation rises when there is a large amount of labeled data in a source domain, but the target domain, does not have enough labeled data to create a model. In domain adaptation setting, the source and the target domains are different, however, the source and target tasks are the same.

In order to alleviate the distribution mismatch of the domains, it effectively integrates the labeled data from source domain with labeled or unlabeled data from the target domain [89]. If we expect to build a model from the source domain for the target domain, neither instance difference nor labeling difference can be large. In another word, the distributions P_s and P_{ta} of instance $\boldsymbol{x}$ in two domains should have a reasonable overlapping, and there should exist a large fraction of $\boldsymbol{x} \in \mathbf{X}$ where $P_{ta}(y|\boldsymbol{x} = x)$ is very close to $P_s(y|\boldsymbol{x} = x)$ [61]. However, if the samples of source and target domains are disjunct and their distributions have nothing in common, domain adaptation between two domains result in a negative transfer.

Several domain adaptation methods have been proposed in a wide variety of applications including, cross-language document classification [90], sentiment classification [91], Wi-Fi localization [92] and email spam detection [93].

Even though there is some similarity between domain adaptation and semi-supervised learning, indeed they are different. Domain adaptation assumes that the labeled and unlabeled data come from different but related domains, while semi-supervised learning methods employ both labeled and unlabeled data from the same domain. More details on domain adaptation are presented in Chapter 4.

2.2.4 Class Imbalance

Class imbalance, or *prior difference* can be a reason of joint distributional difference of source and target domains. Indeed, the distribution difference between $P_{ta}(\boldsymbol{x}, y)$ and $P_s(\boldsymbol{x}, y)$ of two domains may be due to $P_{ta}(y) \neq P_s(y)$, however, it assumes the same class labels as well as

the same conditional distributions $P_{ta}(\boldsymbol{x}|y) = P_s(\boldsymbol{x}|y)$. Under this assumption, the ratio of joint probability distribution in (2.5), can be rewritten as follows:

$$\frac{P_{ta}(\boldsymbol{x}, y)}{P_s(\boldsymbol{x}, y)} = \frac{P_{ta}(y)}{P_s(y)} \frac{P_{ta}(\boldsymbol{x}|y)}{P_s(\boldsymbol{x}|y)} = \frac{P_{ta}(y)}{P_s(y)}. \tag{2.6}$$

In order to reduce the distribution difference between $P_{ta}(y)$ and $P_s(y)$, resampling of training instances from source domain can be applied. In resampling methods, by applying weights to each sample, under-represented classes are over sampled and over-represented classes are under-sampled [94, 95].

Japkowicz and Stephen [68] studied the effect of class imbalance problem on various classifiers, including decision trees (C5.0), neural networks (multi-layer perceptron), and hard margin SVM. The number of the training examples, the degree of the class imbalance and the complexity of the target function will affect the class imbalance problem.

2.2.5 Covariate Shift

Covariate shift or *instance difference*, which is closely related to domain adaptation, refers to a situation where the tasks and conditional distributions are exactly the same, namely $Y_s = Y_{ta}$ and $P_{ta}(y|\boldsymbol{x}) = P_s(y|\boldsymbol{x})$, however, the marginal distributions are different in training and test phase $P_{ta}(\boldsymbol{x}) \neq P_s(\boldsymbol{x})$ [17]. Covariate shift happens in many real-world applications such as robot control, bioinformatics, spam filtering [96], brain-computer interfacing [97] and speaker identification [98].

In classification/regression problems, we are interested only in the conditional distribution $P_s(y|\boldsymbol{x})$, thus it may appear that the covariate shift is not a problem [61]. However, despite the similarity between $P_{ta}(y|\boldsymbol{x})$ and $P_s(y|\boldsymbol{x})$, the created model using source domain does not perform well on the target domain. Shimodaira [17], proved that the covariate shift becomes a problem when misspecified models are used. In misspecified models family, the optimal model will be selected based on $P(\boldsymbol{x})$. Since the marginal distributions in source and target domains are not equal, $P_{ta}(\boldsymbol{x}) \neq P_s(\boldsymbol{x})$, therefore the selected optimal model for source and target models will be different. In fact, an optimal model performs better in dense regions of $\boldsymbol{x}$ than in a sparse region (*i.e.* the classification error of a dense region dominates the average classification error). Due to the differences of dense regions in source and target domains, the optimal model of source domain will no longer be optimal for target domain [61]. Since the covariate-shift based models have been shown to be useful, learning under covariate shift is

currently receiving great attention by the machine learning community [99].
Covariate shift adaptation was first studied in the pioneer work Shimodaira by [17]. The log-likelihood of source domain samples are reweighted in order to minimize the loss of the classifier on the target domain. Several other methods were proposed to correct the distributional mismatch of the source and target domains [70, 100, 101]. More details are presented in chapter 4.

2.2.6 Sample Selection Bias

Sample selection bias, which refers to a nonrandom selection of training samples, is a special case of covariate shift. In many real world applications such as astronomy, econometrics and medical diagnosis we do not have full control on the data gathering process. It happens due to a variety of practical limitations such as the cost of data labeling or acquisition. Even though training samples are drawn according to the test distribution, but in practice, they do not constitute the completely random samples of the underlying distribution. The model created based on these samples is biased and this problem is referred to as sample selection bias problem. In the context of machine learning each training sample $(\boldsymbol{x}, y)$ is drawn independently from a distribution D. Heckman in [69] uses a binary indicator s for each sample, which takes the value $s = 1$ whenever a labeled sample $(\boldsymbol{x}, y)$ is belonging to the training set. There are four types of sample selection bias according to dependency of s from $\boldsymbol{x}, y$ in each sample:

1. The selected sample is not biased: s is completely independent both from features vector $\boldsymbol{x}$ and class label y, i.e., $P(s|\boldsymbol{x}, y) = P(s)$. In this case, the samples $(\boldsymbol{x}, y)$ which have $s = 1$, constitute the random samples from D.

2. The selected sample is biased due to the feature vector $\boldsymbol{x}$. s is dependent on the feature vector $\boldsymbol{x}$ and independent of class label y; (i.e $P(s|\boldsymbol{x}, y) = P(s|\boldsymbol{x})$).

3. The selected sample is biased due to the class label y. s is independent of the feature vector $\boldsymbol{x}$ and dependent on class label y, which corresponds to a change in the prior probabilities of the labels; (i.e $P(s|\boldsymbol{x}, y) = P(s|\boldsymbol{x})$).

4. The selected sample is biased due to the feature vector $\boldsymbol{x}$ and the class label y. s is dependent to $\boldsymbol{x}$ and y. It's not possible to learn a mapping from features to labels using the selected feature vector [70].

To correct the created models under sample selection bias, Hackman [69] uses an estimated probability that an observation is selected into the sample. However, this method is only applicable to linear regression models, which are used in econometrics. Since pioneer work of Heckman [69], which was developed to correct samples selection bias, several methods have been proposed in machine learning. For instance, a cluster based estimation technique and kernel mean matching (KMM) [101] are two commonly used sample bias correction techniques. The basic idea of KMM is to reweight instances, such that the mean of probability distributions of instances become similar for source and target $P_{ta}(\boldsymbol{x}) = P_s(\boldsymbol{x})$. Generally, most of the methods consist of three steps:

- Try to recover sampling distributions;
- Try to correct the distribution of samples through resampling;
- Try to apply the learner to the corrected samples.

2.2.7 Concept Drift

Concept drift or *labeling difference* refers to the changes in statistical properties of streaming data over time. The core assumption of the concept drift problem is uncertainty about the future, which is due to the change of some hidden context. These changes make the model built based on training data inconsistent with the test data. The most popular example of concept drift is to detect and filter out the spam emails. The distinction between unwanted and legitimate e-mails is user-specific and evolves with time. According to the definition in [64], concept drift may occur due to:

1. The change happens in target concept definition, which is represented in class conditional probability distributions.
2. The change happens in data distribution, which is represented by posterior probabilities.

A difficult problem in handling concept drift is distinguishing between true concept drift and noise. There are three main approaches to detect and to handle a concept drift in time-evolving data:

- *Instance selection*: Selects the training instances relevant to the current concept. It consists of a temporal window with fixed or adaptive size [63] that moves over the data stream and uses the learned concepts to make a model for immediate instances [102, 103].
- *Instance weighting*: The instances can be weighted based on relatedness and closeness to the current concept. However, this group of approaches is not robust to overfitting problem [63].
- *Ensemble learning*: It dynamically assigns weights to the target instances by making use of additional classifiers, where these classifiers can select the best base model for prediction of a specific target instance [104]. This approach generally combine diverse models to find the best model in the target domain [66, 105].

One assumption in many approaches [66, 102] is that the distribution of the current data chunk is closest to the most recent data chunks. According to this assumption, models for detecting the concepts in the current data should be built based on the recent data. The ensemble approach should also select and weight the base classifiers from the recent data chunks.

Chapter 3

Sleep and Sleep Staging

This chapter introduces the basic terminology for the understanding of sleep staging methods. First, a definition of sleep and sleep staging with the underlying principles of these concepts are presented. Second, the sleep-related disorders and the effects of sleep disorders and medications on sleep patterns are summarized in the next subsections. Then, an overview of the existing research works on automatic sleep staging in the literature is presented.

3.1 Sleep

Sleep is an active and regulated process with an essential restorative function for physical and mental health [5]. Sleep occupies approximately one-third of our lifetime, and in addition to nutrition, fitness and emotional states, it plays an important role in human wellness. Sleep seems to be a behavior complying with the regular need of the human body to rest. During sleep, human brain goes through several psychophysiological states that are relatively stable. On the other hand, due to inactiveness of many nervous centers during sleep brain becomes a less complex system and is suitable for mathematical modeling [106]. The prevalence of sleep disorders in the general population is considerably high. In fact, millions of people over the world suffer from sleep disturbances [107]. These disorders may have an impact on the psychological and physiological well-being of a person. Therefore, quality of sleep and sleep disorders have an important effect on the health and quality of life.

The study of individual behaviors during sleep, such as monitoring, scoring and detecting

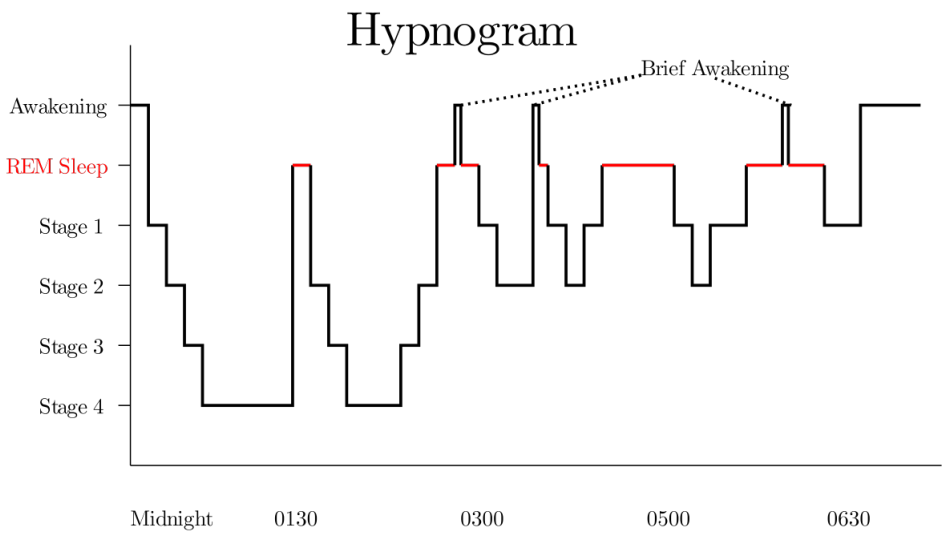

FIGURE 3.1: Sleep cycles through the night, with deep sleep early on and more REM toward morning.

abnormal changes of sleep pattern through all-night recordings have consistently been an important research topic.

Polysomnography is a common and basic technique to analyze and diagnose of the human sleep and sleep disorders. During all-night polysomnographic test in a sleep laboratory, several physiological parameters related to the sleep and vigilance states are simultaneously monitored and recorded. The recordings typically include electrophysiological signals (electrocardiographic activity, brain-wave patterns, eye movements and activation signal of muscles), pneumological signals (airflow, blood oxygen level, and movement of respiratory muscles), and other contextual information (body position, lights, snore recording)[13]. These signals have been collected from human individuals using non-invasive surface electrodes.

3.2 Sleep Staging

Sleep staging, also known as sleep scoring, is an essential part of the diagnostic process in assessment of sleep physiology and sleep disorders such as sleep Apnea syndrome (SAS) or hypersomnia [6]. The main objectives of sleep staging are to identify the events during sleep and to study the sleep architecture. The sleep architecture is referred to the cyclical pattern of sleep as it shifts between the different sleep stages, and it can produce a picture of what sleep looks like over the course of a night (see an example in Fig. 3.1).

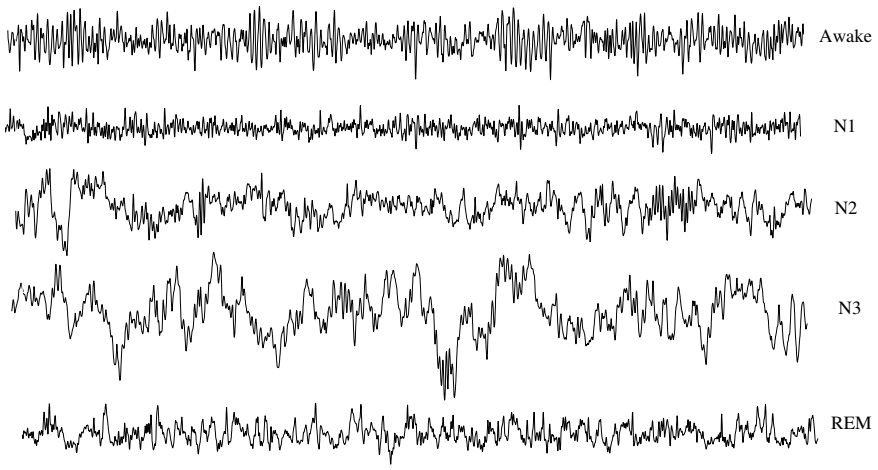

FIGURE 3.2: EEG pattern of different sleep stages

The relation of sleep with the variations of EEG patterns was first studied by Hans Berger [108]. Following this study, Loomis et al.[109] observed that the brain electrical activity during a night sleep changes. Thus, based on the EEG analysis, different sleep stages were determined. In 1953, sleep was classified into rapid eye movements (REMs) and non-rapid eye movement (NREM) stages [110]. Later, the sleep analysis was extended by monitoring physiological signal EMG and EOG.

The transition between the wake and the different sleep stages are fuzzy and there is no crisp border between the stages [111]. Classical approaches to sleep staging involve sleep experts (or sleep technicians) utilizing a manual technique of scoring. Manual visual sleep scoring by highly trained human experts has the following drawbacks:

- Visual interpretation of PSG records uses fixed epoch duration 30 s (more rarely 20 s) (Fig. 3.2). It is a very time consuming task and normally may require hours to score the PSG recording of a whole night.
- It is also a somewhat subjective procedure in which the concordance between the results of visual scoring obtained by experts can vary greatly. In fact, visually analysis of experts can have an inter-rater agreement lower than 80% [112].
- Moreover, many sleep staging results cannot be reproduced since the scorers might not even agree with each other in two trials.

Accordingly, an efficient automatic sleep staging may save time and provide objective assessment of sleep, independent of subjective interpretation of experts.

3.3 Rules for Sleep Staging

The Rechtschaffen and Kales standard (R&K) rules are the basis of a consensus scoring for adults [113]. According to the R&K, there are six possible sleep stages: Wake (for periods when the subject is awake), Stage 1 (light sleep), Stage 2 (the most prevalent stage in healthy subjects), Stage 3, Stage 4 (both stage 3 and 4 are known as deep or slow-wave sleep stages), and rapid eye movement (abbreviated REM, this stage is also known as dream sleep or paralysis sleep). Each stage has its own set of identifiable traits (or features) that can be observed in an PSG recording.

The American academy of sleep medicine (AASM) determined new criteria in the scoring of sleep based on the R&K rules. In adults, sleep-wake cycle is categorized in awake, non-rapid eye movement (NREM) and rapid eye movement (REM) sleep stages. NREM sleep is further divided into three stages: N1, N2 and N3 [114], the last of which is also called delta sleep or slow wave sleep (SWS). The main difference of the ASSM with R&K for sleep staging is in the total number of sleep stages. There are only five sleep stages as opposed to six of R&K. To discriminate the stages, the AASM rules define some characteristics according to the amplitude, frequency and shape of the PSG signals (Fig. 3.2) as described in Appendix A [114].

3.4 Sleep Related Disorders

Sleep disorders affected the normal sleeping pattern of a patient, and sometimes are serious enough to interfere with normal physical, mental and emotional functioning. In-adequate or non-restorative sleep can markedly impair a patient's quality of life [115]. Sleep disorders may need the examination of the patient's sleeping patterns over extended periods of time for a correct diagnosis and treatment. In order to standardize the sleep disorders and create a systematic approach for their diagnose, the International Classification of Sleep Disorders (ICSD) was created in 1990, which in 2005 was updated and renamed as ICSD-2 [116]. The sleep disorders are organized into eight distinct categories:

- Insomnias.
- Sleep related breathing disorders.

- Hypersomnias of central origin not due to a circadian rhythm sleep disorder, sleep related breathing disorder or other cause of disturbed nocturnal sleep.
- Circadian Rhythm Sleep Disorders.
- Parasomnias.
- Sleep Related Movement Disorders.
- Isolated Symptoms, Apparent Normal Variants, and Unresolved Issues.
- Other Sleep Disorders, such as alcohol abuse-related or psychiatric disorders.

In a healthy adult, sleeping on a normal schedule, sleeps follows a certain architecture. The sleep stages during a night sleep, proceeds in cycles of NREM and REM, each cycle normally being $N1 \rightarrow N2 \rightarrow N3 \rightarrow N2 \rightarrow REM$. The cycles typically happen 4 to 6 times during whole night sleep [117] and the first episode of REM sleep approximately happens 80 to 100 minutes after sleep onset. Across the night the length of the episodes of deep sleep and REM sleep vary. In the first cycles of sleep, N3 episodes are longer than REM sleep episodes. As the night progresses, N3 episodes start to be smaller or even absent and REM sleep episodes become longer. In young adults, N1 sleep constitutes about 5-10% of the night. The largest amount of sleep time, 50-60%, is spent in stage N2. Stage N3 constitutes about 10-20% of the total sleeping time while REM sleep 20-25% [19].

3.4.1 Effect of Sleep Related Disorders on Sleep Patterns

Sleep Apnea syndrome (SAS) is the most frequent sleep disorder seen in sleep medicine centers. The syndrome is characterized by repetitive episodes of upper airway obstruction that occur during sleep and are usually associated with a reduction in blood oxygen saturation. These nocturnal respiratory disturbances may sometimes occur more than 300 times a night and result in brief arousals in sleep, which promotes sleep fragmentation that typically disturbs sleep architecture with reduction or even complete deprivation of REM sleep and N3 sleep. An increase of arousals of different length together with an increase in sleep stage changes is a feature of the syndrome. This fragmentation of sleep, inhibiting cortical synchronization, would be responsible for the lower amount of slow wave sequences of the deep sleep [118]. On the other hand, transient experimental hypoxia induced abnormal posterior resting state delta and alpha rhythms in healthy volunteers, and EEG slowing during awake with an increase in

relative theta and delta power in occipital, temporal and parietal areas was observed in sleep apnea subjects [119], which can be correlated with sleepiness in these patients.

Obstructive Sleep Apnea (OSA) is the common type of sleep apnea. OSA occurs when the upper airway occludes (either partially or fully), resulting in oxygen desaturation and arousals from sleep. The primary causes of upper airway obstruction are lack of muscle tone during sleep, excess tissue in the upper airway, and anatomic abnormalities in the upper airway and jaw [120]. This disease has many potential consequences including excessive daytime sleepiness, neurocognitive deterioration, endocrinologic and metabolic effects, and decreased sleep quality [121].

Also, moderate obstructive sleep Apnea (OSA) patients have a lower percentage of slow spindles with deceleration compared to mild OSA or normal groups in frontal and parietal regions, which may represent a disruption of thalamo-cortical loops generating spindle oscillations [122]. All together, these findings can contribute to significant differences in agreement among multiple raters of sleep studies in sleep apnea patients [123]. Affective disorders (depression/anxiety disorder) can induce sleep-EEG changes. In depression, an increase in sleep latency and an increase in sleep fragmentation by arousals or intermittent awakenings, and early-morning awakening can be seen. Frequently a shortened REM latency including sleep onset REM periods (REM latency < 20 min), prolonged first REM periods with an increase of REM density is present in all age groups [124]; NREM-sleep changes include a decreased SWS, EEG delta power throughout the night and increase of stage N2 sleep together with a shift of EEG-delta power from the first to the second sleep cycle in younger patients [125].

In generalized anxiety disorder (GAD) patients no clear differences with control subjects in SWS and REM sleep are seen, being the differences confined to insomnia-like symptoms [126]. Nevertheless, some studies have shown increased REM latency [127], or increased mean REM latency over consecutive nights [128], in GAD patients compared to depressed patients, which could be useful to distinguish both disorders. In primary insomnia (psychophysiological insomnia) the hyperarousal model suggests that a deficit of reducing arousal during sleep may be responsible for non-restorative sleep. It has been shown that in these patients there is elevated spectral power values in the EEG beta (cortical arousal) and sigma (spindle) frequency band during N2 sleep stage with no differences in other frequency bands. This increase in cortical arousal and in an index of sleep protective mechanisms (spindles) may provide further evidence for the concept that a simultaneous activation of wake-promoting and sleep-protecting neural activity patterns contributes to the experience of non-restorative sleep in primary insomnia [129].

In dementia, sleep is characteristically grossly disrupted with lower sleep efficiency, higher percentage of N1 and increase of arousals and awakenings. A decrease of SWS can be expected but the hallmark of the sleep EEG in these patients is a slowing EEG activity with spindles reduction. Therefore, the scoring of sleep stages may be challenging [126].

3.5 Effect of Medications on Sleep Stage Patterns

The medication can also affect the EEG sleep patterns. Chronic use of benzodiazepines has shown to induce an increase of sleep stage N2, decrease of N3 (lower delta activity and theta activity), and an increase of arousals [130]; increase of spindle activity that can intrude in REM sleep is also described [131].

REM sleep reduction or even complete suppression was reported in humans after an administration of tricyclic or tetracyclic antidepressants [132], monoaminoxidase inhibitors [133, 134], SSRIs [135], selective Noradrenaline Reuptake Inhibitors (NRI) [136], SSRIs and selective Noradrenaline Reuptake inhibitors (SNRI) [137]. A few exception should be the noradrenergic and specific serotoninergic antidepressant mirtazapine [138] which also increase total sleep time and sleep efficiency after four weeks of administration.

Most tricyclic antidepressants increase SWS [132], but there is evidence that selective serotonin Reuptake inhibitors (SSRI) impairs sleep continuity by increasing of intermittent wakefulness [126]. On the other side, an increase of sleep stage N3 was found during treatment of depressed patients with the SNRI duloxetine [137] and a decrease in REM sleep and increase in REM sleep latency were observed with the SNRI venlafaxine [139]. An increase in the total sleep time and sleep efficiency, and a decrease in the time spent awake, were verified in patients with depression, under mirtazapine medication. These changes persisted after four weeks [138].

Trazodone, a triazolopyridine antidepressant weak, is a specific inhibitor of serotonin (5-HT) reuptake. The use of this medication showed increases in sleep efficiency, total sleep time, total sleep period, N3 and REM duration, as well as decreases in wakefulness during the total sleep period, early morning awakening, and N2 [140].

Besides alterations in sleep EEG induced by diseases or medications in healthy people, there is specific individual sleep patterns that can be seen as a “fingerprint” that allows a correct discrimination between individuals with a probability of 92% [141]. This EEG fingerprint is genetically determined as shown by studies on monozygotic and dizygotic twins, particularly in the range of alpha and sigma bands [141].

The above observations can explain why human-experts analysis of sleep EEG can be, so far, superior to computer analysis. A better decision about what the most likely sleep stage is [131] can be done: by a better adaptation to the individual characteristics of the electrophysiologic signals, by recognizing specific sleep-related characteristics constructing their own, and by recognizing subjective and patient specific PSG pattern.

On the other hand, this also explains why as much as 25% of overall disagreement can be detected between two human-expert's sleep scorings of the same recording [142]. The disagreement is particularly seen in N1 where the AASM definition includes attenuation or slowing of the alpha rhythm and the presence of slow eye movements, $4-7$ Hz EEG and vertex sharp waves. Many subjects do not generate some (or even any) of these waves as is stated in the manual [131].

3.6 Automatic Sleep Stage Classification

Automated methods for sleep staging aim to improve the accuracy and to provide a reliable and reproducible backup for the manual scoring done by human experts. These improvements would ultimately result in a cost reduction for sleep disorder diagnosis, treatment and research. Several studies have reported the development of computerized methods for sleep pattern analysis. These methods can be considered based on the following aspects: 1) Objective of the methods; 2) Preprocessing and artifact removal; 3) Feature extraction and selection; and 4) Classification, clustering and decision making.

Objective of the methods

In addition to complete sleep staging, to classifiy the epochs as Wake, N1, N2, N3, and REM, some researches only address the classification of sleep versus wake or just a specific part of sleep. For instance, in [143] due to the difficulty in separation, N1 with REM are considered as a one stage. Similarly, since deep sleep or slow wave activity is characterized by delta band activity, thus, Fell et al. [144] considered S3 and S4 as one stage.

On the other hand, due to the importance and application of waveform recognition, recent researches just focused on recognition of the specific waveforms such as alpha, betha, sigma, spindels and K-complexes (see Table 3.1). Some studies attempted to identify the optimal EEG channels [145, 146], and to employ recognition capabilities of ANN and SVM [147] for sleep spindle detection. Like sleep spindles, K-complexes are difficult to identify. However, Erdamar et al. [148] developed a method for the automatic detection of K-complex from EEG

TABLE 3.1: Summary of EEG, EOG and EMG patterns for different sleep stages.

Stage	EEG					EOG	EMG
	Delta ($< 4Hz$)	Theta (4-7Hz)	Alpha (8-13Hz)	Beta ($> 13Hz$)	Other EEG patterns		
Awake			x	x		0.5-2 Hz	Variable amplitude but usually higher than during sleep stages
N1		x	x		Vertex waves	Slow eye movement	Lower amplitude than in stage awake
N2		x			K-complexes; Sleep spindles	Usually no eye movement, but slow eye movements may persist	Lower amplitude than in stage awake and may be as low as in stage REM
N3	x				Sleep spindles may persist	Eye movements are not typically seen	Lower amplitude than in stage N2 and sometimes as lower as in stage REM
REM		x	x		Sawtooth waves	Rapid eye movement	Low chin EMG tone; usually the lowest level of entire recording

recordings based on the morphology of the K-complex. Some features based on amplitude and duration properties of K-complex waveform were considered. In another study [149] an efficient neuro-fuzzy K-complex detector was developed yielding an accuracy of approximately 96%.

In addition, to overcome the limitations and drawbacks of epoch-based sleep staging, new definitions for categorization of the sleep structure were proposed. These methods were designed based different ideas including: 1) clustering techniques such as self-organizing feature map (SOFM) [150]; or 2) the fuzzy inferences for a more continuous evolution of the sleep patterns [151]. Indeed, to avoid the binary decisions, they have provided soft transitions and enabled concurrent characterization of the different states.

The processed signals

Similar to the manual scoring, the automated methods perform the sleep staging using the analysis of PSG records. A significant portion of the methods used electroencephalographic (EEG) records, often in combination with electrooculographic (EOG) and electromyographic (EMG) records collected from human individuals using noninvasive surface electrodes. Since the channels defined in R&K are often criticized for not containing sufficient information, especially in particular cases, such as sleep spindle detection [145] some researchers have attempted to include more channels [143]. Signals other than EEG, EOG, and EMG have been considered for sleep staging as well. Some studies include the electrocardiogram (ECG), respiratory signals, oxygen saturation, blood pressure, body temperature, body position and movement. More inputs translate to more information that can be extracted, but it also means complexity

TABLE 3.2: Recent important works of automatic sleep stage classification.

ASSC approaches	Sleep stages	Nature of feature	Matching process	Subjects/ Channels	Quality evaluation
Zoubek et al.[5]	W, S1, S2, SWS, REM	30 s epoch; 10 features, EEG: RP delta, theta, alpha, sigma, beta (FT coefficients), 75th percentile; EMG: entropy; EOG: entropy, kurtosis number and SD.	Neural Network BP MLP	47 recordings, EEG, EMG	71% (EEG only), 80% (EEG, EOG and EMG): W: 84.57%, S1:64.56%, S2:85.55%, SWS:92.90%, REM: 72.81%.
Jo et al.[152]	W, shallow sleep (SS), deep sleep (DS), REM	30 s epoch; Fast Fourier transform (FFT) with Hamming window; power spectra; Relative powers(RP)	Fuzzy classifier and a genetic algorithm (GA)	4 recording, single EEG(C3-A2)	84.60%
Tang et al.[22]	W, S1, S2, S3, S4, REM	30 s epoch; HHT, Wavelet Transform, Autoregressive model	SVM	6 recordings, EEG(C3-A2), EMG and EOG	Wavelet: 77.9% HHT: 77.6%
Fraiwan et al.[12]	W, N1, N2, N3, REM	30 s epoch; Choi–Williams distribution (CWD), Continuous wavelet transform (CWT), and HHT and Renyi's entropy measures	Random forest classifier	16 recording, Single EEG (C3-A1)	Accuracy 83%; and kappa coefficient of 0.76.
Gunes et al.[153]	W, N1, N2, N3, REM	30 s epoch; 129 features: Welch spectral analysis; k-means clustering based feature weighting (KMCFW)	K-nearest neighbour (KNN) and C4.5 decision tree	4 recording, EEG	55.88% by k-NN; the weighted sleep stages with KM-CFW has been recognized with 82.15% success
Fraiwan et al. [21]	W, S1, S2, S3, S4, REM	30 s epoch; Entropy on CWT, used three different mother wavelets	Linear discriminant analysis (LDA)	32 recording from MIT-BIH, Single EEG	Accuracy 84%, kappa coefficient 0.78.
Estévez et al. [151]	W, S1, S2, S3, S4, REM	Amplitude of EOG, EMG, short time FFT, power Spectral density on FFT	Continuous fuzzy reasoning scheme	33 recordings, EEG(C3-A2 and C4-A1), EMG and EOG	W: 34%, N1: 43%, N2: 51%, N3: 82%, REM: 82%,
Helland et al.[8]	W, N1, N2, N3, REM	30 s epoch; power (P) and beta/delta, alpha/delta, theta/delta, beta/theta, alpha/theta, beta/alpha, beta/P, alpha/P, theta/P, and delta/P; heart rate variability (HRV) parameters	LDA	10 recording, EEG, ECG, and respiratory signals	90% Just EEG; By including EMG and respiratory signals 93%, Agreement with visual 61%
Tagluk et al.[154]	REM, S1, S2, S3, S4	5 s epoch; 5 features	Neural Network BP MLP, RUM with momentum	21 recordings, EEG(C3-A2), EMG and EOG	W: 70.5%, NREM :82.6% REM: 38.3%
Chapotot and Bequ [155]	W, N1, N2, deep N3, REM, MT	20 s epoch with small subset of 2 s epochs; 16 features: Shannon entropy, Hjorth activity, mobility and complexity, Hurst exponent, spectral edge frequency 95%, RP	Neural Network BP MLP, and flexible decision rules	48 recordings, EEG, EMG	W: 34%, N1: 43%, N2: 51%, N3: 82%, REM: 82%, MT: 13%

in the signal collection and additional processing.

Some of the most important published works are summarized in Table 3.2. Generally, the current ASSC methods consist of:

Preprocessing and artifact removal

Artifacts removal processing in sleep staging aim to apply the methods that minimize the effects of the artifacts such as movements and respiration. All PSG records are typically affected

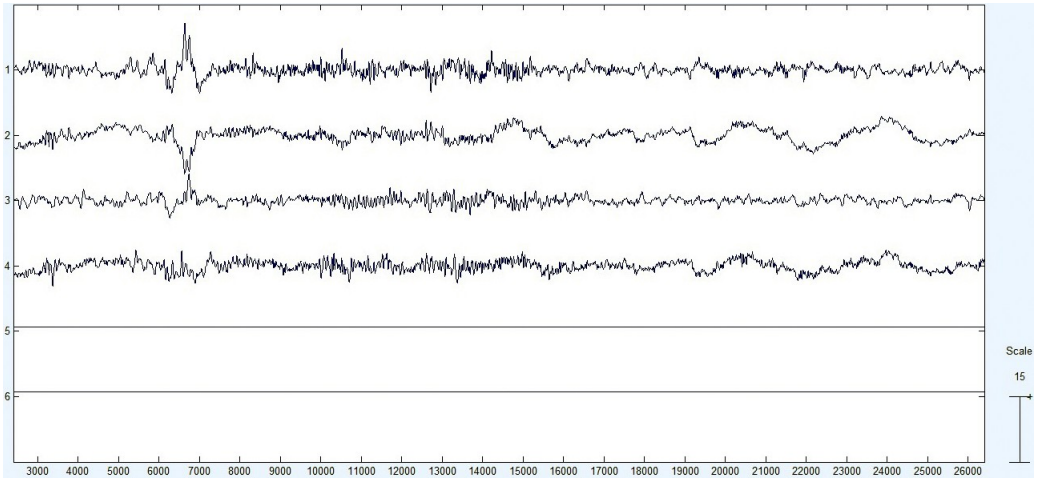

FIGURE 3.3: Example of three types of artifacts. Subject movements, around 6000 until 7000 ms; muscular artifact around 9000 until 16000 ms and small electrodes movement around 16000 until 26000 ms. Channels 1, 2, 3, 4: EEGs; Hypnograms 5: visual sleep scoring and 6: automatic sleep staging.

by some artifacts [156]. It is important to know what kinds of artifacts affect the signals and how to avoid their interference in results of sleep staging. Depending on the origin, there are two kinds of artifacts:

1. Technical artifacts: the origin of these artifacts are not the subjects. The most common technical artifacts are due to the interference of power sources, highlights abrupt transitions, electrodes movements (see Fig. 3.3) , operational problems of electrodes (lack of conductive paste, degradation of contacts and impedance fluctuations) or of the acquisition equipment [157], [158]. This kind of artifacts are usually minimized by applying Notch filter (to eliminate power source interferences); high-pass filters (to eliminate slow oscillations of signal due to small electrodes movements) and by modifying the misplaced electrodes [159].

2. Biological artifacts: the source of that artifacts are the subjects. The most common biological artifacts are ocular artifacts (due to blinking eyes or lateral ocular movements which affect the signal of electrodes placed on frontal and front-polar regions), slow variations and muscular artifacts (due to body movements or muscular tension, mainly in frontal and temporal regions, (see Fig. 3.3), and heart activity interference and body movements [158]. This type of artifacts is usually minimized by applying low-pass filters (to eliminate muscular artifacts) and automatic artifacts correction algorithms [159].

Feature extraction and selection

Many features have been employed in the ASSC methods. The features are extracted from PSG signals and are then used as input for a classifier that provides sleep scoring. Several works deployed time-frequency domain transforms such as discrete Wavelet transform (DWT), Hilbert Huang transform (HHT), and fast Fourier transform (FFT)[5, 12, 22, 152], and extracted time-frequency based features. Moreover, due to the effectiveness, temporal and frequency features were additionally used.

To select the most relevant feature for sleep staging, several feature selection method including, mRMR, sequential feature selectors, harmonic methods, fast correlation based feature selection (FCBF), ReliefF, t-test, and Fisher score algorithms were used [5, 20, 160].

Classification, Clustering and Decision making

Different parametric and nonparametric methods have been applied in the classification process of sleep staging. These methods learn the patterns represented from the features and map the ASSM rules into classification algorithms such as random forest classifiers, artificial neural networks (ANN), fuzzy logic, the nearest neighbour, linear discriminant analysis (LDA), support vector machine (SVM) and kernel logistic regression (KLR) [12, 18, 22, 28, 152, 153, 161]. On the other hand, clustering-based methods, group the sleep stages based on the similarity on their patterns. Self-organizing feature map (SOFM) and K-mean clustering were already applied to Harmonic parameters, and ratio band energy and Hjorth parameters [162]. The data are grouped into several cluster. The identified clusters are mapped into the standard sleep stages.

Several attempts at ASSC, reported in literature, have exhibited some success. Classification accuracies vary widely among the methods reported in scientific literature. However, there is no consensus about the best features, channels, and classification models for ASSC. Rigorous comparisons between the reported systems cannot be done since they differ in recording conditions and validation procedures. State-of-the-art results, summarized in Table 3.2, show agreement level with the manual scores ranging from 55% to 85%. Even though the success in many works, it is commonly accepted that the existing automatic methods are not yet enough accurate and reliable.

Chapter 4

Domain Adaptation: A Survey

This chapter provides a survey on variants of domain adaptation (DA) and its corresponding approaches. First, after describing the domain adaptation concepts, different variants of DA are considered in Section 4.3. Next, the state of the art of domain adaptation is reviewed with focus on the methodologies described in the literature, regardless of their specific application domains. Finally, in Section 4.5 a brief review on instance weights estimation approaches are presented.

4.1 Introduction

The performance of classical learning methods is often affected by several factors, including,

1. The non-existence of enough labeled training data: Nowadays, collecting unlabeled data is considerably easier, due to the availability data in the internet, and low cost ways for data collection; however, their labeling is expensive and time consuming.

2. The distributional change, or domain shift, that occurs between training and testing data. The essential assumption of independent and identical distribution (*i.i.d.*) is violated in real-world applications and classical learning methods are no longer consistent and yield a drop in general performance. Indeed, when the distribution changes, the statistical models based on training data need to be rebuilt.

Domain adaptation (DA) addresses the problem of adapting the obtained model of source domain in order to alleviate the distribution mismatch between the domains, where the objective

is a recognition task on the different but related data distributions [31]. Indeed, a DA method aims to bridge the distribution differences between source and target domains, by building a classifier using the labeled training data of the source domain that performs well on the target domain. For domain adaptation, the dense regions of a large fraction of instances $\boldsymbol{x} \in \mathbf{X}$ with the two distributions P_s and P_{ta} should have a reasonable overlapping, *i.e.* $P_{ta}(y|\boldsymbol{x})$ should be close to $P_s(y|\boldsymbol{x})$ [61].

The domain adaptation is a fundamental problem in machine learning and it started gaining much attention in many areas of applied machine learning [163–168]. The earliest work of learning from different training and test distributions was in the statistics and econometrics communities [169]. In another preliminary work, Good et al.[170] used the hierarchical Bayesian models to share training data across related learning tasks. Furthermore, in another seminal work the linear regression models were used to correct the sample selection bias problem [69]. Domain adaptation is a large area of work; depending on the types of distribution mismatch it was considered under different names as follows:

1. Labeling difference: model shift [62] or concept drift [63].
2. Priors difference: target shift [67], class imbalance [68] or conditional shift [67].
3. Instance difference: covariate shift or sample selection bias [69, 70].

Some concepts, notation and methods, used in the context of domain adaptation, are summarized in the following sections.

4.2 Notation

Let us introduce some notation concerning domain adaptation.

- $X = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^\top$ refers to a $N \times M$ matrix of N samples (random variable) with vector elements, where $\boldsymbol{x}_i \in \mathbb{R}^m$ denotes an M-dimensional feature vector.
- $\boldsymbol{y} = (y_1, \ldots, y_n)^\top$ denotes the corresponding labels (random variable) of the $\boldsymbol{x}_i$ samples, where $y_i \in C$, and $C = \{c_1, c_2, ..., c_k\}$ denotes the set of class labels.
- $D_s = \{(\boldsymbol{x}_i^s, y_i^s)\}_{i=1}^{N_s}$ refers to a source domain (training domain). There are always a relatively large amount of labeled data for source domain.

- $D_{ta,u} = \left\{ \left(\boldsymbol{x}_i^{ta,u} \right) \right\}_{i=1}^{N_{ta,u}}$ refers to a set of unlabeled data of the target domain (test domain).
- $D_{ta,l} = \left\{ \left(\boldsymbol{x}_i^{ta,l}, y_i^{ta,l} \right) \right\}_{i=1}^{N_{ta,l}}$ refers to a small amount of labeled data from the target domain.
- $P(\boldsymbol{x}, y)$ denotes the true underlying joint distribution of $\boldsymbol{x}$ and y, which is unknown. In domain adaptation, the joint distribution $P(\boldsymbol{x}, y)$ in the target domain differs from that in the source domain. We therefore use $P_{ta}(\boldsymbol{x}, y)$ and $P_s(\boldsymbol{x}, y)$ to denote the true underlying joint distribution in the target and test domains, respectively.
- Similarly, $P_{ta}(\boldsymbol{x})$ and $P_s(\boldsymbol{x})$ denote the true marginal distributions of $\boldsymbol{x}$, and $P_{ta}(y|\boldsymbol{x})$ and $P_s(y|\boldsymbol{x})$ denote the true conditional distributions, $P_{ta}(\boldsymbol{x}|y)$ and $P_s(\boldsymbol{x}|y)$ denote the class conditional distributions, and $P_{ta}(y)$ and $P_s(y)$ denote the class priors, in the target and source domains, respectively.

4.3 Domain Adaptation Variants

There are several variants of domain adaptation methods, which improve the performance under domain variation. Depending on the labeled data, considered in the target domain, the domain adaptation methods can be divided into the following categories: supervised, unsupervised, and semi-supervised approaches. However, the existing adaptation methods can be further divided into: multi-domain, and heterogeneous domain adaptation approaches.

4.3.1 Supervised Domain Adaptation

Supervised domain adaptation (e.g. [166, 171]) assumes that, there is a large amount of labeled source data $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ and only a limited (comparatively small) amount of labeled data from the target domain $D_{ta,l} = \left\{ \left(x_i^{ta,l}, y_i^{ta,l} \right) \right\}_{i=1}^{N_{ta,l}}$. The source (out-of-domain) data are considerably more than the target data (also known as in-domain data), *i.e.* $n_s \gg n_t$. The aim is to leverage the limited in-domain data together with the out-of-domain data in order to build a model that performs well on the new target domain (Fig. 4.1).

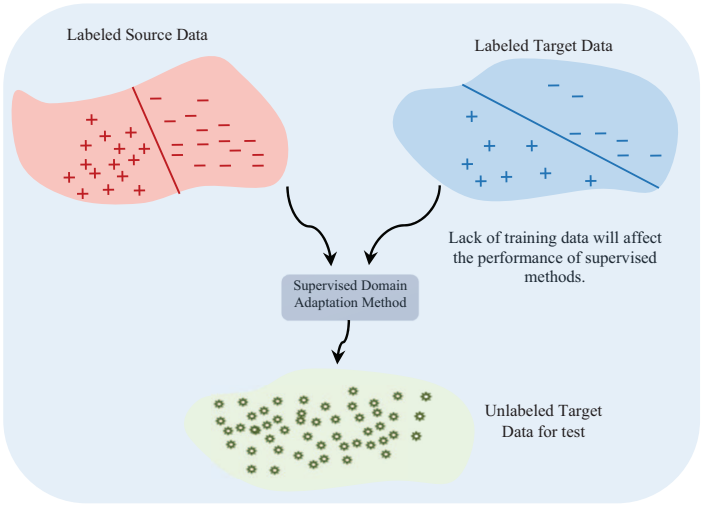

FIGURE 4.1: Supervised domain adaptation; source domain and target domain are labeled, $n_s \gg n_t$.

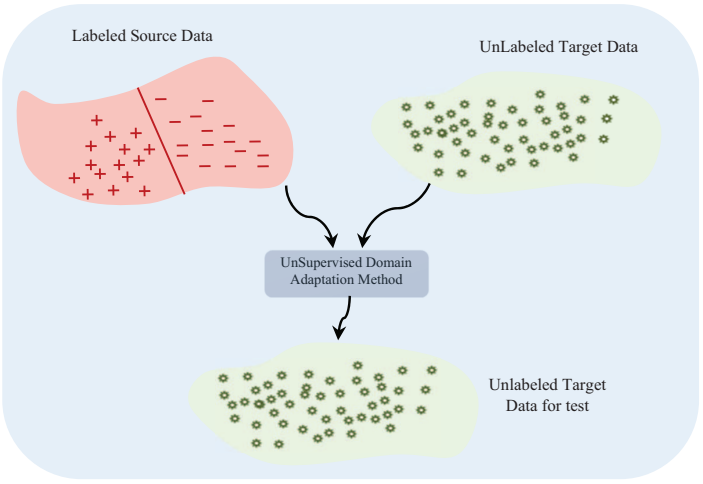

FIGURE 4.2: Unsupervised domain adaptation; only unlabeled data are available in the target domain.

4.3.2 Unsupervised Domain Adaptation

In contrast, unsupervised domain adaptation (*e.g.* [164, 172, 173]) does not require any labels from the target domain. The goal is to build a model that performs well on the new target domain by using the labeled out-of-domain (source) data $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ and the plenty of unlabeled data $D_{ta,u} = \left\{\left(x_i^{ta,u}\right)\right\}_{i=1}^{N_{ta,u}}$ from the target domain (Fig. 4.2). Since the labeling process in many real-world applications is expensive, time consuming and impractical, thus unsupervised domain adaptation approaches, which rely on purely unsupervised target data, are more realistic and useful [89]. The main key of unsupervised domain adaptation methods is how to effectively leverage unlabeled target data [172].

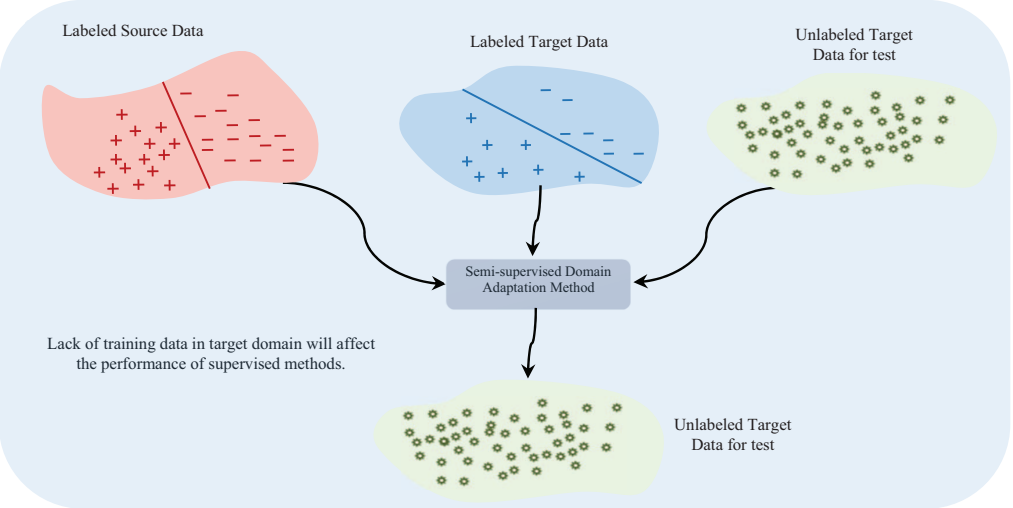

FIGURE 4.3: Semi-supervised domain adaptation. There are both labeled and unlabeled data available for the target domain. Number of labeled target data is small.

4.3.3 Semi-supervised Domain Adaptation

Semi-supervised domain adaptation (*e.g.* [90, 174, 175]) assumes that a small number of labeled samples from the target domain are available during training (Fig. 4.3). The semi-supervised DA approaches exploit the labeled source data $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ as well as a usually rather limited amount of labeled target data $D_{ta,l} = \left\{\left(x_i^{ta,l}, y_i^{ta,l}\right)\right\}_{i=1}^{N_{ta,l}}$ together with a lot of unlabeled target data $D_{ta,u} = \left\{\left(x_i^{ta,u}\right)\right\}_{i=1}^{N_{ta,u}}$. Semi-supervised approaches have been used in many areas including, speech and language processing [98], natural language processing [176] and more recently in computer vision application [177, 178].

4.3.4 Multi-source Domain Adaptation

Multi-source DA approaches [93, 179, 180] use the data of more than one source domain. The simplest way to use multi-source domains is to add up all the sources as one domain. However, the main drawback of this approach is to ignore the difference among the source domains. The multi-source domain adaptation approaches can be divided into the following groups:

- Feature representation approaches [181, 182].
- Approaches based on combination of pre-learned classifiers [183, 184]. These multi-source DA approaches train a classifier per source and combine these multiple base classifiers to maximize their combined accuracy on the target domain.

In multi-source domain adaptation, accompanying of supervised and unsupervised approaches is possible. The key issue of multi-source DA approaches is how to select good sources and samples for the adaptation.

4.3.5 Heterogeneous Domain Adaptation

Heterogeneous domain adaptation (HDA) [80, 83, 185, 186] assumes that the features from the source and the target domains are heterogeneous with different dimensions (Section 2.2.1.1). The data often come from multiple sources and modalities, thus it results in distribution difference among the domains (Fig. 2.9). This setting is common in real-world applications and domain adaptation in this scenario is very challenging.

4.4 Domain Adaptation Approaches

There are some directions of work being proposed to deal the domain adaptation problem. Several state-of-the-art adaptation approaches have been proposed to relate the source and target domains. In survey articles [61, 176, 200] different categories for domain adaptation approaches may be found. The approaches, regardless of their specific applications, can be divided into some categories including: instance-based, feature-representation based, self-labeling based, clustering and transformation based, and dictionary based methods. Almost all the approaches aim to identify the relevant knowledge, from the source domain, that can be beneficial for learning in the target domain. Based on this categorization, a review of the main domain adaptation approaches are summarized in Table 4.1.

4.4.1 Instance Based Approaches

Instance-based approaches [17, 167] assume that there is a subset of labeled instances in the source domain that are relevant to the target domain. In order to alleviate the distribution difference in the source and target domains, these approaches apply some specific weights to the instances of source domain. There are some major techniques for instance-based methods including, instance-weighting, weighted combination, instance pruning and domain re-sampling.

Instance weighting: Covariate shift, a special form of domains difference, is the most widely used assumption in the context of DA. The covariate-shift based adaptive models have been

TABLE 4.1: Some of the recent domain adaptation (DA) approaches.

Category	Approaches	Description	DA type
Instance-based	Instance weighting	Importance reweighting in order to adapt the loss function with noisy labels [65], and to reduce the target and conditional shift [67]; Instance reweighting by direct importance weighting methods [17] such as: ULSIF [26], KMM [101], KLIEP [187] for covariate shift adaptation.	Unsupervised DA
	Weighted combined	Assigning domain dependent weights [188].	Semi-Supervised DA
	Bayesian priors	Maximizing an objective function to adapt a constructed Bayesian prior from labeled instances [189].	Semi-Supervised DA
	Instance pruning	To remove misleading instances [167].	Semi-Supervised DA
	Domain resampling	A subset of labeled data instances which are named landmarks, in the source domain will distribute most similarly to the target domain [190].	Unsupervised DA
Change of the features representation	Distribution similarity	Penalize and/or remove features that vary between domains, and using generalizable features [191].	Unsupervised DA
	Latent feature learning	Construct new features by analyzing large amount of source and target domain data [192].	Unsupervised DA
	Feature augmentation-based	Make of domain specific copy of the features, and maps each feature onto an augmented space[71].	Semi-supervised DA
Self-labeling	Self-training	A model is built using source data. It is improved by annotated unlabeled target data of the previous iteration. In some iterations, the automatically labeled target data is then added to the source data and the model is updated [193].	Unsupervised DA
	Co-training (Multiview learning)	A single optimization problem, which simultaneously learns a target predictor, is formulated. Moreover, a split of the feature space into views, and a subset of source and target features to include in the predictor [194].	Unsupervised DA
Clustering and Transformation-based	Feature transformation-based	Feature adaptation by learning a transformation [175].	Unsupervised DA
	Subspace-based	Exploiting the subspaces spanned by features of the source and target domains. These methods assume the existence of a single subspace for the entire source and target domains [185].	Unsupervised DA
	Manifold-based	Construct a manifold and use incremental learning by gradually following the path between source and target domains [195], Generating domain invariant features by using use the covariance matrix to represent a domain and constructing a geodesic path between the source and target domains on a Riemannian manifold [196].	Unsupervised DA
	Graph-based	The labels from labeled samples propagate to unlabeled nodes based on weights along the paths between them [197].	Semi-supervised DA
	Parameter adaptation-based	Adaptive multiple kernel learning: learn a kernel function based on multiple kernels [198].	Semi-supervised DA
Dictionary-based	Transforming a dictionary learned from a domain to other domains	Data can often be coded using few representative atoms in some dictionary. These methods generate a set of intermediate dictionaries, which smoothly connect the source and target domains [199].	Semi-supervised DA

shown to be useful in target domains [99]. Among different domain adaptation approaches (Table 4.1), instance-weighting (*importance reweighting*) methods have proven almost successful for adaptation of the aforementioned differences (see Section 2.2) between source and target domains. The main focus of instance weighting approaches consists on: 1) comparing the distributions of the source and target instances; 2) weigthing (down/up-weight) an instance in the source domain, based on its importance on the target domain [17, 26, 201]; 3) training a single model over the combined source and target weighted instances.

The instance weighting methods make use of all available instances, including $D_s, D_{ta,u}, D_{ta,l}$. The calculated weights $w(\boldsymbol{x})$ are generally incorporated into the loss function of each training (source) instance. It means that, the log-likelihood terms are weighted according to their importance.

Due to their effectiveness, several instance weighting based methods have been proposed [187]. Shimodaira [17] was the first researcher, proposes an instance weighting approach for covariate shift adaptation. The method reweight the log-likelihood of each source instance, in order to minimize the loss of the model in the target domain. Due to the distribution differences between the source and target domains, empirical risk minimization (ERM) method is not generally consistent anymore. Therefore, the optimal model for the target domain will be approximated via instance weighting of the optimal model in the source domain. Acording to Shimodaira's approach [17], ERM under covariate shift attempts to find the best parameter value of $\boldsymbol{\theta}$ that minimizes a loss function $l(\boldsymbol{x}, y, \boldsymbol{\theta})$ in the target domain,

$$E_{P_{ta}}\left\{l(\boldsymbol{x}, y, \boldsymbol{\theta})\right\} = E_{P_s}\left\{\frac{P_{ta}\left(\boldsymbol{x}, y\right)}{P_s\left(\boldsymbol{x}, y\right)} l(\boldsymbol{x}, y, \boldsymbol{\theta})\right\}. \tag{4.1}$$

Therefore, under covariate shift, $\boldsymbol{\theta}$ can be estimated from the weighted ERM:

$$\sum_{(x_i, y_i) \in D_s} w(\boldsymbol{x}_i) l(\boldsymbol{x}_i, y_i, \boldsymbol{\theta}). \tag{4.2}$$

The weighting of the source instances by $w(\boldsymbol{x})$ provides a solution for domain adaptation problems. In the context of instance weighting, the density ratio

$$w(\boldsymbol{x}) = \frac{P_{ta}(\boldsymbol{x})}{P_s(\boldsymbol{x})} \tag{4.3}$$

plays an important role; however, the density ratio estimation suffers from the curse of dimensionality. To alleviate this problem several direct density ratio estimation methods were proposed. For a brief review see Section 4.5.

TrAdaBoost method, proposed by Dai et al.[202], iteratively changes the weights of the instances in each domain and effectively transfer the knowledge from source to target domain. It builds a high-quality classification model for the target data by utilizing a tiny amount of target instance plus a large amount of source instances. Similarly, Jiang and Zhai [167] proposed an iterative instance weighting method. This method trusts the assigned labels for the unlabled instances of the target domain. It retrains the model by adding the trusted labeled instances. Furthermore, there are several instance weighting methods to handle multiple source domains and to prevent negative transfer in domain adaptation [70, 187, 203].

Weighted Combination: This method assumes a union of source and target data, in which there are labeled training data from both domains. This method is useful in supervised domain adaptation setting. The idea is to change the importance of the source and target domain instances, by assigning domain dependent weights. The relative importance of each domain are integrated into the loss term of the SVM. it can be adjusted by setting domain-dependent cost parameters C_s and C_{ta} for the m and n training examples from the source and target domain, respectively:

$$\begin{aligned} &\underset{w,\xi,\psi}{min} && \frac{1}{2}\|w\|^2 + C_s \sum_{i=1}^{m} \xi_i + C_{ta} \sum_{i=m+1}^{m+n} \psi_i \\ &\text{s.t.} && y_i \left(\langle \mathrm{w}, \xi\left(X_i\right)\rangle + \mathrm{b}\right) \geq 1 - \xi_i \quad \forall i\ \xi\,[1,m] \\ & && y_i \left(\langle \mathrm{w}, \psi\left(X_i\right)\rangle + \mathrm{b}\right) \geq 1 - \psi_i \quad \forall i\ \psi\,[\mathrm{m+n},n] \\ & && \xi_i \geq 0 \qquad \forall i\ \xi\,[1,n] \\ & && \psi_i \geq 0 \qquad \forall i\ \psi\,[\mathrm{m+n},n] \end{aligned} \tag{4.4}$$

If the cost for one domain is significantly higher than for the other domain, the resulting solution is dominated by points from the domain with the highest cost. The weighted combination has then two model parameters (C_s, C_{ta}). It requires training on the union of the training sets, which means both within-domain interactions (in the sense of comparisons) as well as between-domain interactions will be considered [188].

Bayesian Priors: The idea of this approach is to integrate the prior information from the source domain. It uses labeled instances from the target domain to adapt the model built by

source instances. This approach works well in supervised domain adaptation, and it is useful when a small amount of labeled data from the target domain is available. The method utilizes a Bayesian framework where a Bayesian prior can be directly integrated into the maximum a posterior (MAP) estimation [204]. Thus, some prior knowledge about the classification model can be encoded into a Bayesian prior distribution $P(\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is the model parameter. In particular, instead of maximizing

$$\prod_{i=1}^{N} P\left(y_i \mid \mathbf{x}_i; \boldsymbol{\theta}\right), \tag{4.5}$$

it maximizes,

$$P(\boldsymbol{\theta}) \prod_{i=1}^{N} P\left(y_i \mid \mathbf{x}_i; \boldsymbol{\theta}\right). \tag{4.6}$$

In domain adaptation setting, it first constructs a Bayesian prior from labeled instances of source domain. Then, it maximizes the following objective function:

$$P\left(\boldsymbol{\theta} \mid D_s\right) P\left(D_{ta,l} \mid \boldsymbol{\theta}\right) = P\left(\boldsymbol{\theta} \mid D_s\right) \prod_{i=1}^{N_{ta,l}} P\left(y_i^t \mid \mathbf{x}_i^t; \boldsymbol{\theta}\right). \tag{4.7}$$

Instance Pruning: To deal with the labeling difference, (*i.e.* the difference between conditional probabilities $P_{ta}(y|\boldsymbol{x})$ and $P_s(y|\boldsymbol{x})$ for a considerable number of $\boldsymbol{x} \in \mathbf{X}$), a heuristic instance pruning method was proposed [167]. This method removes misleading instances from the source domain (by assigning zero weights to the respective instances) as follows: First, a classifier is trained using the labeled instances from the target domain. Next, the trained classifier is used to predict labels of the instances from the source domain (of which we know the true labels). Then, if the predicted label for a sample does not match with true label, this sample is removed from the source domain. In fact, the instance pruning method avoids using contradictory instances of the training domain. However, it should be pointed out that, the instance pruning approach can only expect to work if a reasonable number of labeled target instances are available. Otherwise, it might discard valuable information [188].

Domain Resampling: Resampling method aims to solve the class imbalance problem (Section 2.2.4). These methods attempt to resample the source domain instances so that the resampled instances roughly have the same class distribution as the target domain [205]. In these methods, under-represented classes are over-sampled and over-represented classes are under-sampled.

A landmark-based method was recently proposed for unsupervised domain adaptation [190,

206]. This method exploits a subset of labeled instances in the source domain that are distributed most similarly to the target domain. The key idea is that all instances are not equally created for adaptation. Thus, it exploits the most desirable instances for adaptation. A variant of Maximum Mean Discrepancy (MMD) is used to select instances from the source domain to match the distribution of the target domain [200].

4.4.2 Approaches Based on Changing the Feature Representation

This group of approaches assume that a better feature representation for learning in source and target domains exist, in which the unlabeled instances of the target domain can help to discover this common representation. These features will minimize the distribution differences between domains, while maintaining their main data characteristics. This group of approaches make some changes in the features representation $\boldsymbol{x}$ to better represent shared characteristics of both domains. In contrast to the instance-based methods, these approaches can be more effective in situations where a few features cause the domains difference. A change of feature representation can affect the marginal distribution $P(\boldsymbol{x})$, as well as the conditional distribution $P(y|\boldsymbol{x})$. Therefore, the final goal of this group of approaches is to find a suitable representation of $\boldsymbol{x}$, such that $P_{ta}(\boldsymbol{x}, y) = P_s(\boldsymbol{x}, y)$. Generally, if we can find a transformation function that transforms an observation $\boldsymbol{x}$, represented in the original form into another form $\boldsymbol{x}^{'}$, such that $P_{ta}(\boldsymbol{x}^{'}, y) = P_s(\boldsymbol{x}^{'}, y)$, then we do not need to adapt the domains. The instances of source and target domains have the same joint distribution as well as the same class label. The changing representation method is the basic idea in many research works such as [91, 164, 165, 192, 207, 208]. The approaches based on changing the feature representation can be categorized into three groups including: distribution similarity approaches, latent feature learning approaches, and feature augmentation-based approaches.

Distribution similarity approaches: These approaches explicitly make the distributions of the source and target domain instances similar,

- by penalizing or removing features whose statistics vary between domains [72, 191, 209];
- by learning a feature space embedding or projection, in which a distribution divergence statistic is minimized [92, 207, 210];

- by learning a weight vector on the *features* [191], or re-weighting the *features* (K-mer Reweighting) [211];
- by scaling the features in the source domain so that their values matches with the features in the target domain [72].
- by avoiding the training features types, that are absent in the target domain.

Moreover, some feature normalization methods can be also considered as feature representation approaches to domain adaptation [20].

Latent feature learning approaches: These approaches aim to construct new features by analyzing large amounts of unlabeled source and target domain instances [91, 164, 192, 212, 213]. This group of approaches assumes that, although the two domains are different, they still share some common latent components. In domain adaptation, it might happens that, some features in the target domain are zero or not available in the source domain, and vice versa. For example, let $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_M]$ be an original feature representation for each instance in source and target domains. The features vector $\boldsymbol{X}$ for each instance can also be considered as $\boldsymbol{X} = [\boldsymbol{X}_s, \boldsymbol{X}_{ta}, \boldsymbol{X}_c]$, where $\boldsymbol{X}_s$ denotes the source domain specific features, $\boldsymbol{X}_{ta}$ represents the target domain specific features, and $\boldsymbol{X}_c$ denotes the features that occur in both domains. Latent feature-based approaches exploit unlabeled source and target data together to construct a new feature representation that collects features in $\boldsymbol{X}_s, \boldsymbol{X}_{ta}$ and $\boldsymbol{X}_c$. These approaches construct aggregated features as linear combinations of the original features. To map the original feature vectors into a new feature space, a set of feature weights vectors $\{\boldsymbol{w}_k\}_{k=1}^{r}$ should be learned, where $\boldsymbol{w}_j = [w_{j,1}, w_{j,2}, ..., w_{j,M}]$. The new feature representation will be in the form of $\boldsymbol{X} = [l_1, \ldots, l_r]$, where:

$$l_k = \sum_{i=1}^{M} w_{k,i} x_i \tag{4.8}$$

Since l_k is a sum of features including $\boldsymbol{X}_{ta}$, thus the classifier implicitly uses the target features during the training. Principal component analysis (PCA), structural correspondence learning (SCL), and canonical correlation analysis (CCA) are examples of these approaches. Generally, these methods utilize the observed feature co-occurrences in unlabeled source and target instances to derive the new feature space [214].

Feature Augmentation-based: Daume-III [163] introduced one of the simplest and computationally efficient methods for domain adaptation. It assumes that, there is a large set of training data in the source domain, and a few labeled instances in the target domain. The goal is to make a domain specific copy of the original features for each domain. Each instance consists of two copies of the same feature, 1) one for capturing domain specific weight; 2) another for capturing domain independent weight. Each mapped feature $\boldsymbol{x}_i$ from the original source and $\boldsymbol{x}_j$ from the original target domain are defined as:

$$\psi^s\left(\boldsymbol{x}_i^s\right) = \begin{bmatrix} \boldsymbol{x}_i^s \\ \boldsymbol{x}_i^s \\ \mathbf{0}_N \end{bmatrix}, \quad \psi^{ta}\left(\boldsymbol{x}_j^{ta,l}\right) = \begin{bmatrix} \mathbf{0}_N \\ \boldsymbol{x}_j^{ta,l} \\ \boldsymbol{x}_j^{ta,l} \end{bmatrix} \tag{4.9}$$

where $\boldsymbol{x}_i^s \in D_s, \boldsymbol{x}_j^{ta,l} \in D_{ta,l}$, and $\mathbf{0}_N$ denotes a zero vector of dimension N. In source domain, each augmented feature consists of three components $< \boldsymbol{x}_i^s, \boldsymbol{x}_i^s, \mathbf{0} >$. The first component $\boldsymbol{x}_i^s$ is source domain specific version, the second component refers to commonality between source and target domains, and the last one means target domain specific version of $\boldsymbol{x}_i^s$. Correspondingly, each feature $\boldsymbol{x}_i^{ta,l}$ in the target domain is replicated as $< \mathbf{0}, \boldsymbol{x}_i^{ta,l}, \boldsymbol{x}_i^{ta,l} >$. For training the classifier, both source and target domain features are transformed using these augmented feature map. This method is simple to implement and works with any classifier. It can be easily extended to a multi-domain case by making more copies of the original feature space. Furthermore, a kernel version of this method is also derived in [71]. However, this method is not limited by the size of target domain as well as the needs for labeled training data. Moreover, a feature augmentation-based method, based on the idea of subspace learning was proposed in [185]. This approach introduces a common subspace for utilizing the heterogeneous instances from the source and target domains. First, the heterogeneous features from two domains are compared. Then, the source and target instances, with different feature dimensions, are projected into a latent features domain.

4.4.3 Self-labeling Approaches

Self-labeling approaches including self-training, co-training [57], and maximum likelihood linear regression (MLLR) [215] are interactive approaches. These approaches, first train an initial model using labeled source instances. Next, the created model is deployed to estimate labels of the target domain instances. Finally, the estimated target labels will be used in an iterative

strategy to build a new model. These approaches were applied to different domain adaptation settings.

Self-training/ Bootstrapping: Self-training is a general single-view bootstrapping algorithm to leverage unlabeled data. When it applies to domain adaptation, the only difference to the common self-training setup is that, the labeled data is from the source domain, while all unlabeled data comes from the target domain. These algorithms gradually shift the distribution of the training set from source to target by "labeling" the target inputs and adding the confident predictions to the training set. Several domain adaptation works exploited self-training in datasets with a small [216] or large number of instances [217]. To improve the results, re-ranking [218], the Expectation-Maximization (EM) algorithm [47], and dictionary learning [76] were used.

Co-training (Multiview Learning): Co-training [57] is a semi-supervised method based on the idea of multi-view learning. In real-world applications, instances often come in multiple views or styles (e.g. object recognition), yielding the differences between source and target instances. A direct comparison of instances across different views is not meaningful since they lie in different feature spaces. Similar to the self-training algorithms, co-training for domain adaptation assumes that, the instances are often given in pairs corresponding to different views, which is the main difference in comparison with normal co-training methods (*i.e.* $P_{ta}(\boldsymbol{x}, y) = P_s(\boldsymbol{x}, y)$).

Co-training and its variants have been applied to many applications. It was first introduced in NLP [219], where it was applied to classify webpages using the text of the page and the anchor text description of links pointing to the webpage. In another work [220], co-traing was used for cross-language sentiment classification, which leverages the English sentiment corpus for Chinese sentiment classification. A machine translation system was used to eliminate the language gap between the training set (English features) and the test set (Chinese features). Moreover, Co-training was also used to adapt a parser [221], which was trained on a news-ware and it was tested on other genres (*e.g.* broadcast conversation and weblog).

4.4.4 Clustering and Transformation-based Approaches

Feature Transformation-based: These methods aim to adapt the features across the source and target domains by learning a transformation. Generally, given a transformation function, which transform an instance $\boldsymbol{x}$ into another form, such that $P_{ta}(\boldsymbol{x}, y) = P_s(\boldsymbol{x}, y)$, there is no need for the domain adaptation. By transform the features, the instances of source and target domains will have the same joint distribution.
A feature transformation-based method was adopted for different object category adaptation in [175]. This method aims to learn a linear transformation given some form of supervision, and then to utilize the learned similarity function in a classification algorithm [175]. A more general case of [175] was proposed by Kulis et al. [222]. This method is not restricted by the same dimantionality of the domains as well as asymmetric transformations. Other transformation-based methods were proposed in [223] and [224].

Manifold-based: This group of approaches aim to transform the features using the manifold alignment [225]. These methods find a new latent space, in order to minimize the distance between the data points of the same class coming from different domains. The manifold-based methods attempt to maximally preserve the original local structure of the data. Manifold-based method for unsupervised domain adaptation was first proposed by Gopalan et al. [173]. The method was based on using incremental learning by gradually following the geodesic path between the source and target domains. Geodesic flows were used to derive intermediate subspaces that interpolate between the source and target domains.
Moreover, some effective approaches have been proposed by Wang and Mahadevan [226–228]. All these works encode the local similarities through a graph Laplacian based on a nearest neighbor adjacency matrix. These highly flexible methods were designed to handle datasets with different dimensionality and characteristic.

Graph-based: A graph is built between the labelled instances of the source domain and unlabeled instances of the target domain. The weights between the edges are computed according to various similarity measures including, feature based similarity and content based similarity. Two different graph-based approaches for cross-domain sentiment analysis were studied by Ponomareva and Thelwall [229] as follows:

- RANK method [230], which uses a ranking strategy to assign scores to the sentiment for the target domain documents;
- OPTIM method [231], which solves an optimization problem for labeling the documents with sentiments.

Moreover, a graph matching algorithm for adaptation in remote sensing was proposed [197]. This method transform instances of the source domain to the target domain using an appropriate nonlinear deformation. The eventually nonlinear transform is based on vector quantization and graph matching. By applying this mapping, the samples in source domain are projected onto the target domain.

Parameter-based: These approaches for DA assume that the source and target domains share some common parameters or prior distributions (*e.g.* class prior distributions), which can be beneficial for transfer learning [204, 213]. The parameter-based methods aim to encode the prior knowledge acquired in the source domain for improving the learning in the target domain. These methods deal with the adjustment of the classifier itself and have been adopted in combination with SVM classifiers. Adaptive SVM (A-SVM), a parameter-based method, to adapt concept classifiers across various video domains was proposed in [232]. This method aims to adapt one or more existing classifiers of any type to the new target domain. The goal is to learn a specific δ function between the original and adapted classifier using an objective function similar to SVMs. This δ function is added to the original decision function in order to properly model the instances of the new target domain. Moreover, two improvements of this work were proposed for object category detection [233] and visual concept classification [234]. In another work [213] an extension of the Perceptron was proposed for domain adaptation, where the adaptive components are parameters estimated from the unlabeled source and target domain data combined with background knowledge.

4.4.5 Dictionary-based Approaches

The idea of learning dictionary from data was first introduced in the seminal work [235]. It aims to code the high dimensional data using few representative atoms in some dictionary. Similar to the other classical learning methods, the models estimated by the dictionary learning methods may not be optimal if the target instances have different distribution than the

training instances.

Several dictionary learning-based methods have been proposed to deal with the domain mismatch problem [199, 236, 237]. An unsupervised domain adaptive dictionary learning method was proposed in Ni et al. [237], where the source and target domain instances are represented with their dictionaries. This method gradually captures the domain mismatch by generating a set of intermediate dictionaries, which iteratively alleviate the source and target domain differences. The source and target domains will connect via a series of intermediate dictionaries. Then, the intermediate dictionaries can be used to build a classifier under domain mismatch. In contrast, Shekhar et al. [199] proposed a semi-supervised dictionary-based DA method, which learn a dictionary to optimally represent the source and target domains. This method represents the instances of source and target domains into a low-dimensional common sub-space and then learns a shared dictionary. Other dictionary-based methods for DA were proposed in [238] and [236].

4.5 Instance Weights Estimation Approaches

Among the domain adaptation approaches, instance weighting-based methods have shown a reasonable performance to moderate the bias caused by covariate shift [17]. The computed weights $w(\boldsymbol{x})$ are generally incorporated into the loss function of each training (source) instance. It means that, the log-likelihood terms are weighted according to their importance. In the context of instance weighting, estimating the ratio of probability densities from two collections of source and target data

$$w(\boldsymbol{x}) = \frac{P_{ta}(\boldsymbol{x})}{P_s(\boldsymbol{x})} \tag{4.10}$$

plays an important role; it is here henceforth named *importance ratio*. A naive approach to compute the importance ratio consists of: 1) to estimate the training and test density functions from the sets of training and test samples separately; 2) to take the ratio of the estimated densities [239]. However, this naive approach is not effective, because in practice there is no effective parametric density model [187]. To cope with this problem, direct importance estimation methods, which do not involve density estimation, were developed. Several methods have recently been proposed for inferring individual weights for each training sample. For instance, non-parametric density estimation [17, 100] and kernel mean match-based methods

[101] have been proposed in the literature to directly estimate the importance ratio. Generally, approaches for importance ratio estimation make use of methods, such as the maximum mean discrepancies (MMD) [60], to compare the distributions of probability density functions from source and target domains. In what follows, some of the recent importance ratio estimation methods are summarized:

- The Kernel density estimator (KDE), which is a non-parametric technique, estimates a probability density function from the samples. KDE first estimates the density of source and target samples separately, and then obtains the importance ratio. The KDE method suffers from the curse of dimensionality, due to its two-step estimation process.

- The Kernel mean matching (KMM) [101] directly gives estimates of the importance ratio $w(\boldsymbol{x})$ (6.2) without going through density estimation. It re-weights instances in Reproducing Kernel Hilbert Space (RKHS), based on the MMD theory. In fact, the KMM attempts to estimate the importance ratio such that the MMD between nonlinearly transformed samples is minimized in RKHS. KMM is shown to work well, given that tuning parameters such as the kernel width are chosen appropriately.

- Sugiyama et al. [187] proposed Kullback-Leibler importance estimation procedure (KLIEP) for direct estimation of the importance ratio. The goal of this method is to estimate model parameters by minimizing the Kullback-Leibler divergence between the test and the reweighted training distribution [169]. It estimated importance ratio w(x) by means a liner model and try to find the best parameter for of the parameter vector. The tuning parameters in KLIEP can be optimized based on a variant of cross-validation. This procedure was extended by Tsuboi et al. [240] to large-scale problems.

- Kanamori et al. [26] proposed the least-squares importance fitting (LSIF), a model based method for direct estimation of the importance ratio. The importance ratio is modeled as a linear combination of a series of basis functions as follows,

$$w(\boldsymbol{x}) = \sum_{i=1}^{M} \theta_i K\left(\boldsymbol{x}, \boldsymbol{x}_i^{ta,u}\right) \tag{4.11}$$

  where $\boldsymbol{x}_i^{ta,u}$ are instances from $D_{ta,u}$, which randomly selected as the reference instances, $K(.,.)$ denotes a Gaussian kernel centered on these reference instances, and M denotes a subset of target domain instances, which randomly selected as reference instances. The

goal of LSIF is to learn the parameter θ_i by minimizing the squared loss of density-ratio function fitting. In fact, the model parameters are determined using squared error minimization.

- Due to a numerical problems, sometimes LSIF is not reliable in practice [31]. To cope with the problem of the LSIF, an approximation method called unconstrained-LSIF (uLSIF) has been introduced by Kanamori et al. in [26]. The objective function of uLSIF is also the same as LSIF; however, in optimization problem of uLSIF, the non-negativity constraint is dropped. The main advantage of this unconstrained formulation is that the solution can be computed just by solving a system of linear equations. Therefore, the computation is fast and stable. Moreover, the method is computationally very efficient [31].

- Relative uLSIF (RuLSIF) [241], an extension of uLSIF, uses the relative Pearson divergence to approximate relative density ratios.

Part II

Sleep Staging with and without Considering Distribution Mismatch

Chapter 5

Automatic Sleep Stage Classification

This chapter proposes an efficient subject-independent method with application in sleep-wake detection and in multiclass sleep staging (awake, non-rapid eye movement (NREM) sleep and rapid eye movement (REM) sleep). In Section 5.2, the methodology and algorithms of the proposed method are detailed. An extensive set of feature extraction techniques are applied, covering temporal, frequency and time–frequency domains. Then, the extracted feature set is transformed and normalized to reduce the effect of extreme values of features. The most discriminative features are selected through a two-step method composed by a manual selection step, based on features' histogram analysis, followed by an automatic feature selector. Finally, the selected feature set is classified using support vector machines (SVMs).

5.1 Introduction

Several methods for feature extraction representing different aspects of the signals were reported in ASSC systems. The selection of the most discriminative features for ASSC is still an open problem. Besides, some works such as [12, 21] used just EEG channels, whereas others [5, 22, 151] used EEG channels in combination with EOG and/or EMG channels. Therefore, to reduce the computational cost and to improve the classification performance, a combination of the best PSG channels and the selection of discriminative features are highly desirable.
To improve applicability of automatic sleep staging, an efficient subject-independent method,

henceforth named Sirvan supervised method for sleep staging *SSM4S* is proposed with application in sleep-wake detection and in multiclass sleep staging (awake, non-rapid eye movement (NREM) sleep and REM sleep). In turn, NREM is further divided into three stages denoted here by N1, N2, and N3. The maximum overlap discrete wavelet transform (MODWT), which is shift invariant, is applied to extract relevant features from time-frequency domain. The proposed method takes advantage of the features extracted, from PSG signals, in temporal and frequency domains. The extracted feature set is transformed and normalized to reduce the effect of extreme values of features. The most relevant features are selected through a two-step method composed by a manual selection step followed by a minimum-redundancy maximum-relevance (mRMR) based feature selection. The selected feature set is classified using support vector machines (SVMs).

5.2 Methodology and Algorithm Description

The proposed system is organized in various inter-operating parts as detailed in the Fig. 5.1: preprocessing, feature extraction, feature transformation and normalization, feature selection, and classification.

5.2.1 Preprocessing

The preprocessing of PSG signals, consisting of artifacts processing, is one of the most important issues to work in ASSC. It aims to apply artifact correction (whenever possible) in order to lose a minimum of data, and to identify the remaining artifacts so as not take them into account during the sleep stage classification.
Since the collected PSG signals are characterized by low signal-to-noise ratio (SNR), a preprocessing stage is applied to improve the quality of the signals; *i.e.* some channels of the recorded signals are filtered to eliminate noise and undesired background EEG, aiming to enhance the PSG signal quality and increase the SNR. The filtering stage comprises: 1) a notch filter to eliminate the 50 Hz electrical noise; 2) a bandpass Butterworth filter with a lower cutoff of frequency 0.3 Hz and higher cutoff of frequency 35 Hz for EEG and EOG channels, and a lower cutoff of frequency 10 Hz and higher cutoff of frequency 70 Hz for EMG channels. Moreover, the PSG signals were segmented in 30s epochs.

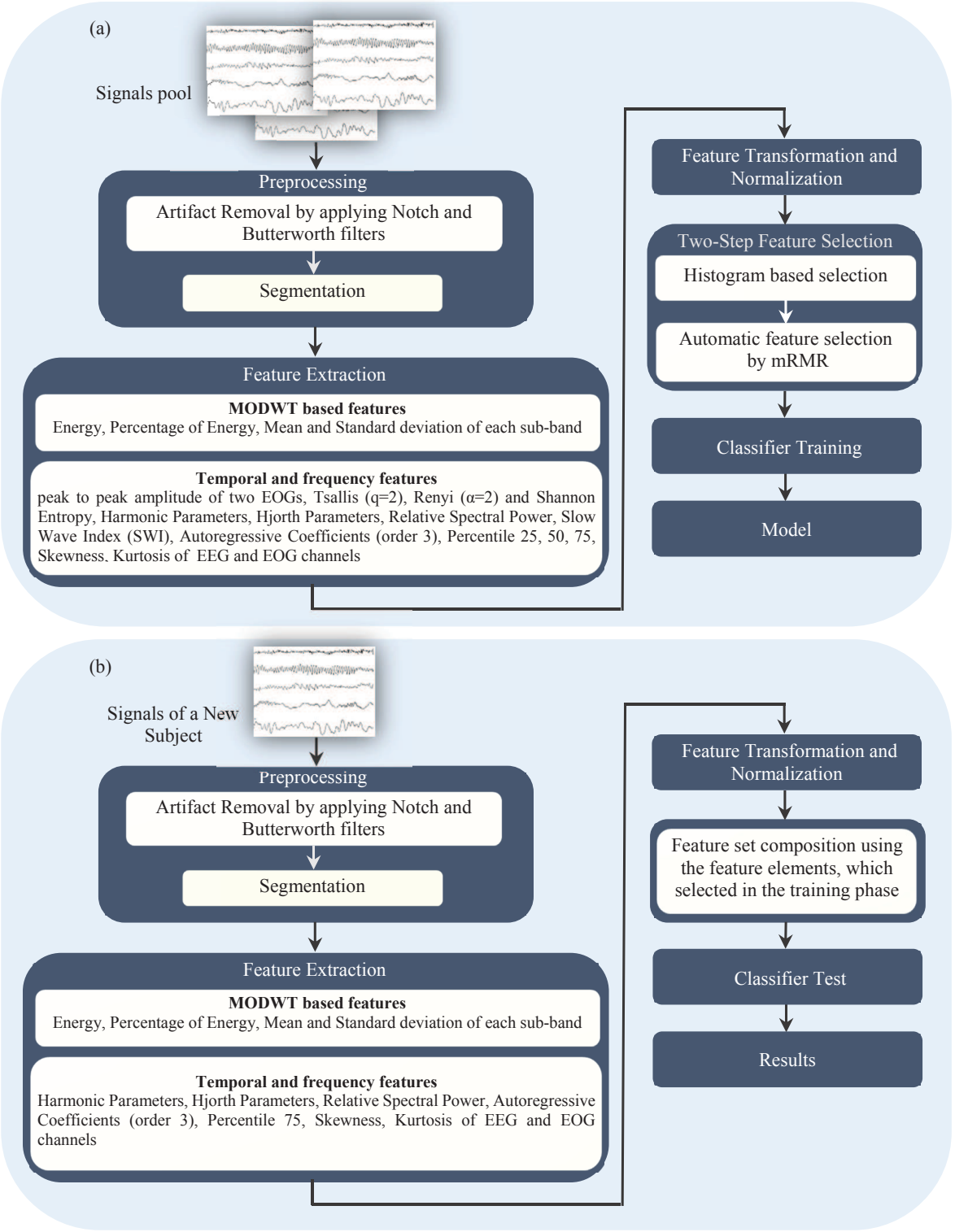

FIGURE 5.1: The *SSm4S* system architecture; (a) training flowchart, (b) test flowchart.

5.2.2 Feature Extraction

Since the sleep stages are characterized by the specific patterns of frequency contents, the PSG signals are traditionally analyzed in the frequency domain. However, further useful information can be extracted from temporal analysis of PSG signals. Once PSG signals are nonstationary, time-frequency based analysis is very useful. Thus, after preprocessing, some features are extracted using several methods in the time–frequency, temporal and frequency domains.

5.2.2.1 The Maximum Overlap Discrete Wavelet Based Features

Wavelet transform acts like " *a mathematical microscope zooming into small scales to reveal compactly spaced events in time and zooming out into large scales to exhibit the global waveform patterns*" [242]. The discrete wavelet transform (DWT) generates coefficients, which are local in time and frequency and represent the energy distribution of the signals. Therefore, signals can be reconstructed as a linear combination of the wavelet functions weighted by the wavelet coefficients. The maximum overlap discrete wavelet transform (MODWT) [243] is a DWT in which the operation of sub-sampling from an output filter is omitted. By giving up of the orthogonality property, the MODWT gains new features; although losing efficiency in computation, this transform does not have any restriction on the sample size and it is shift invariant. As a result, in the MODWT, the wavelet and scaling coefficients must be rescaled to retain the variance preserving property of the DWT. Although the components of MODWT are not mutually orthogonal, their sum is equal to the original time series. Additionally, the detail and smooth coefficients of a MODWT are associated with zero phase filters. This means that temporal events and patterns in the original signal are meaningfully aligned with the features in the multi resolution analysis. Furthermore, the MODWT is invariant to circularly shifting the original time series. Hence, shifting the time series by an integer unit will shift the wavelet and scale coefficients by the same amount. This property does not hold for the DWT because of the sub-sampling involved in the filtering process. In addition, the MODWT does not induce the phase shifts within the component series. The MODWT variance estimator is also preferred because it has been shown to be asymptotically more efficient than an estimator based on the DWT [244]. In this study a MODWT of depth 6 with Daubechies order four (db4) is applied to every 30s epochs with a sampling rate of 200 Hz. As shown in Table 5.1, the frequency ranges are broken down in a decomposition of $D1 - D5$, which almost correspond to δ range

TABLE 5.1: Frequency ranges corresponding to different decomposition levels.

Decomposition	Frequency range (Hz)
D1	25–50
D2	12.5–25
D3	6.25–12
D4	3.125–6.25
D5	0–3.125

($< 4\ Hz$), θ range (4−8 Hz), α range (8 −13 Hz) and β range (13−30 Hz). Finally, a set of statistical MODWT-based features are extracted to represent the time-frequency distribution of the EEG, EOG and EMG signals.

Energy and Percentage of Energy

Parseval's theorem is employed to extract the distribution of energy of the signals. According to Parseval's theorem, the energy of the distorted signal can be partitioned at different resolution levels. Mathematically it can be presented as:

$$E_i = \sum_{j=1}^{N} |D_{ij}|^2, \quad i = 1, \ldots, l \tag{5.1}$$

where $i = 1, \ldots, l$ denote the MODWT decomposition level. E_i is energy at decomposition level i, N is the number of the coefficients at each decomposition level and D_{ij} is value of a coefficient j at decomposition level i:

$$PE_i = E_i / \sum_{j=1}^{l} (E_j) \tag{5.2}$$

where PE_i is Percentage energy at decomposition level i [245].

Mean and standard deviation of each sub-band

In order to reduce the dimensionality of the extracted feature vectors, mean M_i and standard deviation Std_i at decomposition level i are used.

$$M_i = \Sigma_{j=1}^{N} (D_{ij}) / N \tag{5.3}$$

$$Std_i = \sqrt{\frac{\Sigma_{j=1}^{N} (D_{ij} - M_i)^2}{N - 1}} \tag{5.4}$$

5.2.2.2 Frequency and Temporal Features

Due to the importance of spectral and temporal analysis, some features are extracted in these domains. The following features are suggested in [5, 22, 246, 247].

Peak to Peak Amplitude

Peak-to-peak amplitude $P(X)$ is calculated by

$$P(X) = max(X) - min(X) \tag{5.5}$$

where $X = \{x_1, x_2, \ldots, x_N\}$ denotes set of signal amplitudes.

Entropy

The entropy gives a measure of signal disorder and can provide relevant information in the detection of some signal disturbs. Shannon entropy [248] is computed from histogram of the PSG samples, where $p_1, p_2, ..., p_n$ are a series of events; $p_i = n_i/N$, where N is the number of samples within the signal X, and n_i is the number of samples within the i_{th} bin. Shannon entropy H is defined as

$$H(P) = -K \sum_{i=1}^{N} p_i \ln p_i, \quad i = 1, \ldots, l \tag{5.6}$$

where K is Boltzmann constant.

Extensions of Shannons original work have resulted in many alternative measures of information or entropy. Renyi [249] was able to extend Shannon entropy to a continuous family of entropy measures as follows:

$$H_q(P) = \frac{1}{1-q} \ln \sum_{i=1}^{N} {p_i}^q \tag{5.7}$$

The Renyi entropy tends to Shannon entropy as $q \to 1$.

Furthermore, recently Tsallis entropy has proposed the use of the same quantity as a physical entropy measure which has some provoked considerable controvers [250]. Tsallis defined his entropy as [251] :

$$T_q = \frac{1}{q-1} \left(1 - \sum_{i=1}^{N} P_q^i \right), \quad i = 1, \ldots, l \tag{5.8}$$

TABLE 5.2: Spectral sub-bands used in PSD computation.

Bands	Sub-bands	Bandwidth $f_L - f_H$(Hz)
Delta	Delta 1	0.5-2.0
	Delta 2	2.0-4.0
Theta	Theta 1	4.0-6.0
	Theta 2	6.0-8.0
Alpha	Alpha 1	8.0-10.0
	Alpha 2	10.0-12.0
Sigma	Sigma 1	12.0-14.0
	Sigma 2	14.0-16.0
Beta	Beta 1	16.0-25.0
	Beta 2	25.0-35.0

Relative Spectral Power

Spectral analysis provides some of the most important features. For each signal X, an FFT squared modulus estimator was applied to estimate the power spectral density (PSD). The spectrum is divided into five frequency sub-bands as shown in Table 5.2. For each frequency sub-band, the Relative Spectral Power (RSP) is computed. This parameter is given by the ratio between the sub-band spectral power (BSP) and the total spectral power, *i.e.*, sum of all five BSP sub-bands [252]. Moreover, the spectral bands Delta, Theta and Alpha can be highlighted over slow wave bands by means of slow wave indexes defined by the following ratios:

$$TSI = \frac{BSP_{Theta}}{BSP_{Delta} + BSP_{Alpha}} \tag{5.9}$$

$$ASI = \frac{BSP_{Alpha}}{BSP_{Delta} + BSP_{Theta}} \tag{5.10}$$

$$DSI = \frac{BSP_{Delta}}{BSP_{Theta} + BSP_{Alpha}} \tag{5.11}$$

where TSI, ASI [253] and DSI stand for theta-slow-wave index, alpha-slow-wave index and delta-slow-wave index, respectively.

Harmonic Parameters

Harmonic Parameters of the PSG signals include three parameters: the center frequency (f_c) (12), the bandwidth (f_σ) (13) and the spectral value at center frequency (S_{fc}) (14). These parameters are defined as follows [22]:

$$f_c = \sum_{f_L}^{f_H} f p_{xx}(f)) / \sum_{f_L}^{f_H} p_{xx}(f) \tag{5.12}$$

$$f_\sigma = \sqrt{\sum_{f_L}^{f_H}(f - f_c)^2 f p_{xx}(f) / \sum_{f_L}^{f_H} p_{xx}(f)} \tag{5.13}$$

$$S_{f_c} = p_{xx}(f_c) \tag{5.14}$$

where $p_{xx}(f)$ denotes the PSD, which is calculated for the frequency bands $f_L - f_H$ (see Table 5.2). These parameters allow the analysis of a specific band in the EEG spectrum.

Hjorth Parameters

The Hjorth parameters provide dynamic temporal information of the PSG signal X. The Activity, Mobility and Complexity parameters are computed from the variance of X, $(var(X))$, and the first and second derivatives $X^{'}$, $X^{''}$ according to [254]:

$$Activity = var(X) \tag{5.15}$$

$$Mobility = \sqrt{var(X')/var(X)} \tag{5.16}$$

$$Complexity = \sqrt{var(X'') * var(X)/var(X')^2} \tag{5.17}$$

Skewness and Kurtosis

The skewness describes a measure of symmetry, or more precisely, the lack of symmetry of a distribution. Skewness of a signal X with N samples x_i is defined as

$$Skewness = \frac{1}{N}\sum_{i=1}^{N}(\frac{x_i}{\sigma})^3 \tag{5.18}$$

The Kurtosis is a measure of whether the data are peaked or flat relative to a normal distribution and as expressed by:

$$Kurtosis = [\frac{1}{N}\sum_{i=1}^{N}(\frac{x_i}{\sigma})^4] - 3 \tag{5.19}$$

Autoregressive Coefficients

Autoregressive (AR) modeling estimates the parameters of the mathematical models that describe the recorded PSG as an overlap of the real PSG and artifact interference [255]. AR

TABLE 5.3: Transformation methods [3].

#	Transformation	#	Transformation	#	Transformation
T1	$1/\sqrt{x}$	T4	$\lg(x)$	T7	$\lg((1/1+x))$
T2	$\sqrt[3]{x}$	T5	$\lg(1+x)$		
T3	$\sqrt{x}$	T6	$\arcsin(\sqrt{x})$		

model is a representation of a time series such that it specifies that the output variable depends linearly on its previous values. An AR process is defined by:

$$x_i = \sum_{j=1}^{N} a_j x_{i-j} + \varepsilon_i \tag{5.20}$$

where a_j are the autoregression coefficients, x_i is the series under investigation, which is a linear combination of its N past values and a purely random process ϵ_i. The noise term or residue, epsilon in the above, is almost always assumed to be Gaussian white noise [252].

Percentile 25, 50, 75

The percentile analysis provides some information about the amplitude of the signal and might be useful in discerning certain sleep stages [5]. The 25th, 50th and 75th percentile of the signal distribution is defined as

$$Percentile(X) = \left\lceil \frac{P}{100} \times N \right\rceil \tag{5.21}$$

where N is the number of samples x_i of the measured signal X and $P \in \{25, 50, 75\}$

5.2.3 Feature Transformation and Normalization

The extracted features are transformed and normalized in order to reduce the influence of extreme values. The transformation methods applied to each feature are described in Table 5.3 [3]. It was verified that some of those transformations improved the classification results. After a thorough experimental evaluation of each transform operator over extracted features, it was verified that the best classification results were attained with the transform

$$\mathbf{X} = arcsin(\sqrt{\mathbf{Y}}) \tag{5.22}$$

where Y denotes the feature matrix, and

$$\mathbf{X} = \{x_{ij}\}, \quad i = 1, 2, \ldots, N \quad j = 1, 2, \ldots, M \tag{5.23}$$

is the transformed feature matrix, where N and M denote the number of epochs and the number of features, respectively. Thereby this transform was adopted in the overall sleep staging system.

To avoid features in greater numeric ranges dominating those in smaller numeric ranges, as well as numerical difficulties during classification; each feature element of the transformed matrix $\mathbf{X}$ is independently normalized (scaled) by applying

$$\overline{x}_{ij} = x_{ij}/(max(\boldsymbol{x}_j) - min(\boldsymbol{x}_j)) \tag{5.24}$$

where $i = 1, 2, \ldots, N$, $j = 1, 2, \ldots, M$ and $\boldsymbol{x}_j$ is a vector of each independent feature.

5.2.4 Feature Selection

To reduce the dimension of the features vector and to find the most discriminative features, a two-step method that consists on a filtering and a wrapper phases is proposed: firstly, as detailed in Algorithm 1, the less discriminative feature-types are removed. In fact, by investigation on the feature distribution's histogram and corresponding hypnogram during a whole night sleep, the features with a higher discriminative histogram are selected (see Fig. 5.2). Then, in the second step, to select the best elements of each feature-type, the obtained feature vector is fed into an automatic feature selector. Feature selectors such mRMR, SFS, SFFS and SFBS are highly dependent on their objective function, however, we are going to find the most discriminative features for ASSC independently than selector/classifiers. Therefore, to find the most discriminative feature-elements, six different features selectors, were considered and their results were compared. Six different strategies for feature selection are described in the sequel.

minimal-Redundancy and Maximal-Relevance

The mRMR method uses the mutual information between a feature and a class to infer its relevance for the class. The mutual information of two random variables measures the mutual

Algorithm 1: Two-step Feature Selection Method.

Input: Extracted Feature Set.
Featurevector $= \{F_1, F_2, \ldots, F_N\}, F_i = \{y_1, y_2, \ldots, y_M\}$, $F_{iSleep} = \{F_{i1}, F_{i2}, .., F_{iz}\}$. *% N: number of feature type, M: number of the feature elements, % Z: number of sleep epochs.*
Output: Selected Feature Elements.

Step 1
Set $SelectedFeatureType = \{\}$, $d = 1$. *% initializes preliminary set of features*
while $(d <= N)$ **do**
    Comparison of feature distribution F_{dSleep} with the corresponding hypnogram (see Fig. 5.2) ,
    if *(F_{dSleep} has different distribution for sleep stages)* **then**
        Add F_d to SelectedFeatureType.
Step 2
SelectedFeatureType $= \{y_{11}, y_{12}, \cdots, y_{1M}, y_{21}, y_{22}, \cdots, y_{2M}, \cdots, y_{k1}, y_{k2}, \cdots, y_{kM}\}$ $\%y_{ij}$: an element of feature set
Initialize
a. $Z = 1, SelectedElement = \{y_{11}\}$,
b. $MaxPerformance = performance(\{y_{11}\})$.
while $(Z <= length(SelectedFeatureType))$ **do**
    i. Add y_{ij} to SelectedElement.
    ii. FinalSel= AutomaticFeatureSelection (SelectedElement)
    iii. **if** $(performance(finalSel) > MaxPerformance)$ **then**
        Maxperformance=performance(FinalSel)
    iv. Update SelectedElement to FinalSel

dependence between them [256]. Maximal Relevance is to search a feature set S satisfying:

$$max\left[D(S,c)\right], \quad D(S,c) = \frac{1}{S}\sum_{x_i \in S} I(x_i, c) \tag{5.25}$$

where $I(x_i; c)$ means the mutual information between feature x_i and class c. mRMR also uses the mutual information between features as redundancy of each feature. The minimal redundancy feature set R can be determined under condition

$$min[R(S)], \quad R(S) = \frac{1}{|S|^2}\sum_{x_i, x_{j \in S}} I(x_i, x_j) \tag{5.26}$$

where $I(x_i, x_j)$ indicates the mutual information between features x_i andx_j. The "minimal-Redundancy and Maximal-Relevance" (mRMR) criterion combines measures (5.25) and (5.26) as follows:

$$max_{\varphi}(D, R), \quad \varphi = D - R \tag{5.27}$$

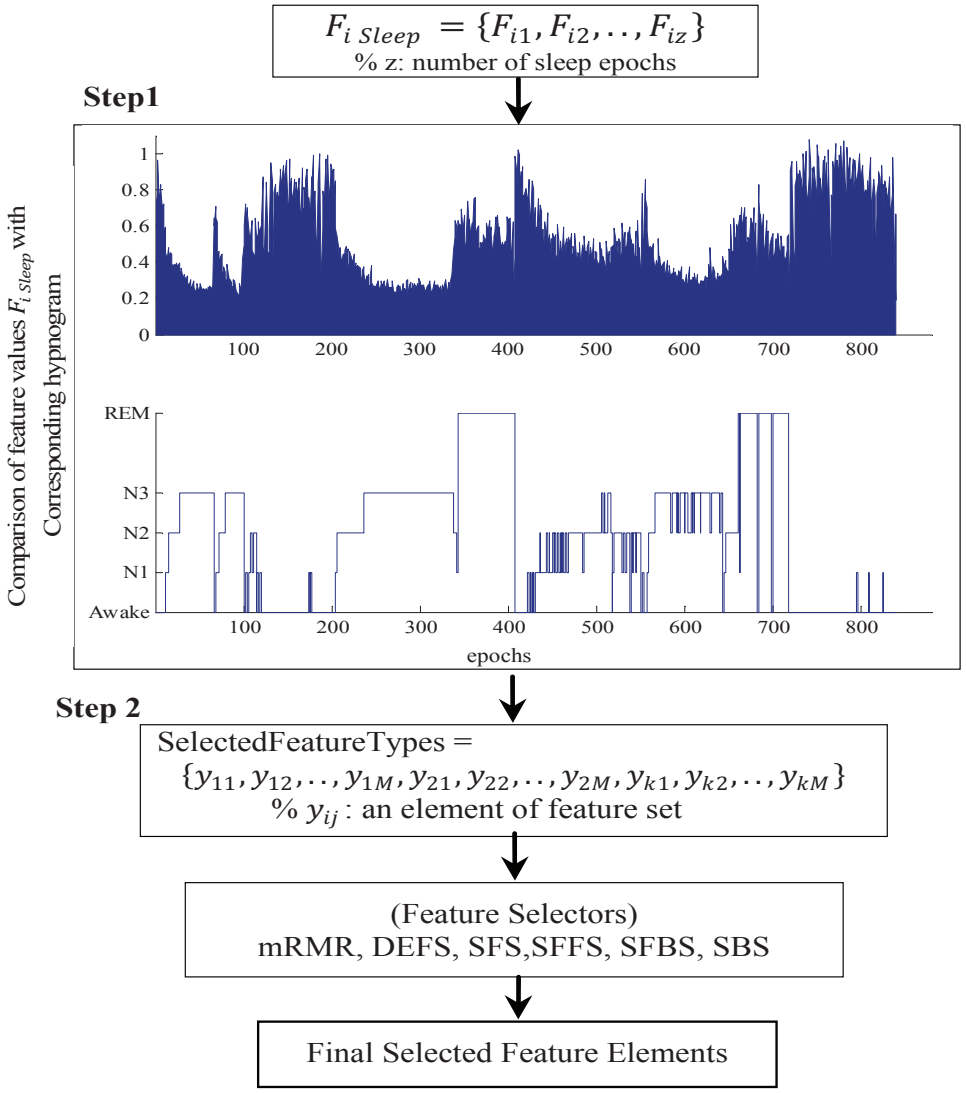

FIGURE 5.2: A sample of the two-step feature selector; step1: histogram of a feature values corresponding to each sleep stages during a whole night hypnogram to select the best feature types; step2: selection of the best feature elements.

Sequential Floating Feature-Selection Approaches

Sequential forward selection (SFS) [257], is the simplest feature selection method among the sequential strategies. It is a greedy search algorithm that iteratively determines an optimal subset of features. by adding one feature per iteration, if it increases the value of the chosen objective function. Sequential backward selection (SBS) [257] is similar to SFS but works in the opposite direction, i.e., it starts with the superset of all the features and sequentially removes one feature if it increases the value of the objective function.

The main drawback of these sequential approaches is that they gravitate toward local minima due to their inability to reevaluate the usefulness of features that were previously added or

discarded, *i.e.*, once a feature is added to or removed from the final set of features, it cannot be changed. Therefore, two expansions for SFS and SBS algorithms were proposed [258]. The sequential forward floating selection (SFFS) [258] finds an optimum subset by insertions (*i.e.*, appending a new feature to the subset of previously selected features) and deletions (*i.e.*, discarding a feature from the subset of already selected features) of selected features by the SFS algorithm. The sequential backward floating selection (SBFS) [257] is similar to SFFS but works in the opposite direction; it finds an optimum subset of features by insertions (*i.e.*, appending an already deleted feature to the subset of selected features) and deletions (*i.e.*, discarding a feature from the subset of already selected features) in the SBS algorithm.

Differential Evolution Feature Selection (DEFS)

DEFS approach uses a combination of differential evolution (DE) optimization method and a repair mechanism based on feature distribution measures. This method, utilizes the DE float number optimizer in the combinatorial optimization problem of feature selection. In order to make the solutions generated by the float optimizer suitable for feature selection, a roulette wheel structure is constructed and supplied with the probabilities of features distribution. These probabilities are constructed during iterations by identifying the features that contribute to the most promising solutions [259].

5.2.5 Classification

As classifier an SVM is applied [260]. Furthermore, linear discriminant analysis (LDA), Naïve Bayes (NB), and AdaBoost are used to compare the efficiency of the system.

Chapter 6

Importance Weighted Import Vector Machine for Adaptive ASSC

Current automatic sleep stage classification (ASSC) methods that rely on polysomnographic (PSG) signals suffer from the subjects and sessions' variability that make them unreliable in facing with new and different subjects. In real-world ASSC, the assumption of independent and identical distribution (i.i.d.) is no longer consistent. To develop the ASSC methods that are robust to the limitations of the subjects and sessions' variability are highly desirable. We assume that the sleep quality variants follow a covariate shift model, where the probability distributions of the sleep patterns change in the training and test subjects. Therefore, to alleviate the significant mismatch between training and testing subjects, importance weighting import vector machine (IWIVM), which is an adaptive classifier, is proposed. This adaptive probabilistic classification method, which is sparse and computationally efficient, can be used for unsupervised domain adaptation. In Section 6.5, we also introduce a reliable importance weighted cross validation (RIWCV), which is an improvement of importance weighted cross validation (IWCV), for parameter and model selection. The RIWCV avoids falling down in a local minimum, by selecting a more reliable combination of the parameters instead of the best parameters. Finally, in Section 6.6, an adaptive method for ASSC based on unsupervised covariate shift adaptation is proposed.

6.1 Adaptive Sleep Stage Classification

Most of the current ASSC methods reported in scientific publications are based on classical learning approaches including: 1) supervised methods, such as linear discriminante analysis (LDA), hidden Markov model (HMM), artificial neural networks (ANN), and kernel methods such as support vector machine (SVM) [5, 20, 21, 152, 155], and 2) unsupervised methods such as fuzzy clustering [153]. In these methods, it is implicitly assumed that the test data instances follow both the same probability distribution and the same feature space as the training instances. Sleep pattern characteristics may vary due to the different factors, such as the non-stationarity of data, recording environment changes, the health problems and the subjects' physical conditions. The properties of data evolve over time and change from one subject to another, thus, the training and test probability distributions are not necessarily the same in practice. The distribution discrepancy negatively affects the sustainability of traditional data mining techniques over time and prevents their applicability to new subjects. Therefore, the ASSC methods that rely on PSG signals suffer from the subjects and sessions' variability. To develop ASSC methods that are robust to the variations and that are adaptable to new subjects, is highly desirable.

6.1.1 Unsupervised Domain Adaptation for ASSC

As mentioned in Chapter 4, the domain adaptation addresses the problem of adapting the obtained generative or discriminative model of source domain in order to alleviate the mismatch between the domains distributions, where the objective is a recognition task on the different but related data distributions [89].

The distribution differences between the training and the test instances are represented in the joint probability distributions $P_{ta}(\boldsymbol{x}, y)$ and $P_s(\boldsymbol{x}, y)$. The ratio of joint distributional difference of source and target domains, $P_{ta}(\boldsymbol{x}, y)/P_s(\boldsymbol{x}, y)$, indicates how different the two domains at $(\boldsymbol{x}, y)$, where $\boldsymbol{x} \in \mathbf{X}$ denotes an instance from the observations set $\mathbf{X}$, $y \in Y$ denotes the class label y in class labels set Y, and $P_{ta}(\boldsymbol{x}, y)$ and $P_s(\boldsymbol{x}, y)$ denote the true underlying joint distributions in the target and source domains, respectively.

Estimating the probability $P(\boldsymbol{x}, y)$ is a challenging task, especially for a test instance, where the labels are not available. However, based on Bayes' rule, the joint probability distributions $P_{ta}(\boldsymbol{x}, y)$ and $P_s(\boldsymbol{x}, y)$ can be decomposed as follows:

$$P(\boldsymbol{x}, y) = P(y|\boldsymbol{x})P(\boldsymbol{x}) = P(\boldsymbol{x}|y)P(y). \tag{6.1}$$

Since the distributions of the training and testing instance are different, finding an optimal separator is more possible and practical than finding a suitable generative model or estimating the underlying distributions. Therefore, it will be expected that, the discriminative approaches such as SVM, perform better classification than generative approaches. The focus of these approaches are on the separation boundary or the posterior probability $P(y|\boldsymbol{x})$. Based on Bayes' rule (6.1), we decompose the joint probability distribution $P(\boldsymbol{x}, y)$ into $P(y|\boldsymbol{x})P(\boldsymbol{x})$.

In discriminative approaches, we are only interested in $P(y|\boldsymbol{x})$. The dense regions of a large fraction of instances $\boldsymbol{x} \in \mathbf{X}$ with the two distributions P_s and P_{ta} have a reasonable overlapping, (*i.e.* for different source and target instanecs, $P_{ta}(y|\boldsymbol{x})$ is close to $P_s(y|\boldsymbol{x})$) [61]. However, the classifier learned from the source instances does not perform well on the target instance. The difference between training and testing instances may be due to the difference of $P_{ta}(\boldsymbol{x})$ and $P_s(\boldsymbol{x})$, while the posterior distributions of the labels are the same in the two domains, which means that, the instances $\boldsymbol{x}$ may have different dense regions in two domains yielding a misspecified model for the target instances.

The most likely assumption for possible difference between the training and test subjects in ASSC is instance difference or covariate shift between the subjects. It means that, the marginal distribution $P(\boldsymbol{x})$ of the instances are different in the training and the test subjects, while the posterior distributions of the labels are the same in the two subjects.

Learning under covariate shift and the discriminative models based on covariate shift adaptation have been shown to be useful in the target domains. Among the different domain adaptation approaches (Section 4.4), instance weighting (importance sampling) methods have proven almost successful for adaptation of the instance differences. The main focus of instance weighting approach, consists on comparing the distributions of source and target subjects. In particular, the instance weighting technique, down/up weight an instance in the source subjects, based on their importance in the target subject [17, 26, 201].

On the other hand, even though the semi-supervised adaptation approaches have been used in many areas including, speech and language processing [98], natural language processing [176] and in computer vision application [177, 178], however, the labeling process in ASSC is expensive, time consuming and impractical. Thus, unsupervised domain adaptation approaches, which rely on purely unsupervised target data, are more realistic and useful [89]. Due to the effectiveness of unsupervised domain adaptation, several instance weighting based methods have

been proposed [187]. To the best of our knowledge, these methods (*e.g.* Table 4.1) are not sparse and successfully enough to alleviate the distributions mismatch of real-world domains such as ASSC [28]. Therefore, to cope with the subjects and sessions' variability, development of efficient and sparse unsupervised domain adaptation methods is highly desirable.

6.2 Import Vector Machine as Base Classifier

Selection of an suitable base classifier is one of the main effective factors on domain adaptation. SVM, have been proven almost successful for ASSC. However, comparing to SVM, import vector machine (IVM)[261] has the following advantages:

1. it offers a natural estimate of the probability $P(x) = \frac{e^f}{(1+e^f)}$, while the SVM only estimates $sign[P(x) - 1/2]$ or $sign\left[\log \frac{P(x)}{1-P(x)}\right]$;
2. it can naturally be generalized to the multi-class case through kernel multi-logit regression, whereas the SVM classifier uses strategies one-against-one and one-against-all for regression problems;
3. IVM is a sparse and computationally efficient classifier with computation cost $O(N^2q^2)$, where q is the number of import points. Since q does not tend to increase as N increases, the IVM can be faster than the SVM with the computational cost $O(N^3)$, especially for large training data sets.

Given the mentioned advantages, IVM is a suitable **base** classifier for domain adaptation in real-world problems. However, due to the structure of IVM, which is based on $i.i.d$ assumption, it is no longer consistent for the classification problem under covariate shift.
Importance weighted import vector machine (IWIVM), an adaptive probabilistic classification method, based on direct importance estimation, is proposed for unsupervised domain adaptation. This instance-weighting adaptation method, which is sparse and computationally efficient, can be used for asymptotically canceling the bias caused by covariate shift [26]. In what follows, we discuss the detailing descriptions of direct importance weighting methods as well as the proposed IWIVM.

6.3 Instance Weighting for Covariate Shift Adaptation

As mentioned in Chapter 4, among the domain adaptation approaches, instance weighting-based methods have shown a reasonable performance to moderate the bias caused by covariate shift [17]. The calculated weights $w(\boldsymbol{x})$ are generally incorporated into the loss function of each training (source) instance. It means that, the log-likelihood terms are weighted according to their importance. In the context of instance weighting, the density ratio (*importance ratio*),

$$w(\boldsymbol{x}) = \frac{P_{ta}(\boldsymbol{x})}{P_s(\boldsymbol{x})} \tag{6.2}$$

plays an important role. The naive approach kernel density estimator (KDE) is not so effective, since in practice there is no effective parametric density model [187]. However, direct importance estimation methods (Section 4.5), which do not involve density estimation, are more robust in calculating the importance ratio. Several methods have recently been proposed for directly inferring individual weights for each training sample. Generally, approaches for importance ratio calculation make use of methods such as the maximum mean discrepancies (MMD) [60] to compare the distributions, based on the mean of features in the Hilbert space induced by a kernel $K(.,.)$. A brief description of recent importance ratio calculation methods such as Kullback-Leibler importance estimation procedure (KLIEP) [187], least-squares importance fitting (LSIF) [26], unconstrained-LSIF (uLSIF) [26] and Relative uLSIF (RuLSIF) [241] are presented in Section 4.5.

6.4 Importance Weighted Import Vector Machine

Import vector machine (IVM), a kernel-based discriminative and probabilistic classifier, was first introduced by Zhu et al. [261]. The kernel logistic regression (KLR) is a basis for the IVM, which was made efficient by introducing the sparseness. The main idea of IVM is reflected in using the similarity in the curve shape of the loss term of SVM, and the negative logarithmic likelihood (*NLL*) function of the binomial distribution.

Bayes decision rule is the basis for the prediction of class label y for a test input sample $\boldsymbol{x}$. Based on Bayes rule the predicted label is c_j if

$$P\left(y = c_j \,|\boldsymbol{x}\,\right) > P\left(y = c_r \,|\boldsymbol{x}\,\right) \quad \forall\ j \neq r. \tag{6.3}$$

In multiclass logistic regression to estimate the class-posterior probabilities $P = [P_1, P_2, ..., P_n]$, where $P_i(x) = P(y_i = c_j | \boldsymbol{x}_i ; \boldsymbol{\theta})$, a parametric model should be computed. The following parametric model, in the form of the softmax function, is widely used:

$$P(y = c_j | \boldsymbol{x} ; \boldsymbol{\theta}_y) = \frac{\mathrm{e}^{f_{\theta_{y_i}}(\boldsymbol{x})}}{\sum_{n=1}^{c_k} \mathrm{e}^{f_{\theta_n}(\boldsymbol{x})}}, \tag{6.4}$$

where $\boldsymbol{\theta}_y = [\theta_{y_1}, \theta_{y_2}, \ldots, \theta_{y_n}]^\top$ is the parameter vector and $f_{\theta_n}(\boldsymbol{x}) \in H_K$ is a discriminant function corresponding to label n.

SVMs learn a predictive function $f(\boldsymbol{x})$ from the training data, which is based on ERM principle in the form of *loss+penalty*. There is a relationship between the SVM and the regularized function estimation in Reproducing kernel Hilbert space (RKHS). SVM optimization can be expressed by

$$J(f^*) = \min_{f \in H_K} J(f) = \min_{f \in H_K} \left(L(f(\boldsymbol{x}_1), ..., f(\boldsymbol{x}_n)) + \lambda \|f\|^2_{H_K} \right), \tag{6.5}$$

with regularization parameter $\lambda : \mathbb{R} \to \mathbb{R}$ and the loss function $L : \mathbb{R}^n \to \mathbb{R}$. Indeed, equation (6.5) is in the optimization form,

$$\min_{f \in H_K} \frac{1}{n} \sum_{i=1}^{n} [1 - y_i f(\boldsymbol{x}_i)]_+ + \frac{\lambda}{2} \|f\|^2_{H_K}, \tag{6.6}$$

where H_K is the RKHS generated by the kernel $K(.,.)$. Therefore, based on Representer theorem [262], [263], a solution for regularized ERM (optimal $f(\boldsymbol{x})$) is always in the form,

$$f^*(\boldsymbol{x}) = \beta_0 + \sum_{n=1}^{N} \theta_n \, K(\boldsymbol{x}, \boldsymbol{x}_n) \tag{6.7}$$

where θ_n has nonzero values only for the support vectors, and $\beta_0 \in \mathbb{R}$. This confirms that, to solve SVM optimization problem we only need to find parameter θ_n.

By replacing the loss function $(1 - yf)_+$ by the *NLL* of the binomial distribution $ln(1 + e^{-yf})$, the problem becomes a KLR problem.

In KLR, to estimate the parameter vector $\boldsymbol{\theta}$, *NLL* function $P(y | \boldsymbol{x} ; \boldsymbol{\theta})$ is calculated as

$$\min_{f \in H_K} \frac{1}{n} \sum_{i=1}^{n} ln[1 + e^{-y_i f(\boldsymbol{x}_i)}] + \frac{\lambda}{2} \|f\|^2_{H_K}, \tag{6.8}$$

where the regularization term $\frac{\lambda}{2}\|f\|^2_{H_K}$, was introduced to avoid overfitting problems; the regularization strength is regulated by the λ parameter. The KLR compromises the hinge loss function of SVM, and θ_n has nonzero values for all samples. Following the idea of representative vector machine (RVM), which is a unified framework for classical classifiers [264], and due to the similarity of the loss functions in (6.5) and (6.8), similar performances are expected. Similar to SVM, which is a method to maximize the margin of the training data, KLR can also be regarded as a margin maximizer. Therefore, based on the representer theorem (similar to 6.7), the discriminant function is in the following form:

$$f_{\theta_n}(\boldsymbol{x}) = \beta_0 + \sum_{i=1}^{N_s} \theta_{n,i}\ K\left(\boldsymbol{x}, \boldsymbol{x}_i^s\right) \tag{6.9}$$

where $K\left(\boldsymbol{x}, \boldsymbol{x}_i^s\right)$ is a kernel function.

Since, to train the classifier, KLR includes all training samples $\boldsymbol{x}$ (*i.e.* θ_n is nonzero for all training samples), the computational cost of KLR is expensive $O(N^3)$. To reduce the computational cost, similar to SVM, which only uses a few samples for defining the decision boundaries, the IVM classifier finds a sub-model to approximate the full model given by KLR using a subset-search. The sub-model consists of a subset of training samples, which are called *import points*. There are several methods for searching through all possible subsets to determine the best set. In IVM, the model parameters are optimized in a greedy procedure with simultaneous import vector selection. In particular, the subset is found by using both the samples and the output, *i.e.* the posterior probabilities. This idea distinguishes IVM from those that only use the samples, such as random sampling or methods that identify cluster representatives.

Under covariate shift the IVM classifier, which works based on the standard ERM method, is no longer consistent [31]. To cope with this problem, and to improve the performance of the import vector selection, a method called importance weighted import vector machine (IWIVM) (Fig. 6.1) is proposed. In order to systematically adjust the distributions, this method assigns the instance weights to the training samples based on their importance in the test domain. The instance weighting can compensate distribution changes under covariate shift. Indeed, for the weight $w(\boldsymbol{x})$ the expectation function $f(\boldsymbol{x})$ of the probability density $P_{ta}(\boldsymbol{x})$ can be

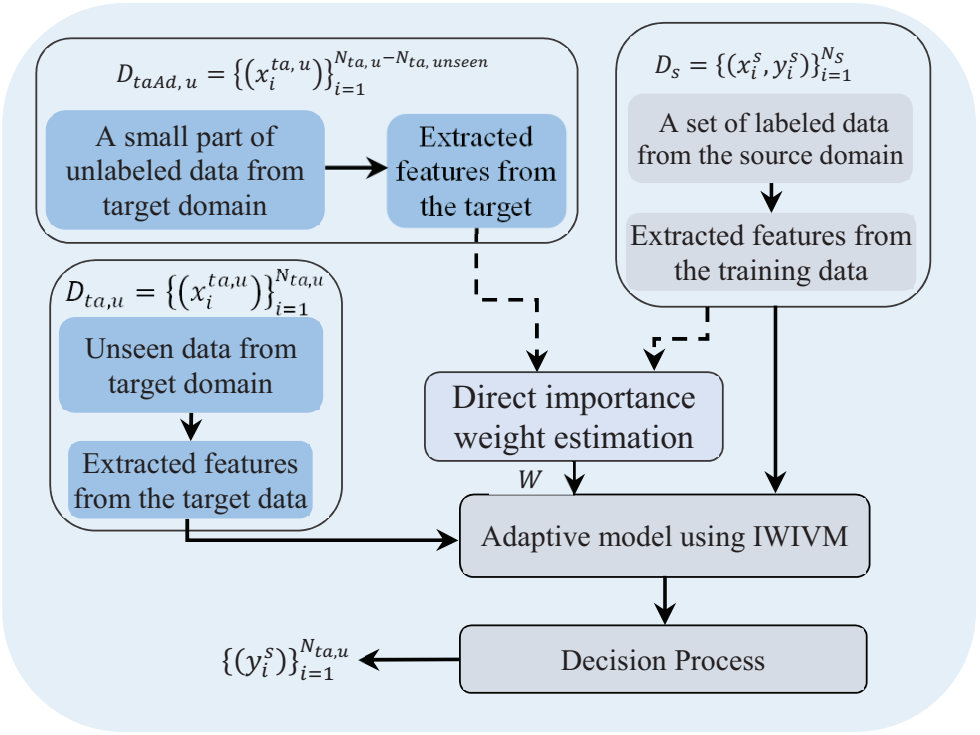

FIGURE 6.1: Structure of the importance weight import vector machine (IWIVM) for unsupervised domain adaptation.

computed as follows,

$$\begin{aligned} E_{P_{ta}(\boldsymbol{x})}[f(\boldsymbol{x})] &= \int f(\boldsymbol{x}) P_{ta}(\boldsymbol{x}) d\boldsymbol{x} \\ &= \int f(\boldsymbol{x}) w(\boldsymbol{x}) P_s(\boldsymbol{x}) d\boldsymbol{x} \\ &= E_{P_s(\boldsymbol{x})}[w(\boldsymbol{x}) f(\boldsymbol{x})] \end{aligned} \tag{6.10}$$

Generally, *importance weighted ERM* (IWERM) [31] results from applying instance weights, calculated by direct importance estimation methods, to ERM:

$$\boldsymbol{\theta}_{IWERM} = \underset{\boldsymbol{\theta}}{argmin} \left[\frac{1}{N_s} \sum_{i=1}^{N_s} \frac{P_{ta}(\boldsymbol{x}_i^s)}{P_s(\boldsymbol{x}_i^s)} loss(\boldsymbol{x}_n^s, y_n^s, \mathrm{f}(\boldsymbol{x}_i^s; \boldsymbol{\theta})) \right]. \tag{6.11}$$

IWERM has shown to be consistent even for misspecified models, and it satisfies

$$\lim_{N_s \to \infty} [\boldsymbol{\theta}_{IWERM}] = \boldsymbol{\theta}^*,$$

Algorithm 2: Weighted Newton Optimization Method for IWIVM

Input: WNLL represented by $H = W^{\delta} ln(1 + e^{-\boldsymbol{y}.(K_1\boldsymbol{\theta})}) + \frac{\lambda}{2}\boldsymbol{\theta}^{\top} K_{reg}\boldsymbol{\theta}$

Output: Optimal values of parameters $\boldsymbol{\theta}^*$

Initialize $\boldsymbol{\theta}_0, \lambda$ and $W = diag(w^{\delta}(\boldsymbol{x}_1), w^{\delta}(\boldsymbol{x}_2), \ldots, w^{\delta}(\boldsymbol{x}_{n_s}))$;

while $\boldsymbol{\theta}_t$ *does not converge* **do**

  for *each class label* $c \in D_s$ **do**

    Compute $\boldsymbol{y} = (y_1, \ldots, y_{n_s})^{\top}, \boldsymbol{p} = (p_1, \ldots, p_{n_s})^{\top}, R = diag(w_1^{\delta} v_1, .., w_{n_s}^{\delta} v_{n_s}), v_i = p_i(1 - p_i), p_i = \frac{e^{-y_i f(\boldsymbol{x}_i)}}{1+e^{-y_i f(\boldsymbol{x}_i)}}, \quad i = 1, \ldots, n_s;$

  Compute K_1, K_{reg}; Iteratively update $\boldsymbol{\theta}_t$

$$
\begin{aligned}
\boldsymbol{\theta}_t &= \boldsymbol{\theta}_{t-1} - (\frac{\partial^2 H}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^{\top}})^{-1}(\frac{\partial H}{\partial\boldsymbol{\theta}}) \\
&= \boldsymbol{\theta}_{t-1} + (K_1^{\top} R K_1 + \lambda K_{reg})^{-1}(-K_1^{\top} W(\boldsymbol{y}.\boldsymbol{p}) + \lambda K_{reg}\boldsymbol{\theta}_{t-1}) \\
&= (K_1^{\top} R K_1 + \lambda K_{reg})^{-1}(K_1^{\top} R\boldsymbol{z}) \\
\textit{where} \quad \boldsymbol{z} &= K_1\boldsymbol{\theta}_{t-1} + R^{-1}(\boldsymbol{y}.\boldsymbol{p})
\end{aligned}
$$

where $\boldsymbol{\theta}^*$ denotes the optimal parameter values. Following this idea, by applying instance weights to (6.8), the weighted negative log-likelihood (WNLL) function is obtained [31]:

$$\min_{f \in H_K} \frac{1}{N_s} \sum_{i=1}^{N_s} w\,(\boldsymbol{x}_i)^{\delta}\, ln[1 + e^{-y_i f(\boldsymbol{x}_i)}]_+ + \frac{\lambda}{2}\|f\|_{H_K}^2, \tag{6.12}$$

where $(0 \leq \delta \leq 1)$ denotes the flattening parameter, which controls the bias-variance trade-off in importance sampling [17]. In particular, whenever δ is close to 1 the bias gets smaller. However, the variance tends to be larger. The WNLL function is still convex and the unique minimizer can be obtained using the Newton optimization method. Modifying the notation of (6.12) for a finite dimension form, the WNLL becomes,

$$H = W^{\delta} ln(1 + e^{-\boldsymbol{y}.(K_1\boldsymbol{\theta})}) + \frac{\lambda}{2}\boldsymbol{\theta}^{\top} K_{reg}\boldsymbol{\theta} \tag{6.13}$$

Algorithm 3: Importance-Weighted Import Vector Machine

Input: $\varepsilon_{Forward} = 0.0001$, $\varepsilon_{Backward} = 0.0001$, $H_L = \infty$, $IV = \phi$, $D_s = \{(\boldsymbol{x}_i^s, y_i^s)\}_{i=1}^{N_s}$, $D_{ta,u} = \left\{\left(\boldsymbol{x}_i^{ta,u}\right)\right\}_{i=1}^{N_{ta,u}}$, $L = 1, \Delta L = 1$, λ, σ;

Output: Optimal subset of input vectors IV^*

Compute weights W using direct importance estimation methods:
$W = RuLSIF(D_s, D_{ta,u}, \sigma, \lambda)$;

while $\frac{|H_L - H_{L-\Delta L}|}{H_L} \leq \varepsilon_{Forward}$ **do**

  Forward selection (select an sample form D_s as an import vector if the objective function value can be decreased)

  for *each* $\boldsymbol{x}_l \in D_s$ **do**

    $f_l(\boldsymbol{x}) = \beta_0 + \sum_{\boldsymbol{x}_i \in IV \cup \{\boldsymbol{x}_l\}} \theta_i K(\boldsymbol{x}_i^s, \boldsymbol{x})$

    Use Newton optimization method (Algorithm 2) to find $\boldsymbol{\theta^*}$ that minimize

$$H(x_l) = \sum_{i=1}^{n} w(x_i)^{\delta} ln[1 + e^{-y_i f(x_i)}] + \frac{\lambda}{2}\|f_l(\boldsymbol{x})\|_{H_K}^2$$
$$= W^{\delta} ln[1 + e^{-y.(K_1^l \boldsymbol{\theta})}] + \frac{\lambda}{2}\boldsymbol{\theta}_{t-1}^{\top} K_{reg}^l \boldsymbol{\theta}_{t-1}$$

    where the regressor matrix $K_1^l = [\mathbf{1}, K(x_i, x_{i'})_{i,i'=1}^n]_{n \times (m+1)}$, $x_i \in D_s$, $x_{i'} \in IV \cup \{x_l\}$;

    and the regularization matrix $K_{reg}^l = \begin{pmatrix} 0 & 0 \\ 0 & K(x_i, x_{i'})_{i,i'=1}^n \end{pmatrix}_{(m+1)\times(m+1)}$,

    $x_i, x_{i'} \in IV \cup \{x_l\}$; and $m = |IV|$

  find the best point $x_l^* = argmin_{x_l \in D_s} H(x_l)$.

  Update $IV = IV \cup \{x_l^*\}, D_S = D_S \setminus \{x_l^*\}, H_L = H(x_l^*), K_{reg}, K_1$

  Backward deselection

  while *IV can not be reduced any more* **do**

    for *all IV* **do**

      Let $IV = IV \setminus \{x_u\}$;

      for *each* $\boldsymbol{x}_u \in IV$ **do**

        Use Newton optimization method (Algorithm 2) to find $\boldsymbol{\theta^*}$ that minimize

$$H(x_u) = \sum_{i=1}^{n} w(x_i)^{\delta} ln[1 + e^{-y_i f(x_i)}] + \frac{\lambda}{2}\|f_u(\boldsymbol{x})\|_{H_K}^2$$
$$= W^{\delta} ln[1 + e^{-y.(K_1^u \boldsymbol{\theta})}] + \frac{\lambda}{2}\boldsymbol{\theta}_{L-1}^{\top} K_{reg}^l \boldsymbol{\theta}_{L-1}$$

      find the best point $x_u^* = argmin_{x_u \in D_s} H(x_u)$.

      if $H_{x_u^*} < H_{Current}$ **then**

        Update $IV = IV \setminus \{x_u^*\}$;

  Update $D_s, L = L + 1$;

where

$$W = diag(w^{\delta}(\boldsymbol{x}_1), w^{\delta}(\boldsymbol{x}_2), \ldots, w^{\delta}(\boldsymbol{x}_{N_s})) \tag{6.14}$$

$$\boldsymbol{y} = (y_1, \ldots, y_{N_s})^{\top} \tag{6.15}$$

$$\boldsymbol{\theta} = (\beta_0, \theta_1, \ldots, \theta_{N_s})^{\top} \tag{6.16}$$

$$K_1 = (\mathbf{1}, K(x_i, x_{i'})_{i,i'=1}^{N_s}) \tag{6.17}$$

$$K_{reg} = \begin{pmatrix} 0 & 0 \\ 0 & K(x_i, x_{i'})_{i,i'=1}^{N_s} \end{pmatrix}_{(N_s+1)\times(N_s+1)} \tag{6.18}$$

and "." in (6.13) denotes element-wise multiplication. To find $\boldsymbol{\theta}$ in (6.13) a derivative of H with respect to $\boldsymbol{\theta} = 0$ is computed. Then, as shown in Algorithm 2, a weighted Newton optimization (WNO) method is calculated. In this WNO, the vector $\boldsymbol{\theta}$ is updated by iteratively embedding the instance weights as,

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - (\frac{\partial^2 H}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{\top}})^{-1} (\frac{\partial H}{\partial \boldsymbol{\theta}}) \tag{6.19}$$

where

$$\frac{\partial H}{\partial \boldsymbol{\theta}} = -K_1^{\top} W(\boldsymbol{y}.\boldsymbol{p}) + \lambda K_{reg} \boldsymbol{\theta} \tag{6.20}$$

$$\frac{\partial^2 H}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{\top}} = K_1^{\top} R K_1 + \lambda K_{reg} \tag{6.21}$$

and

$$\boldsymbol{p} = (p_1, \ldots, p_{n_s})^{\top} \tag{6.22}$$

$$p_i = \frac{e^{-y_i f(\boldsymbol{x}_i)}}{1 + e^{-y_i f(\boldsymbol{x}_i)}}, \ i = 1, \ldots, n_s \tag{6.23}$$

$$R = diag(w_1^{\delta} v_1, .., w_{n_s}^{\delta} v_{n_s}) \tag{6.24}$$

$$v_i = p_i(1 - p_i) \tag{6.25}$$

To reduce the computational cost of KLR, following the idea of import vector machine, IWIVM also attempts to find a submodel that approximates the full model given by KLR. Indeed, as

summarized in Algorithm 3, which performs the sparse KLR by a subset selection. The sub-model has the form:

$$f_{\theta_n}(\boldsymbol{x}) = \beta_0 + \sum_{i \in IV} \theta_{n,i} \; K\left(\boldsymbol{x}, \boldsymbol{x}_i^s\right) \tag{6.26}$$

where IV is the subset of the import points, which is the best approximation of KLR model. Based on the idea of Zhu et al. [261], these subsets are determined in a greedy forward search, where it starts with an initially empty set $IV = \varnothing$, then in each iteration one instance is chosen to append to the set until a convergence criteria H_L is satisfied. Rochester et al. [265] proposed a hybrid forward/backward strategy, which successively adds import vectors to the set, but also tests if import vectors can be removed in each step. Since IWIVM starts with an empty import vector set and only add import vectors sequentially in the first iterations, the decision boundary can be very different from its final position. A greedy forward selection is unable to remove inappropriate import vectors. However, a removal of import points can lead to a sparser and more accurate solution than only using forward selection steps. To prevent infinite loops in the process of import vector selection, the deselected import vectors are excluded from selection for the next few iterations.

As shown in Algorithm 3, the convergence criterion is to look at the regularized WNLL. Following Zue et al. [261], if the ratio $\frac{|H_L - H_{L-\Delta L}|}{H_L}$ is less than a pre-chosen small number *e.g.* $\epsilon = 0.0001$, the algorithm stops adding new import points to IV. This threshold influences the sparsity of the model.

6.5 Reliable-IWCV for Model Selection and Parameters Tuning in Direct Importance Estimation and IWIVM

The performance of IWIVM and direct importance estimation methods depend on the choice of basis kernel functions and parameters. In fact, model selection is affected by values of parameters such as Gaussian kernel width σ, regularization parameter λ and flatting parameter δ, which are the most effective performance factors.

Normally, model selection and parameter tuning of IVM and direct importance estimation methods are straightforward by cross validation (CV) over the performance of subsequent learning algorithms. IVM as well as some of the importance ratio estimation methods such as KLIEP and KMM use ordinary CV for tuning the parameters. However, under covariate shift, tuning the parameters using ordinary CV is highly unreliable. In fact, these values are highly

Algorithm 4: Reliable Importance Weighted Cross Validation (RIWCV)

Input: A set of parameters $\boldsymbol{\Psi} = (\Psi_1, \ldots, \Psi_M)^\top$ for grid search, where $\Psi_1 = (\sigma_0, \sigma_1, \ldots, \sigma_k), \Psi_2 = (\lambda_0, \lambda_1, \ldots, \lambda_r), \ldots, \Psi_M = (\delta_0, \delta_1, \ldots, \delta_g)$;
Output: Γ = A vector of the most reliable values of the parameters.

for $\boldsymbol{\Psi}$ **do**
  for $k - Fold\ Cross\ Validation$ **do**
    $G_{err(kIWCV)} = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{|D_{s_i}|}\sum_{(\boldsymbol{x},y)\in D_{s_i}}\frac{P_{ta}(\boldsymbol{x})}{P_s(\boldsymbol{x})} loss\left(\boldsymbol{x}, y, f_{D_{s_i}}(\boldsymbol{x})\right)$
  Update $\left[G_{err(kIWCV)}\right]_{\boldsymbol{\Psi}}$
for $\Psi_i = \Psi_1 : \Psi_M$ **do**
  $\mathbf{AVG} = \left[G_{err(kIWCV)}\right]_{\boldsymbol{\Psi}}$
  for $\boldsymbol{\Omega} = \boldsymbol{\Psi} - \Psi_i$ **do**
    $\mathbf{AVG} = mean(\mathbf{AVG}, on\ Dimension(\boldsymbol{\Omega}\ mod\ \Psi_M))$;
  $\Gamma_{\Psi_i} = argmin(\mathbf{AVG})$;

biased under covariate shift. Therefore, the CV procedure itself needs to be weighted. To cope with this problem, Sugiyama et al. [97] proposed a variant of CV called importance-weighted CV (IWCV), which is almost unbiased under covariate shift. To estimate the generalization error G_{err}, the K-fold IWCV (kIWCV) is defined as follows:

$$G_{err(kIWCV)} = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{|D_{s_i}|}\sum_{(\boldsymbol{x},y)\in D_{s_i}}\frac{P_{ta}(\boldsymbol{x})}{P_s(\boldsymbol{x})} loss\left(\boldsymbol{x}, y, f_{D_{s_i}}\right) \tag{6.27}$$

where $D_s = \{(\boldsymbol{x}_i^s, y_i^s)\}_{i=1}^{N_s}$ will randomly be divided in k disjoint nonempty same-size subsets $\{D_{s_i}\}_{i=1}^{k}$. In case of $k = N_s$, $kIWCV$ is changed to Leave-one-out importance weight cross validation ($LOOIWCV$):

$$G_{err(LOOIWCV)} = \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{P_{ta}(\boldsymbol{x}_i^s)}{P_s(\boldsymbol{x}_i^s)} loss\left(\boldsymbol{x}_n^s, y_n^s, f_i(\boldsymbol{x}_i^s)\right) \tag{6.28}$$

where $f_i(\boldsymbol{x}_i^s)$ is a function learned from all samples except sample $(\boldsymbol{x}_i^s, y_i^s)$. The model validation techniques of CV and IWCV, attempt to find the best parameters and models based on a greedy strategy. In fact, these techniques provide the values of the parameters that correspond to the best result of cross validation. Due to the distributions difference of source and target domains, the parameters' values obtained by CV and even IWCV methods may not be the best selection for the target domain. Therefore, CV and IWCV may fall down into local minima for the target domain and the built models based on these parameters are inaccurate. To alleviate this possibility, a heuristic method is proposed, henceforth named reliable IWCV (RIWCV). This method attempts to find a more reliable combination of parameters, instead of

the best parameters combination (like in [31]). In fact, ordinary CV and IWCV will return the values of parameters that correspond to the index of an element with the minimum value in the generalization error matrix (*e.g.* $\left[G_{err(kIWCV)}\right]_{\boldsymbol{\sigma},\boldsymbol{\lambda},\boldsymbol{\delta}}$). However, to find the most reliable value for each parameter, RIWCV method decreases the effect of other parameters in the selection process. As shown in Algorithm 4, firstly, the matrix $\left[G_{err(kIWCV)}\right]_{\boldsymbol{\psi}}$ is calculated for different values of the parameters based on K-fold IWCV strategy. Then, for each parameter from the set of parameters $\boldsymbol{\Psi}$, an average vector with the length of the parameter Ψ_i is calculated. The average vector is obtained in an iterative way, where in each iteration the matrix AVG is updated (by loop 5-9 in Algorithm 4). In order to present additional explanation of how RIWCV works, an example of $[G_{err}]_{\boldsymbol{\sigma},\boldsymbol{\lambda},\boldsymbol{\delta}}$ matrix with three parameter $\boldsymbol{\sigma},\boldsymbol{\lambda},\boldsymbol{\delta}$ is presented. To find the most reliable value for $\boldsymbol{\sigma}$ from the generalized error matrix,

$$[G_{err}]_{\boldsymbol{\sigma},\boldsymbol{\lambda},\boldsymbol{\delta}} = \begin{bmatrix} e_{\sigma_1,\lambda_1,\delta_1} & \dots & e_{\sigma_k,\lambda_r,\delta_1} \\ \dots & \dots & \dots \\ e_{\sigma_k,\lambda_1,\delta_1} & \dots & e_{\sigma_k,\lambda_r,\delta_1} \end{bmatrix} \quad \dots \quad \begin{bmatrix} e_{\sigma_1,\lambda_1,\delta_g} & \dots & e_{\sigma_k,\lambda_r,\delta_g} \\ \dots & \dots & \dots \\ e_{\sigma_k,\lambda_1,\delta_g} & \dots & e_{\sigma_k,\lambda_r,\delta_g} \end{bmatrix}, \tag{6.29}$$

First, the averages on the other dimensions $\boldsymbol{\lambda},\boldsymbol{\delta}$ are calculated, where for each dimension the averages are calculated on the previous matrix, *i.e.*

$$M_\lambda = Avg\left(G_{err},\lambda\right) = \begin{bmatrix} \left(e_{\sigma_1,\lambda_M,\delta_1}\right) \\ \dots \\ \left(e_{\sigma_k,\lambda_M,\delta_1}\right) \end{bmatrix} \quad \dots \quad \begin{bmatrix} \left(e_{\sigma_1,\lambda_M,\delta_g}\right) \\ \dots \\ \left(e_{\sigma_k,\lambda_M,\delta_g}\right) \end{bmatrix} \tag{6.30}$$

then,

$$M_\delta = Avg\left(M_\lambda,\delta\right) = \begin{bmatrix} \left(e_{\sigma_1,\lambda_M,\delta_M}\right) \\ \dots \\ \left(e_{\sigma_k,\lambda_M,\delta_M}\right) \end{bmatrix} \tag{6.31}$$

In the resulted vector M_δ, the index of the best value (element with a minimum average error) corresponds to the most reliable value of $\boldsymbol{\sigma}$. The same method can be applied to find the most reliable values for the other parameters. This method is extendable to a higher number of parameters.

6.6 Unsupervised Domain Adaptation for Automatic Sleep Staging

Aiming to improve the applicability of automatic sleep staging, an adaptive ASSC approach has been developed based on unsupervised covartiate shift adaptation. The main goal of this method, which has two applications of sleep/awake detection and multiclass sleep stage classification, is to cope with the variations between the training set and new subjects.

Similar to SSM4S method (Section 5.2), after applying common preprocessing (a notch filter at 50 Hz, band-pass Butterworth filter with lower cutoff of 0.3 Hz and higher cutoff of 35 Hz), and segmentation of the signals in 30s epochs, some features are extracted using several methods in the temporal, frequency and time-frequency domains (see details in Section 5.2.2). The PSG signals are traditionally analyzed in the frequency domain, since each sleep stage is characterized by a specific pattern of frequency contents. Moreover, PSG signals are non-stationary; therefore time-frequency transformations like wavelets are very useful. Due to superiority of the maximal overlap discrete wavelet transform (MODWT) [20, 266] versus discrete wavelet transform, a MODWT of depth 6 with Daubechies order four (db4) is applied to every 30s epochs with a sampling rate of 200 Hz. The frequency ranges are broken down into δ range ($<$ 4 Hz), θ range (4-8 Hz), α range (8-13 Hz) and β range (13-30 Hz). To represent the time-frequency distribution of the EEG, EOG and EMG signals, features such as energy, percent of energy [18], mean and standard deviation are extracted from each sub-band. Furthermore, due to the importance of spectral and temporal analysis, features such as relative spectral power, peak to peak amplitude of two EOGs, Tsallis ($q = 2$), Renyi ($\alpha = 2$), Shannon entropy, Hjorth parameters, harmonic parameters, percentile 25, 50, 75, autoregressive coefficients (order 3), slow wave index (SWI), Kurtosis and Skewness [5] are extracted from 6 EEG and 2 EOG and 1 EMG channels.

To reduce the influence of extreme values, a transformation is applied to the matrix of features $\mathbf{Y}$, as follows:

$$\mathbf{X} = arcsin(\sqrt{\mathbf{Y}}) \tag{6.32}$$

where

$$\mathbf{X} = \{x_{ij}\}, \quad i = 1, 2, \ldots, N \; and \; j = 1, 2, \ldots, M \tag{6.33}$$

is the transformed feature matrix, where N and M denote the number of subjects and the number of features, respectively. Then, to avoid features in greater numeric ranges dominating

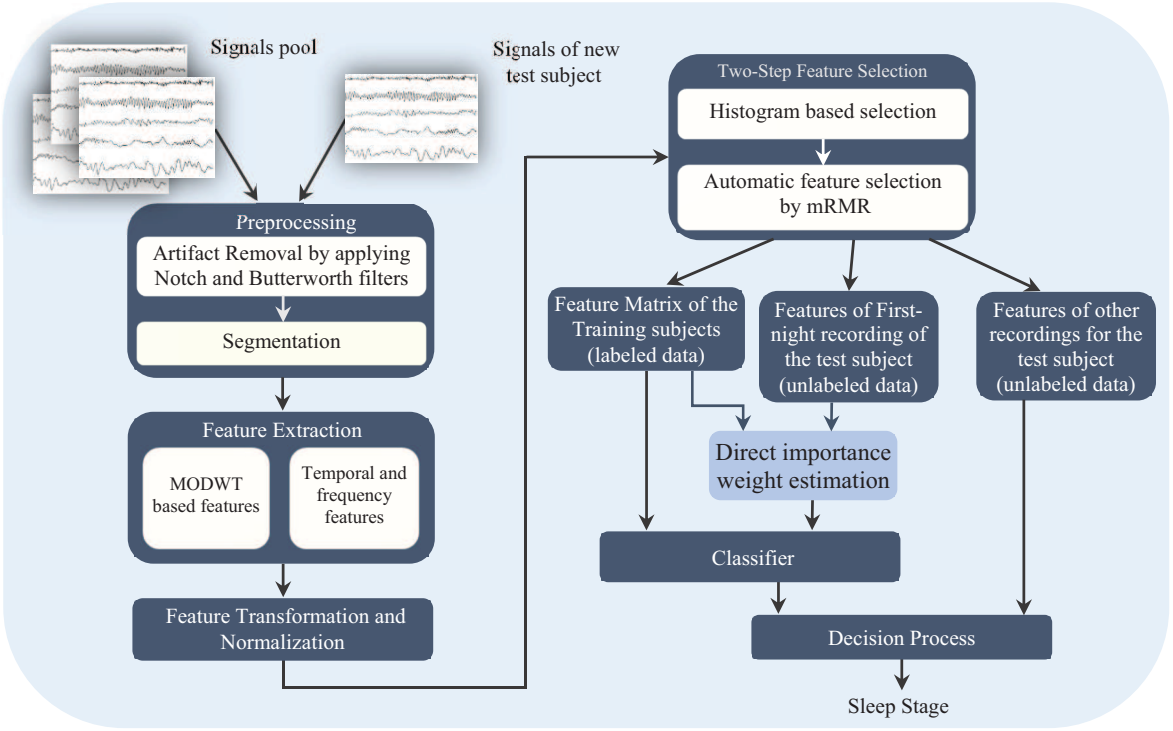

FIGURE 6.2: Structure of the adaptive automatic sleep stage classification method.

those in smaller numeric ranges, as well as numerical difficulties during classification, each feature of the transformed matrix $\mathbf{X}$ is independently normalized to the [0,1] range by applying

$$\overline{x}_{ij} = x_{ij}/(max(\boldsymbol{x}_j) - min(\boldsymbol{x}_j)), \tag{6.34}$$

where, $i = 1, 2, \ldots, N$, $j = 1, 2, \ldots, M$ and $\boldsymbol{x}_j$ is a vector of each independent feature. Next, a two-step feature selection process that consists on a filtering and a wrapper phases is performed: firstly, as detailed in [20], the less discriminative feature-types are removed and features such as relative spectral power, harmonic parameters, percentile 75, autoregressive coefficients (order 3), kurtosis and skewness [5] are selected. Then, in the second step, to select the best elements of each feature-type, the aforementioned feature vector is fed into a mRMR feature selector. As depicted in Fig. 6.2, to handle the adaptive classification, the unsupervised strategy IWIVM for covariate shift adaptation is applied. In this strategy, a limited number of unlabeled data from the testing subjects is used to calculate the importance ratio (weights).

Chapter 7

Experiments

This chapter lays out the results of the proposed methods. We evaluate the methods in the context of automatic sleep stage classification, a synthetic toy problem, and cross-domain object recognition. The chapter is organized into five sections. The general experimental setups for each problem are provided in Section 7.2, including an introduction to the benchmark datasets, the feature representation of data and the setting up the evaluations. In Section 7.2.1, a review on the current sleep datasets followed by the details of ISRUC-Sleep, a publicly-available comprehensive sleep dataset, is provided. In order to analyses the proposed unsupervised domain adaptation method IWIVM, a synthetic toy problem (Section 7.2.2), and a real-world cross-domain object recognition (Section 7.2.3) are employed. Section 7.3 summarize several experiments for the performance assessment of the SSM4S method employing ISRUC-Sleep dataset. In addition, comparison of baselines and other competing domain adaptation methods are provided in Section 7.5. We also report the results of applying the IWIVM to the real-world sleep staging problem. The results of the adaptive ASSC method are presented in Section 7.6.

7.1 Introduction

To evaluate the performance of the proposed methods, different kinds of problems are considered: 1) Sleep stage classification with two applications sleep-wake classification and multiclass sleep staging; 2) A synthetic binary toy problem; and 3) A cross-domain object recognition task. The first problem is approached by applying the *SSM4S* and the adaptive sleep staging methods on a comprehensive ISRUC-Sleep dataset [27]. The second and third problems

are approached by applying the importance weighting import vector machine, an unsupervised covariate shift adaptation method, on a Toy and Office-Caltech-256 [175, 267] datasets, respectively.

7.2 Experimental Setup

This section describes the experimental setup, including benchmark datasets, learning tasks, and the feature representations of data. In particular, we will describe, i) ISRUC-Sleep dataset, which is a comprehensive sleep dataset for sleep studies; ii) a synthetic binary dataset; and iii) two object recognition dataset office and caltech-256.

7.2.1 Sleep Dataset

To assess the efficiency of sleep pattern analysis methods, each research team collects their own test data with expenditure of time and/or financial resources [268, 269]. These datasets, mainly used in the context of their own research, often lack several relevant information details regarding acquisition and subject pathological conditions (neural, cardiorespiratory, medication effects). Some of these datasets [270, 271] also lack statistical significance and just recorded some of the PSG channels. Therefore, an accurate and comparative evaluation of the performances of these methods with new methods cannot be done effectively.

Recognizing the need and usefulness of publicly available sleep datasets, which can be used as a common reference for researchers, some sleep-related datasets are developed by sleep research groups. As shown in Table 7.1, these datasets contain multiple signals from some healthy and patient subjects.

The sleep datasets of PhysioBank [272] have been used in a few works (see Table 7.1). Even though *MIT-BIH*, *Sleep-EDF* and *Extended Sleep-EDF* are general purpose datasets, they do not have enough subjects for generalization purposes. *CAP-Sleep* dataset is an exception in PhysioBank repository, containing 108 recordings, however, it consists of the specific data useful for studies related to the cyclic alternating pattern (CAP).

The *sleep heart health study (SHHS)* dataset [273], which has a convenient number of recordings, is not a completely public dataset. It is available only upon special request and approval. On the other hand, due to providing just the signals of two EEG (C3-A2 and C4-A1) channels, *SHHS* has limitations for general-purpose sleep research. In fact, it is a specific purpose

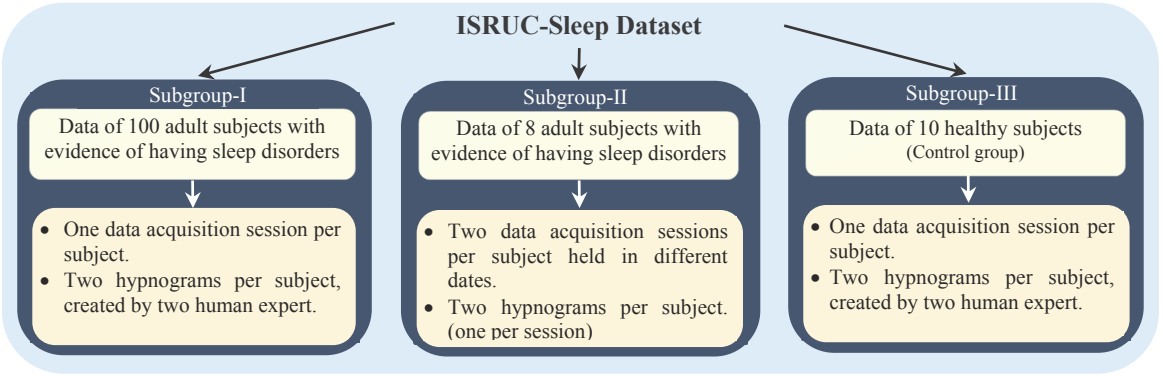

FIGURE 7.1: Details of ISRUC-Sleep sataset.

dataset useful in research studies involving relationships between sleep-disordered breathing and heart diseases.

Recently, *Montreal archive of sleep study (MASS)*, which is an open-access sleep dataset collected from healthy subjects, was proposed by O'Reilly et al. [10]. Although it is reported that the dataset contains data of 200 participants, it is a collection of five different subgroup of data. These subgroups were pooled from 8 different research protocols performed in 3 different hospital-based sleep laboratories. Furthermore, there exists some access restrictions regarding different kinds of information of the dataset. As detailed in Table 7.1, the subgroups of this dataset have significant differences in terms of number of channels, filtering methods applied to the signals, acquisition software, annotations, scoring criteria and epoch size.

In summary, all the dataset detailed in Table 7.1 have limitations in some aspects and as far as we know except Sleep-EDF dataset(expanded), which were recorded during years 1987-1991 in two subsequent day-night periods at the subjects' homes, only one acquisition session (one recording) per subject is available for the other datasets.

Therefore, we introduce a publicly-available comprehensive sleepdataset, called ISRUC-Sleep, which comprises three subgroups as illustrated in Fig. 7.1. The subgroups of the dataset contain PSG signals of different adult individuals, including healthy subjects, subjects with sleep disorders, and subjects under the effect of sleep medication. Sleep stages were labeled by two sleep experts. Furthermore, for 8 subjects (subgroup-II), two sets of PSG data, which have been recorded at different time dates, are provided.

Table 7.1: Some of the public sleep datates

Dataset	Subjects/ Sampling rate	Recorded channels	Recording duration	Purpose of creating the dataset	Subjects age	Literatures cited dataset
MIT-BIH [272]	18 Recording of 16 subjects with or without sleep apnea syndrome(SAS) / 250 Hz	Four-, six-, and seven-channel recordings of ECG signal, an invasive blood pressure signal, an EEG signal, a respiration signal and a text. Some records contain other signals such as, respiratory effort signal., an EOG signal, an EMG (from the chin) signal, a stroke volume signal	8-10 hours	General purpose	32–56 Avg.=43	Adnane et.al [23], Nicolaou and Georgiou [274], Fraiwan et.al [21]
Sleep-EDF [272]	8 subjects without any sleep-related medication, scored based on R&K / 100 Hz	EEG (Fpz-Cz and Pz-Oz), Horizontal EOG, submental-EMG envelope, oro-nasal airflow, rectal body temperature and an event marker	1.25 to 6.5 hours	General purpose	21-35	Bajaj and Pachori [11], Ronzhina et al. [161]
Expanded Sleep-EDF [272]	61 recordings from healthy subjects, without any sleep-related medication/ 100 Hz	EEG(Fpz-Cz and Pz-Oz), Horizontal EOG, submental chin EMG, and an event marker.	Around 9 hours	Study of age effects on sleep; study of temazepam effect on sleep	25-101	Kemp et. al [275], Yaghouby et.al [276]
CAP-Sleep [272]	108 recordings scored based on R&K / 512 Hz	3 EEG channels (F3 or F4, C3 or C4 and O1 or O2, referred to A1 or A2), 2 EOG channels, EMG of the submental is muscle, bilateral anterior tibial EMG, respiration signals (airflow, abdominal and thoracic effort and SaO2) and ECG.	8-10 hours	Study of the cyclic alternating pattern (CAP)	30-75	Terzano et. al [277]
SHHS-1, -2 [273]	9736 recordings scored based on R&K / 125 Hz	2 EEG channels (C3 or C4, referred to A1 or A2), 2 EOG channels, EMG of the submental, bilateral anterior tibial EMG, respiration signals (airflow, abdominal and thoracic effort) and ECG.	Overnight	Study of OSA, sleep-disordered breathing, and heart diseases	40 and older	Ebrahimi et. al [278]
MASS [10]	Collection of 200 recordings (5 different subsets) scored based on R&K or AASM / 256 Hz	4-20 EEG channels, 2-4 EOG channels, 1-5 EMG of the submental, bilateral anterior tibial EMG, sometimes with respiration signals (airflow, abdominal and thoracic effort and SaO2) and ECG.	Overnight	Study of the sleep spindles and general purpose	18-76	Tsanas et. al [279]

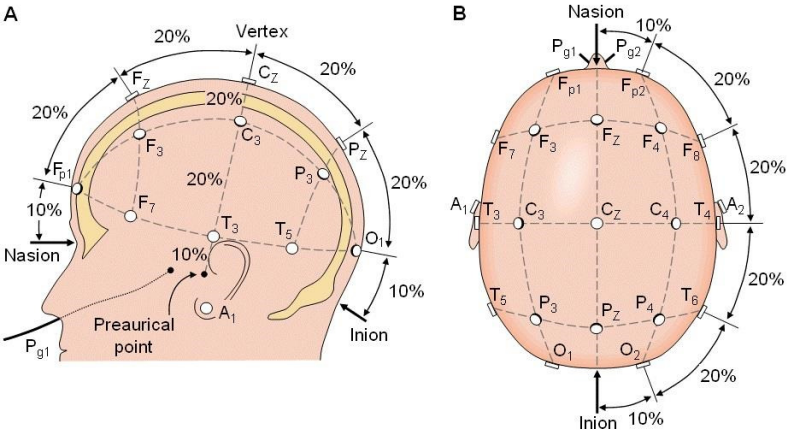

FIGURE 7.2: The international 10-20 system seen from (A) left and (B) above the head. A = Ear lobe, C = central, Pg = nasopharyngeal, P = parietal, F = frontal, Fp = frontal polar, O = occipital [2].

TABLE 7.2: Details of recorded signals of ISRUC-Sleep dataset

Channel Number	Type of the signal	Label	Frequency rate/Hz	Butterworth	Notch filter	Description
1	EOG	LOC-A2	200	0.3Hz-35Hz	50Hz	left eyes movements
2		ROC-A1				Right eyes movements
3	EEG	F3-A2	200	0.3Hz-35Hz	50Hz	Brain channels with the references A1 and A2, which placed in the left and right ear-lobes.
4		C3-A2				
5		O1-A2				
6		F4-A1				
7		C4-A1				
8		O2-A1				
9	Chin EMG	X1	200	10Hz-70Hz	50Hz	chin EMG, placed between the chin and the lower lip.
10	ECG(EKG)	X2	200		50Hz	electrocardiographic.
11	Leg-1 EMG	X3	200	10Hz-70Hz	50Hz	left leg movement.
12	Leg-2 EMG	X4				right leg movement.
13	Snore	X5	200	10Hz-70Hz	50Hz	snore (derived).
14	Flow-1	X6	12.5			airflow (pressure based).
15	Flow-2	DC3	25			
16	abdominal	X7	25			abdominal efforts.
17		X8				
18	Pulse oximetry	SaO2	12.5			pulse oximetry (SaO2).
19	Body position	DC8	25			body position (BPOS).

7.2.1.1 ISRUC-Sleep Dataset

ISRUC-Sleep dataset contains data collected from all-night PSG recordings with duration around eight hours. Each recording was randomly selected between PSG recordings that were acquired by the Sleep Medicine Centre of the Hospital of Coimbra University (CHUC), in the period 2009-2015. Overall standard setup setting for data acquisition, comprised a biosignal acquisition equipment (a SomnoStar Pro sleep system which is a multi-channel ambulatory recording device), and a set of sensors collecting data in a non-invasive way, according to

the international 10-20 standard electrode placement (Fig. 7.2) [280]. With regard to the arrangements, the subject sleeps in a bed in a patient's room, and the experts and technicians stay in a separate room. All patients referred were submitted to an initial briefing with the support of an informed consent document. The ethics committee of CHUC approved the use of the data of the referred patients as anonymous for the research purposes.

The PSG signals were recorded according to the recommendations of the AASM manual. As described in Table 7.2, each recording consists of signals from 19 channels. All EEG, EOG, and chin EMG signals were sampled at 200 Hz and stored using the standard EDF+ data formats with .REC extension [281]. All recordings of the dataset were segmented into epochs of 30s and visually scored by two different sleep experts in CHUC according to the guidelines of AASM [114], with the stages: awake, NREM (N1, N2, and N3) and REM sleep. Calculating Cohen's kappa index between two experts over the subjects of the subgroups yields to the following Kappa indexes: overall kappa index of 0.87±0.09 for subgroup-I, 0.82±0.15 for subgroup-II, and 0.9±0.06 for subgroup-III [1]. The labels are stored in standard text file format, where each line corresponds to one epoch; Moreover, the gender, height, weight, age and date of recording, of individuals tested are recorded in the header of each text.

Further analysis such as sleep events, sleep related disorders, other diseases, sleep pathology, used medications, EEG pattern alterations, and percentage of each sleep stage for each subject are presented.

ISRUC-Sleep dataset comprises three subgroups of data[2] as described in Table 7.3.

7.2.2 Binary Toy Problem

To illustrate the behavior of the proposed unsupervised domain adaptation method IWIVM, we consider a two-dimensional toy problem for binary classification under covariate shift. The toy dataset is generated using the following characteristics:

A set of training instances $\{\boldsymbol{x}_n^s\}_{n=1}^{N_s}$ and test instances $\left\{\boldsymbol{x}_n^{ta}\right\}_{n=1}^{N_{ta}}$ are created with the input densities $P_s(\boldsymbol{x})$ and $P_{ta}(\boldsymbol{x})$, respectively.

$$P_s(\boldsymbol{x}) = \frac{1}{2}\ N\left(\boldsymbol{x}; \begin{bmatrix} -2 \\ 3 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}\right) + \frac{1}{2}\ N\left(\boldsymbol{x}; \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}\right),$$

[1] Distribution of the individual kappa indexes over the recordings are presented in the result files, which are available via http://sleeptight.isr.uc.pt/ISRUC_Sleep/.

[2] Recordings, summary of the characteristics, clinical information, and the overall performance of the methods for each subject of the subgroups are available via http://sleeptight.isr.uc.pt/ISRUC_Sleep/, The dataset will be turn available for researchers in the dedicated website.

TABLE 7.3: Characteristics of ISRUC-Sleep Dataset

Dataset	Subjects	Number of recording per subject	Subject characteristics	Subjects age
Subgroup-I	100 Subjects (55 male, 45 female) with evidence of having sleep disorders	one data acquisition session per subject	most of the subjects have detected sleep apnea events; the subjects could be under medication, but all were in position to breathe without the help of machine.	20-85, Avg.=51, std.=16 year
Subgroup-II	8 Subjects (6 male, 2 female) with evidence of having sleep disorders	two data acquisition sessions were performed in two different dates.	detected sleep apnea events; the subjects could be under medication, but all were in position to breathe without the help of machine.	26-79, Avg.=46.87, std.=18.7 year
Subgroup-III	10 Subjects (9 male, 1 female)	one data acquisition session per subject	Healthy subjects (control group).	30-58, Avg.=40, std.=10 year

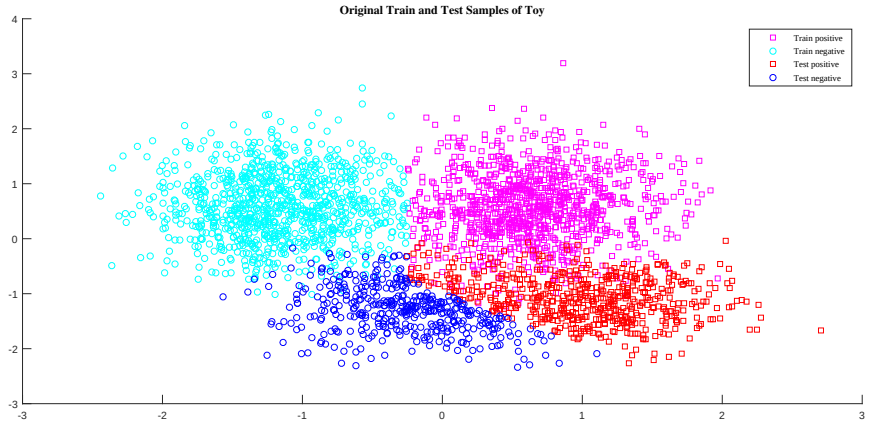

FIGURE 7.3: Original training and test samples of the Toy problem.

$$P_{ta}(\boldsymbol{x}) = \frac{1}{2}\ N\left(\boldsymbol{x}; \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) + \frac{1}{2}\ N\left(\boldsymbol{x}; \begin{bmatrix} 3 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right),$$

where $N(\boldsymbol{x}; \mu, \sigma^2)$ denotes the multivariate Gaussian density with mean μ and variance σ^2. The corresponding training labels $\{y_n^s\}_{n=1}^{N_s}$ and test labels $\{y_n^{ta}\}_{n=1}^{N_{ta}}$ follow posterior probabilities $P(y \mid \boldsymbol{x}_n^s)$ and $P\left(y \mid \boldsymbol{x}_n^{ta}\right)$, respectively. The class posterior probabilities are defined as follows,

$$P(y = +1|\boldsymbol{x}) = \frac{1}{2} + \frac{1}{2}\tanh\left(x^{(1)} + min\left(0, x^{(2)}\right)\right),$$

$$P(y = -1|\boldsymbol{x}) = 1 - P(y = +1|\boldsymbol{x}).$$

Training and test input samples are normalized in the element-wise manner so that, each element has mean zero ($\mu = 0$) and unit variance ($\sigma = 1$). As shown in Fig. 7.3, the generated training instances are distributed in the upper half of the graph and test instances are distributed in the lower half. The optimal decision boundary is the set of all $\boldsymbol{x}$ such that

$$(y = +1 \mid \boldsymbol{x}) = p\,(y = -1 \mid \boldsymbol{x}) = \frac{1}{2}.$$

Using the decision boundary that is naively estimated from training data, the test data may not be classified correctly. This is a typical example of covariate shift.
We randomly select 200 instance of the test set for the *importance ratio* calculation. Number of test instances are set to $N_{ta} = 1000$ and the experiments are repeated 50 times with different random seeds.

7.2.3 Cross-Domain Object Recognition Problem

To evaluate the IWIVM, proposed in Section 6.4, we employ two widely used datasets, Office [175] and Caltech-256 [267], in the context of cross-domain object recognition task. Office dataset consist of 4106 images in total, with 31 different object categories collected from three different sources:

1. Amazon (images downloaded from web, typically show objects only from a canonical viewpoint);
2. DSLR camera (images captured by a digital SLR camera on average with 3 images taken from different viewpoints);
3. Webcam (webcam images with low resolutions that show significant noise and color as well as white balance artifacts.

Each of these sources is regarded as a distinct domain. The Caltech-256 dataset contains 256 object categories from a single domain collected from the internet using Google. The 10 object categories that are in common to all four domains were used in the experiments.
The common categories to all four datasets are include: BACKPACK, TOURING-BIKE, CALCULATOR, HEADPHONES, COMPUTER-KEYBOARD, LAPTOP-101, COMPUTER-MONITOR, COMPUTER-MOUSE, COFFEE-MUG, and VIDEO-PROJECTOR. For each domain (Amazon, DSLR, Webcam, Caltech), the number of samples per category range from 8

FIGURE 7.4: Instance images from the monitor category in Caltech-256, Amazon, DSLR, and Webcam domains. The images of Caltech and Amazon are downloaded from the web, while DSLR and Webcam images are captured by high resolution DSLR camera and low resolution webcam camera, respectively.

to 151. Figure 7.4 illustrates the differences among these domains with some example images from each domain. The covariate shift is caused by several factors including lighting variations and changes in resolution, pose and background.

To better compare with the state-of-the-art reported performances, we follow the same settings as described in [175] for analyzing the considered datasets. Specifically, SURF-BoW features are provided for all the images [172]. The SURF-BoW features are quantized into 800-bin histogram by using k-means. Then, the histograms are standardized by z-score to have zero mean and unit standard deviation in each dimension. We randomly select 20 training images per category for the source domain amazon and 8 for all other source domains. For semi-supervised setting, in each target domain, 3 labeled images per category are randomly selected for training, and the remaining images are used for testing.

For the experiments, different domains Amazon (A), Caltech-256 (C), DSLR (D), and Webcam (W), were selected as the source and the target domain. We apply PCA to the source and target data, and use different 10-dimensional features in all the experiments.

7.3 Performance Assessment of SSM4S Using ISRUC-Sleep

The performance of the proposed SSM4S method was assessed using the subjects of different subgroups (Fig. 7.1) of ISRUC-Sleep dataset detailed in Section 7.2.1.1. Two types of experiments have been carried out: sleep-wake detection and multiclass sleep staging. In order to verify reliability of the results, the assessments were determined by using five-fold, ten-fold, and leave-one subject-out cross-validation (LOOCV) strategy. In our experiments, a fourth order Daubechies with MODWT decomposition was adopted. Also mRMR algorithm [256] and Libsvm toolbox [282] with sigmoid kernel were used in the second phase of feature selector and classification phases, respectively. The sigmoid degree and C parameter of SVM were set

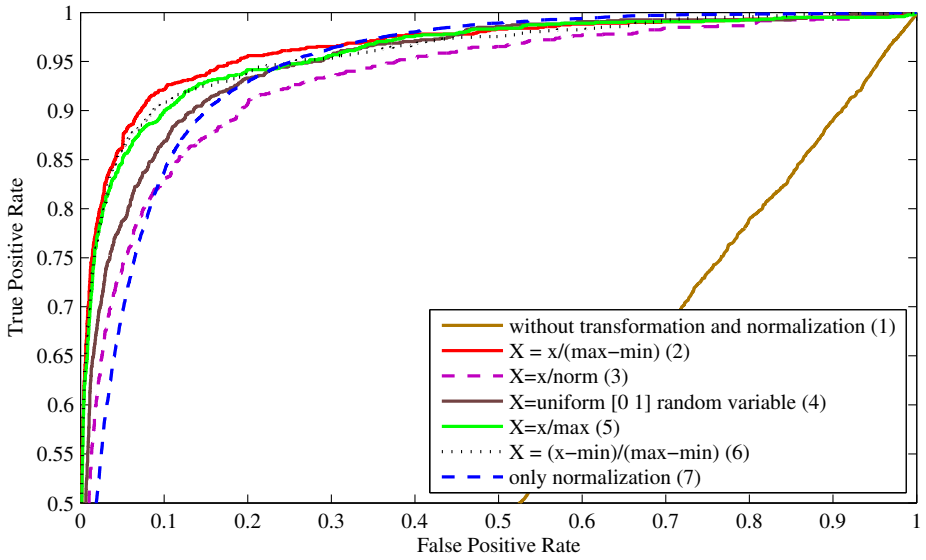

FIGURE 7.5: ROC curves corresponding to (1) without any transformation and any normalization; (2) with normalization $x_{ij} = x_{ij}/(max(x_j) - min(x_j))$; (3) with normalization from (2) over the transformed features by $x_{ij} = x_{ij}/(max(x_j) - min(x_j))$, (4) and (5) normalizations $x_{ij} = x_{ij}/(max(x_j) - min(x_j))$ and x = uniform [0 1] random variable with the same transformation of (3).

to 0.13 and 1.25 respectively, as they produced the best empirical results.

In order to characterize the performance of the method some well-known measures such as accuracy(ACC), receiver operating characteristic (ROC), balanced error rate (BER), sensitivity(SENS), specificity (SPEC), balanced correction rate (BCR) and confidence interval 95% were used. In particular, F-measure or balanced F-score is a weighted average of precision and recall where precision is the fraction of retrieved instances that are relevant and recall is the fraction of relevant instances that are retrieved. The details of the performance measures are presented in Appendix A.

7.3.1 Evaluation of Feature Transformation and Normalization

ROC curves related to the application of transformation and normalization methods (see Section 5.2.3) on extracted features are provided in Fig. 7.5. As it is shown, the best result was obtained by a combination of transformation $arcsin(\sqrt{x})$ and normalization $x_{ij} = x_{ij}/(max(x_j) - min(x_j))$. Furthermore, the performance of the system was remarkably improved when transformation and normalization operators were applied over all features. It confirms that feature transformation and normalization have an important effect in selection of the most discriminative features.

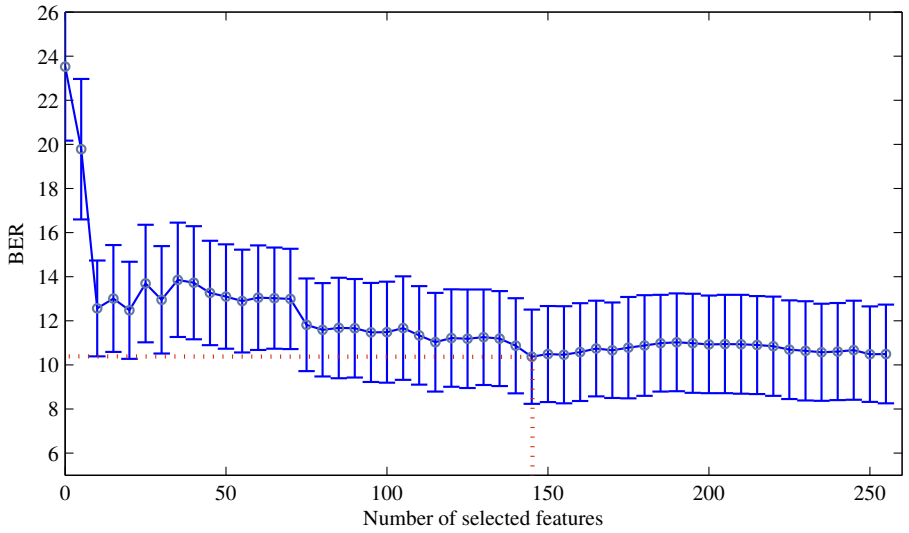

FIGURE 7.6: Balanced error rate (BER) and standard deviation values corresponding to different number of selected features for sleep-awake detection.

7.3.2 Evaluation of Different Number of Features

In order to determine the best number of features in sleep-wake detection and multiclass sleep staging, a grid search was carried out over results obtained with the two-step feature selector (with mRMR) and SVM classifier. As shown in Fig. 7.6 and Fig. 7.7, the lowest average BER values occur for 147 (average BER=10.34) and 326 (average BER=15.32) features for sleep-wake and multiclass sleep staging, respectively. Nevertheless, for both cases, above 100 features the BER values do not improve significantly, *e.g.*, in multiclass sleep staging, the BER value corresponding to 110 features is nearly similar to BER of 326 features, which performs the best result.

7.3.3 Channel Selection

Experiments to find the best combination of EEG, EOG and/or EMG channels were performed. Basically, the AASM rules were followed in channel selection. Table 7.4 and Table 7.5 summarize the attained results of different combinations. As highlighted in the tables for sleep-wake detection, the best channels were 3 EEG channels (C3, C4, and O1), 2 EOG channels (ROC and LOC) and 1 EMG channel (X1). On the other hand, the best performance for multiclass sleep staging was achieved using 9 channels: 6 EEG channels (C3, C4, O1, O2, F3 and F4), 2 EOG channels (ROC and LOC) and 1 EMG channel (X1). Furthermore, distributions of the selected features per channel are shown in Fig. 7.8 and Fig. 7.9. These results show

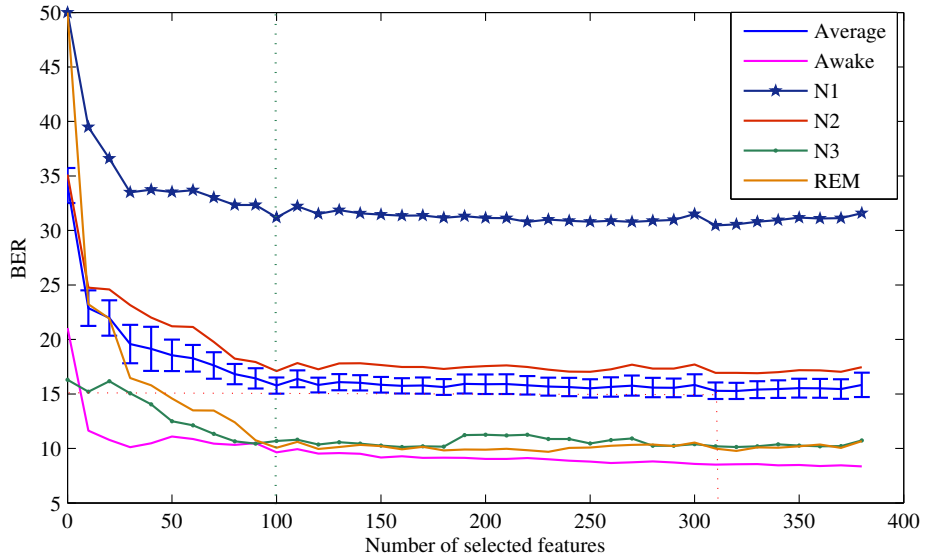

FIGURE 7.7: Balanced error rate (BER) and standard deviation values corresponding to different number of selected features for: multiclass sleep staging; (1) average; (2) awake stage; (3) sleep stages N1; (4) N2; (5)N3; and (6)REM.

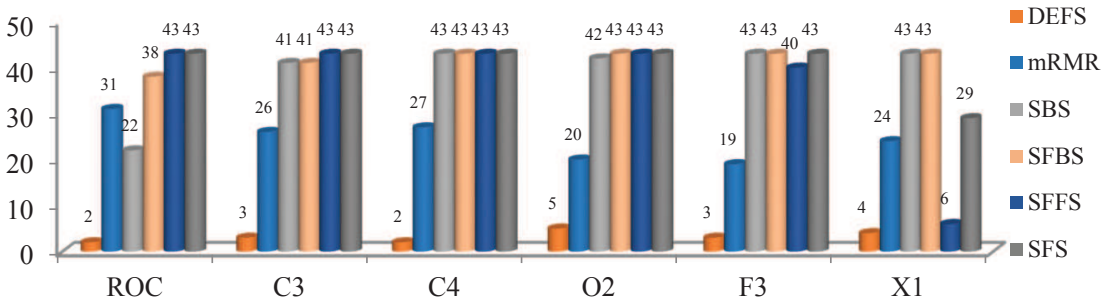

FIGURE 7.8: Number of selected features per channels using different feature selection methods in sleep-wake detection.

the importance of the identified best electrophysiological channels (combinations highlighted in Table 7.4 and Table 7.5). Moreover, the results confirm the fact that sleep-wake detection is highly dependent on the level of alpha activity in central and occipital channels. The EOG and the EMG channels complemented the information. The EOG should record diverse ocular movements during the wake stage, as the EMG chin channel should record high tonic activity. Furthermore the results of Table 7.5, confirm the importance of frontal channels in multiclass sleep staging (*e.g.* in discrimination of REM stage).

TABLE 7.4: Algorithm performance in Sleep–Awake detection with different channels combination.

Channels	BNF	CI	AUC	ACC	BER	F-me	SEN	SPE
C3	20	0.051	81.80	84.51	18.201	72.141	72.14	91.46
C3C4	45	0.052	84.97	90.10	15.030	75.996	76.00	93.94
C3C4O1	66	0.041	88.45	93.28	11.548	81.479	81.48	95.43
C3C4O1F3	90	0.043	88.83	93.77	11.172	82.019	82.02	95.64
C3C4O1LOCROC	118	0.049	89.08	94.35	10.919	82.218	82.22	95.94
C3C4O1LOCROCX1	147	0.042	89.66	94.58	10.344	83.257	83.26	96.06
C3C4O1O2	97	0.043	88.73	93.40	11.272	82.134	82.13	95.32
C3C4O1O2F3F4	112	0.046	89.06	93.83	10.944	82.618	82.62	95.49
C3C4O1O2F3F4LOC	114	0.052	88.81	94.02	11.185	81.948	81.95	95.68
C3C4O1O2F3F4LOCROC	110	0.052	88.34	94.07	11.660	80.846	80.85	95.84
C3C4O1O2F3F4LOCROCX1	160	0.047	89.01	94.55	10.993	81.965	81.96	96.05
C3F3O1	99	0.035	89.27	93.10	10.726	83.221	83.22	95.33
C3F3O1LOC	57	0.046	87.76	92.89	12.240	79.716	79.72	95.81
C3F4O2	88	0.044	87.98	91.13	12.020	80.877	80.88	95.08
C3F4O2LOC	105	0.053	88.66	92.30	11.339	82.155	82.16	95.17
C3F4O2LOCROC	75	0.053	88.95	93.46	11.052	82.411	82.41	95.48
C4F3O1	84	0.048	88.41	93.72	11.593	81.136	81.14	95.68
C4F4O2	80	0.060	86.42	91.63	13.575	77.824	77.82	95.03
C4F4O2ROC	98	0.062	87.51	93.44	12.494	79.163	79.16	95.85

BNF: Best Number of Features, ACC: Accuracy, F-me: F-measure, SEN: Sensitivity, SPE: Specificity

TABLE 7.5: Algorithm performance in multiclass sleep staging with different channels combination.

Channels	BNF	CI	AUC	ACC	BER	F-me	SEN	SPE
C3	26	0.025	73.78	85.66	26.220	58.838	57.03	90.66
C3C4	60	0.023	77.30	88.27	22.700	63.333	62.39	92.31
C3C4O1	93	0.024	79.86	89.79	20.136	68.190	66.60	93.19
C3C4O1F3	129	0.033	79.54	89.56	20.464	67.654	66.01	93.11
C3C4O1LOCROC	175	0.017	83.52	91.69	16.477	72.617	72.62	94.44
C3C4O1LOCROCX1	223	0.014	84.10	91.77	15.899	73.778	73.78	94.47
C3C4O1O2	116	0.024	80.32	89.80	19.682	69.971	67.51	93.19
C3C4O1O2F3F4	200	0.033	80.19	89.81	19.807	67.133	67.13	93.28
C3C4O1O2F3F4LOC	230	0.020	82.88	91.28	17.122	73.244	71.64	94.14
C3C4O1O2F3F4LOCROC	264	0.018	83.89	91.78	16.108	73.291	73.29	94.51
C3C4O1O2F3F4LOCROCX1	326	0.015	84.67	92.04	15.329	74.738	74.74	94.64
C3F3O1	95	0.034	78.59	89.05	21.412	66.583	64.46	92.74
C3F3O1LOC	138	0.025	81.70	90.53	18.297	70.561	69.67	93.66
C3F4O2	100	0.028	78.40	88.63	21.599	64.155	64.16	92.57
C3F4O2LOC	138	0.018	82.64	90.60	17.362	71.381	71.38	93.80
C3F4O2LOCROC	170	0.018	83.40	91.28	16.598	72.540	72.54	94.22
C4F3O1	90	0.032	79.45	89.59	20.553	69.135	65.84	93.09
C4F4O2	98	0.028	78.57	89.01	21.433	66.005	64.49	92.66
C4F4O2ROC	136	0.028	82.44	91.06	17.555	70.902	70.90	94.01

BNF: Best Number of Features, ACC: Accuracy, F-me: F-measure, SEN: Sensitivity, SPE: Specificity

7.3.4 Performance Evaluation with Different Selectors/Classifiers

Fig. 7.10, Fig. 7.11 compare the performance obtained by the proposed method with different combinations of mentioned feature selector/classifiers detailed in Sections 5.2.4 and 5.2.5. In the experiments, six of the best feature selection approaches were used. DEFS, mRMR, and sequential methods (SBS, SFBS, SFFS and SFS). Moreover, four different types of classifiers were considered: Naive Bayes (NB), AdaBoost, LDA and SVM classifiers. They are capable

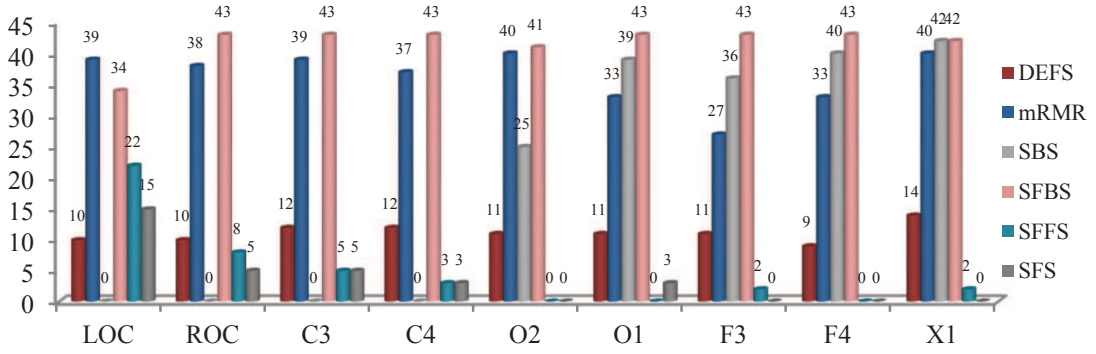

FIGURE 7.9: Number of selected features per channels using different feature selection methods in multiclass sleep staging.

of handling large-scale classification problems. The results are expressed in terms of box-whisker plots showing the average, median, the first and third quartile values of the average accuracies. The horizontal lines outside each box identify the upper and lower whiskers, and dot points denote the outliers. It can be observed from the figures that the higher second and third quartiles and the highest average were attained using mRMR-SVM in both sleep-wake detection and multiclass sleep staging. Moreover, as shown in Fig. 7.11 and Fig. 7.13, some of other combinations (*e.g.* DEFS-SVM approach) also perform very close results requiring, however, much less number of features. In multiclass sleep staging the SVM attained the lowest interquartile range and the highest average of accuracies, as shown in Fig. 7.11. As concerns the sleep-wake detection there is no significant difference between SVM and the other classifiers (see Fig. 7.10).

7.3.5 Evaluation of Feature Relevance

To account with the high dimensionality problem and to infer about the most discriminative features, an analysis was performed using our two-step feature-selection approach (See Fig. 5.2). Firstly, as detailed in Algorithm 1, some of the extracted feature types were selected manually. By analyzing on the histogram of features distribution of whole night sleep and corresponding hypnogram, we inferred the following types of features as being the most discriminative: MODWT based features (energy, percentage of energy, mean and standard deviation of sub-bands), and relative-power, harmonic of theta, sigma, beta and alpha frequency ranges, percentile 75%, kurtosis and skewness. The second step is carried out with the purpose of selecting the final feature elements, *i.e.*, for each feature type the final elements are selected. Therefore, resulted features of the first step, are fed into the feature selectors mentioned in

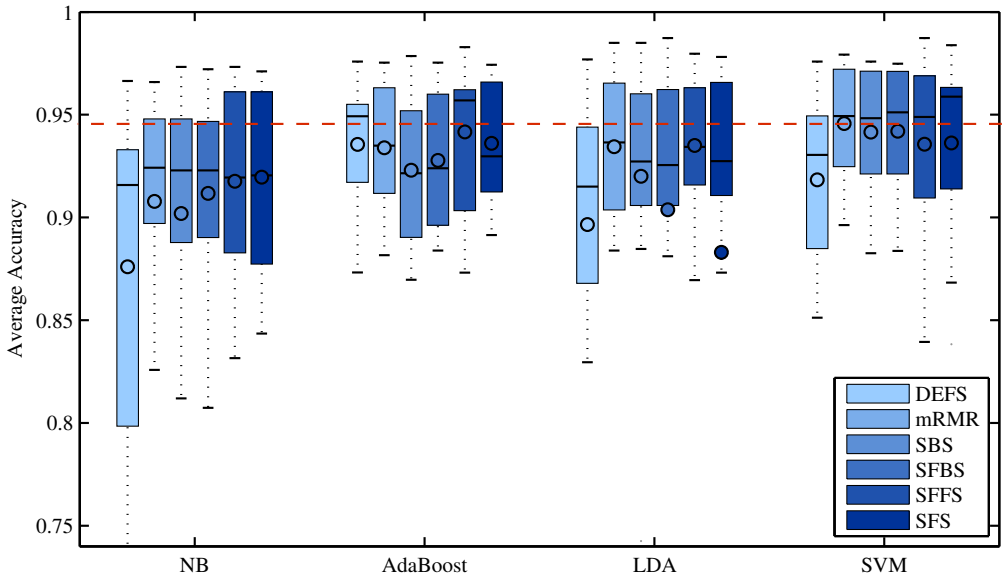

FIGURE 7.10: Accuracy of sleep-wake detection, corresponding to 6 feature selectors (DEFS, mRMR, SBS, SFBS, SFFS and SFS) and 4 classifiers (NB, Adaboost, LDA and SVM).

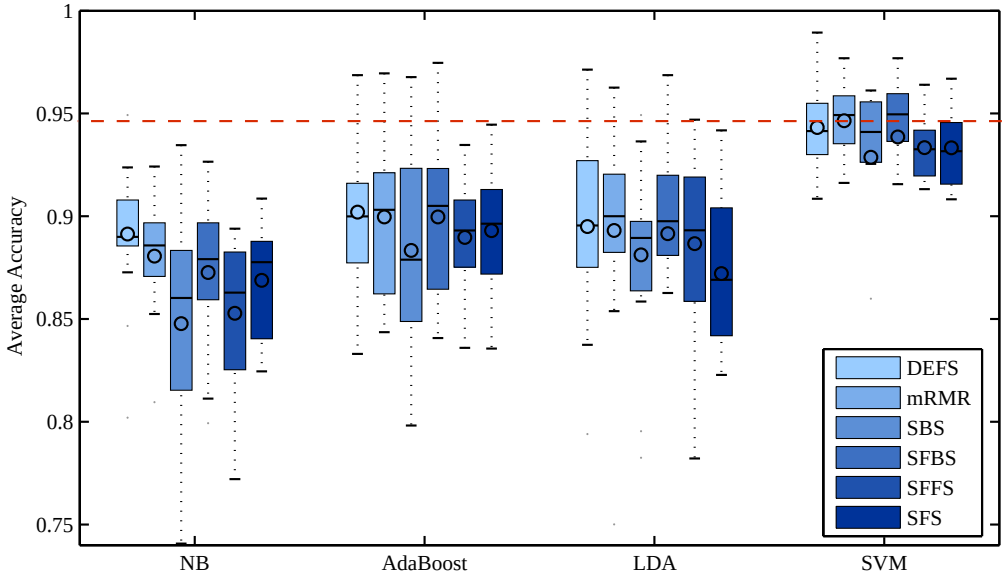

FIGURE 7.11: Accuracy of multiclass sleep staging corresponding to 6 feature selectors (DEFS, mRMR, SBS, SFBS, SFFS and SFS) and 4 classifiers (NB, Adaboost, LDA and SVM).

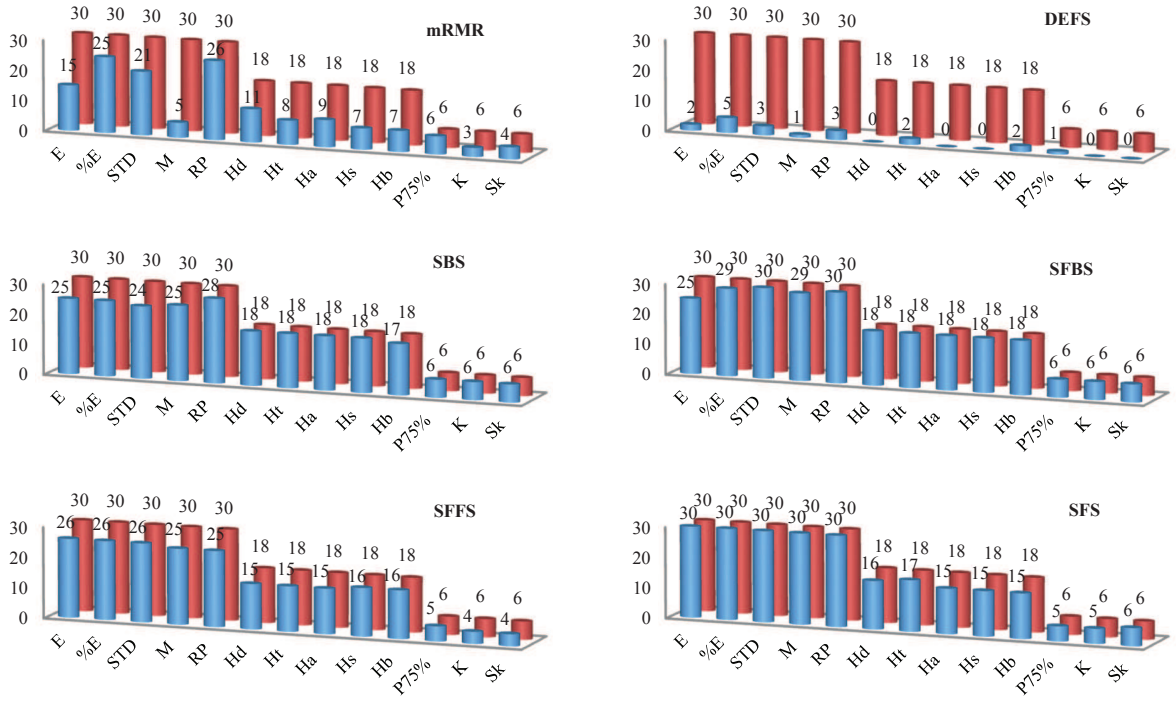

FIGURE 7.12: Selection of the best elements of feature matrix for sleep-awake detection; red: extracted features; blue: selected features. E: energy of sub-bands; %E: percentage of sub-band energy; STD: standard deviation of sub-band energy; M: mean of sub-band energy; RP: relative power; Hd: harmonic-delta; Ht: harmonic-theta; Ha: harmonic alpha; Hs: harmonic-sigma; Hb: harmonic-beta; P75%: percentile 75th; K: kurtosis; Sk: skewness.

Section 5.2.4. As illustrated in Fig. 7.12 and Fig. 7.13 relative-power and percentage-of-energy are the most discriminative features for both sleep-wake detection and multiclass sleep staging. Moreover, it can be inferred from indicated figures that all features which were selected in the first step are important and useful for the classification phase.

7.3.6 Analysis by Gender

In order to evaluate the performance of the proposed method by gender, two experiments were performed: 1) the proposed SSM4S method was trained with data of subjects of both genders; 2) the SSM4S system was trained and tested separately per each type of gender. Fig. 7.14 and Fig. 7.15 provide the accuracies, F-measures, and specificities, obtained with the subgroup-I of ISRUC-sleep dataset, comprising 40 random subjects, 14 female and 26 male subjects. The ASSC method achieved a better performance for both applications when trained/tested separately by gender. Actually, in both applications we had a lower interquartile range on the accuracy, F-measure and specificity of when the ASSC system was trained and tested

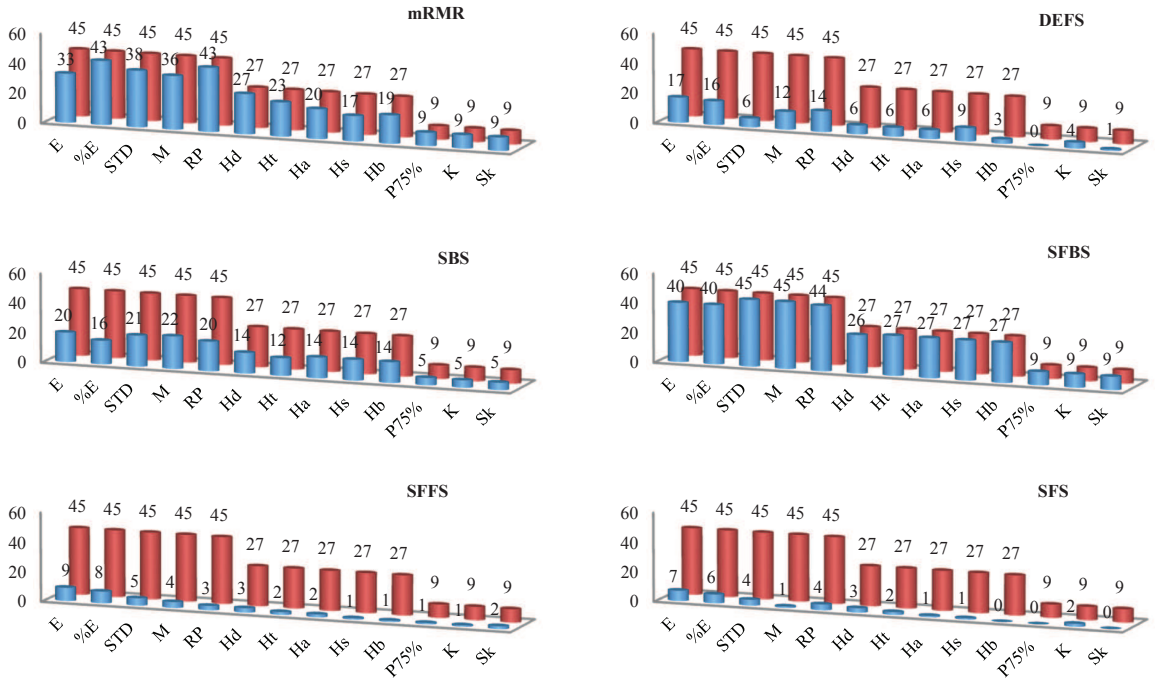

FIGURE 7.13: Selection of the best feature elements for multiclass sleep staging; red: extracted features; blue: selected features. E: energy of sub-bands; %E: percentage of sub-band energy; STD: standard deviation of sub-band energy; M: mean of sub-band energy; RP: relative power; Hd: harmonic-delta; Ht: harmonic-theta; Ha: harmonic alpha; Hs: harmonic-sigma; Hb: harmonic-beta; P75%: percentile 75th; K: kurtosis; Sk: skewness.

separately by gender. Moreover, no significant differences were found in accuracy, F-measure, or specificity between female and male subjects.

7.3.7 Global Performance of The Proposed SSM4S Scheme

Table 7.6 and Table 7.7 summarize the details of the overall performance of the proposed ASSC method using the data of subgroup-I, -II and -III of ISRUC-Sleep dataset. Due to the high number of subjects, all the assessments with the data of subgroup-I, were determined by five- and ten-fold cross validation. However, since there are eight and ten subjects in subgroup-II and -III, respectively, to verify reliability of the results, all the experiments with these two subgroup were done using leave-one subject-out cross-validation (LOOCV) strategy. From the analysis of detailed results[3], it was verified that, the highest performance values were attained with the subjects with longer periods of awake stage during the all-night recording (approximately 8h of data collection). As expected, the ASSC method of *SSM4S*, achieved

[3]The detailed results of the overall performance of *SSM4S*, associated with the dataset is available via http://sleeptight.isr.uc.pt/ISRUC_Sleep/Results

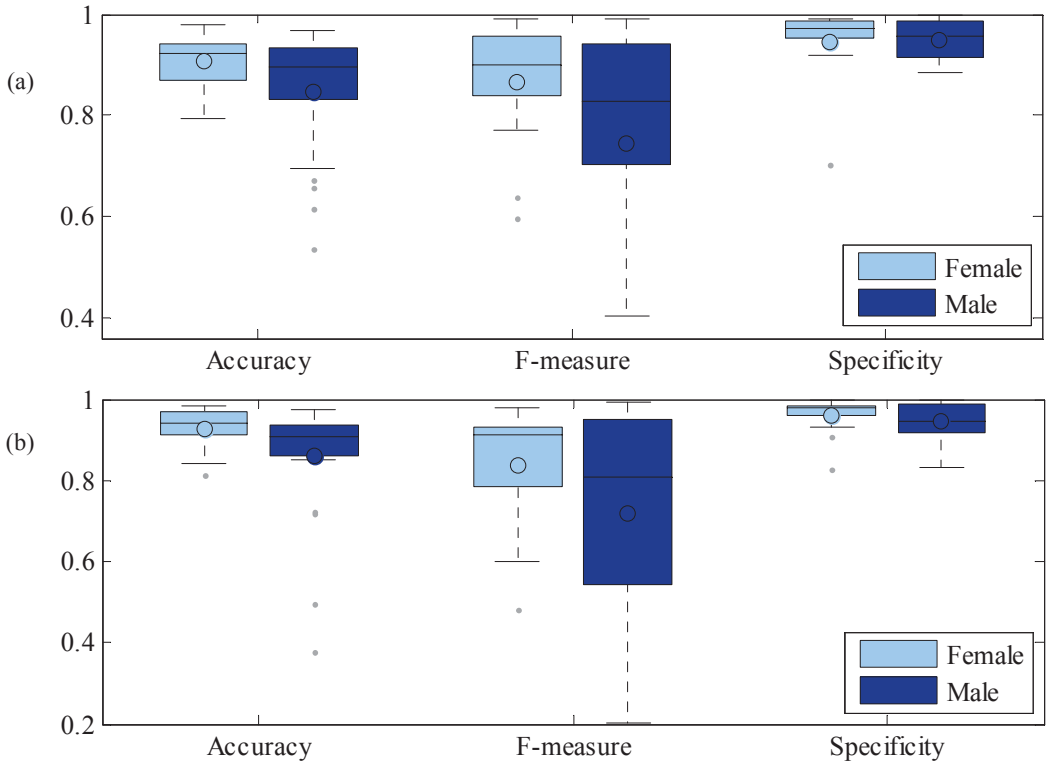

FIGURE 7.14: Accuracy F-measure and specificity of sleep-wake detection correspond to (a) training and test with the same genders (b) training with the both genders.

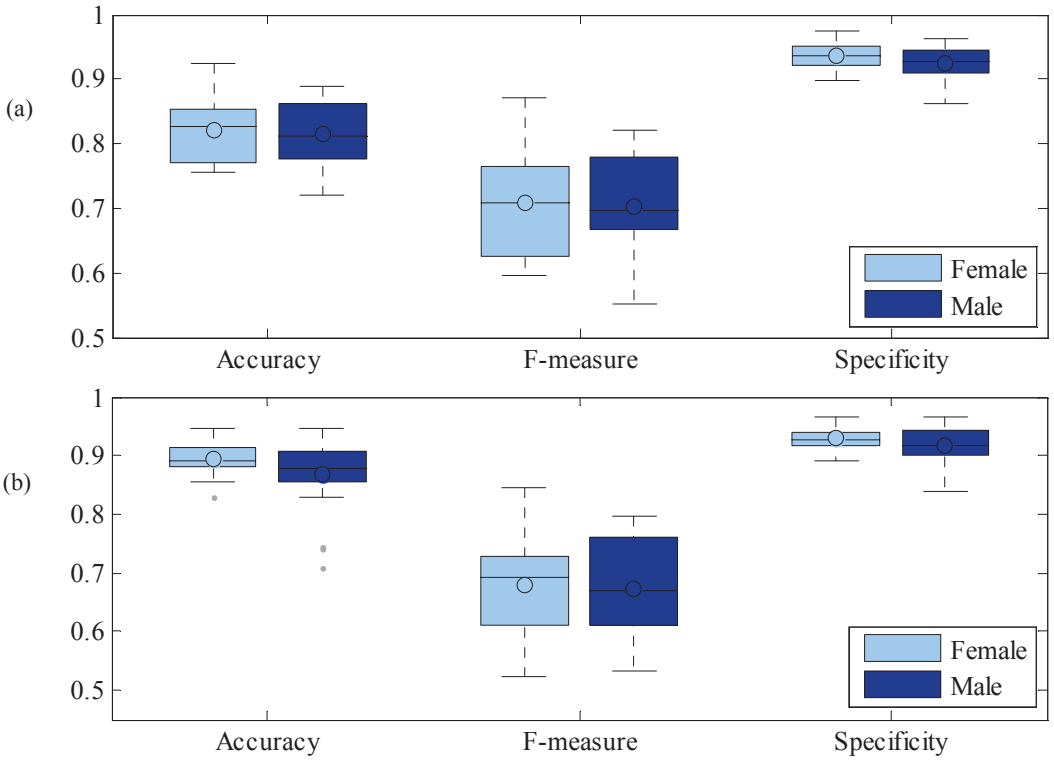

FIGURE 7.15: Accuracy, F-measure and specificity of multiclass sleep staging correspond to (a) training and test with the same genders (b) training with the both genders.

worse average and standard deviation, with data of the subjects with suspected sleep disorders (subgroup-I). The ambiguous patterns on PSG recordings, mainly due to sleep disorders and artifacts, can affect the performance of the method.

Moreover, the recordings of subgroup-I, -II and -III were used to evaluate *SSM4S*, in multiclass sleep staging. Based on average/std., shown in Table 7.7, the best discriminations were achieved for awake, N3, REM, and N2 stages, respectively. The lowest average performance resides in the classification of stage N1.

As concerns REM stage, remarkable results have been achieved by SSM4S method (Table 7.7). In fact, it was verified that employing EOG and EMG channels, improved the REM stage discrimination in comparison with only using EEG channels. In REM stage EOG signals capture the high ocular activity (rapid eye movements) and the EMG signal captures the low level of (chin) muscle tone (the opposite of awake stage). As mentioned, the worst accuracies occurred for N1 stage. Indeed, recognition of N1 is one of the main challenges of sleep staging. There is a lack of discriminative features that characterize N1 stage clearly from the other stages. This has been observed previously by many authors (e.g., Anderer et al. [142]). This could be due to N1 being a transition phase between wakefulness and different sleep stages as discussed in [283]. In fact, the neurophysiologic signals of N1 and N2 stages present similarities between themselves and a mix of patterns with similarities to awake, N3 and REM stages (Table 3.1); *e.g.*, the N1 epochs can present alpha activity (typical of awake stage) and can present theta activity (typical of N2 sleep stage). Moreover, most of the times the N2 sleep stage automatically is misclassified as N1 or N3. Furthermore, concerning pattern similarities between N2 with N3, critical cases are the transition epochs: epochs with a relevant percentage of slow waves but not enough to be classified as N3 sleep stage (Table 3.1). Finally, the worst cases of performance in multiclass sleep staging were mainly related to the older subjects with the high percentages of epochs in N1 and N2 sleep stages[4].

7.4 Analysis of ISRUC-Sleep dataset for ASSC

This section summarizes the main conclusions derived from applying *SSM4S* over ISRUC-Sleep dataset. To analyze the relation of experts agreement and classification performance

[4]The details of ISRUC-Sleep dataset as well as the performance results of applying method *SSM4S* to data of the subgroups-I, -II, -III, for the sleep–wake detection and multiclass sleep staging are available via http://sleeptight.isr.uc.pt/ISRUC_Sleep/Results

TABLE 7.6: Average results of the ASSC method SSM4S for sleep-wake detection. Balanced classification rate(BCR), Specificity (SPEC), Sensitivity (SENS), Accuracy(ACC) are calculated for sleep and Awake stages.

Method	**Applying method on subgroup-I**		**Applying method on subgroup-II**	**Applying method on subgroup-III**
Cross Validation	Five-fold CV	Ten-fold CV	LOOCV	LOOCV
Average BCR	90.16±06.88	90.30±06.29	82.84±10.26	91.19±03.15
Average SENS	83.98±14.99	84.13±14.67	69.11±23.07	85.03±07.05
Average SPEC	96.34±06.88	96.47±03.50	96.57±05.12	97.35±01.37
Average ACC	93.97±06.32	94.10 ±06.19	92.40±05.15	95.39±01.10

TABLE 7.7: Average results of the ASSC method SSM4S for multiclass sleep staging. Balanced classification rate(BCR), Specificity (SPEC), Sensitivity (SENS), Accuracy(ACC) are calculated for each stage

Method	**Applying method on subgroup-I**		**Applying method on subgroup-II**	**Applying method on subgroup-III**
Cross Validation	Five-fold CV	Ten-fold CV	LOOCV	LOOCV
Average BCR-Awake	91.63±5.57	91.75±5.29	85.82±07.41	92.77±03.09
Average BCR-N1	67.34±8.08	67.12±7.94	63.32±10.35	72.35±23.57
Average BCR-N2	83.93±6.64	83.86±6.91	78.06±08.32	79.58±15.26
Average BCR-N3	89.83±8.08	90.11±7.77	87.26±23.56	91.41±05.61
Average BCR-REM	91.63±5.57	92.28±3.55	85.00±23.02	86.50±06.29
Average SPEC-Awake	95.53±4.26	95.23±4.58	95.34±06.26	93.03±03.78
Average SPEC-N1	95.07±3.90	94.92±4.37	91.24±06.47	79.71±13.56
Average SPEC-N2	86.80±7.14	87.50±6.91	81.98±15.98	82.61±04.56
Average SPEC-N3	96.49±4.43	96.70±4.11	87.26±23.56	83.37±17.95
Average SPEC-REM	97.36±2.53	97.04±4.26	95.61±05.16	87.86±10.77
Average SENS-Awake	87.68±12.06	88.28±11.72	76.30±17.58	93.36±04.25
Average SENS-N1	39.61±17.59	39.32±17.51	35.40±24.22	73.41±20.27
Average SENS-N2	81.06±13.14	80.22±14.54	74.14±10.34	86.86±03.78
Average SENS-N3	83.18±17.33	83.52±16.66	78.41±28.35	91.93±06.44
Average SENS-REM	81.10±21.73	81.76±20.33	73.39±27.00	89.19±07.64
Average ACC-Awake	94.15±04.97	94.07±05.05	92.61±04.71	93.43±06.85
Average ACC-N1	88.26±04.50	88.10±04.69	83.18±06.33	75.91±22.10
Average ACC-N2	85.28±05.29	85.49±05.20	77.00±14.03	87.15±06.14
Average ACC-N3	94.00±03.82	94.18±07.77	91.37±05.45	94.38±03.14
Average ACC-REM	89.22±15.43	89.37±15.19	92.64±03.79	91.46±05.44

two measures were calculated. Auto-regressive coefficients, which is a representation of a time series such that it specifies that output variable depends linearly on its own previous values, and balanced correlation rate (BCR) were evaluated.

7.4.1 For sleep-wake detection

- There is a remarkable correlation between the agreement levels of two experts and the classification performance (Fig. 7.16). Despite of this inference, a few exception (recordings related to subjects 12 and 40 of 100 subjects of subgroup-I) were found with high agreement level of experts and very low classification performance. Since alpha activity is one of the relevant patterns in awake stage, this factor can affect the performance of

awake detection. The observed low amplitude of the alpha activity in the EEG signals of subject 12 of subgroup-I can be the main reason of performance degradation. For subject 40 of subgroup-I, the artifacts in EEG signals, which resulted from the low quality of data acquisition, affected the classification performance.

- There is also a correlation between degradation of the classification performance, and the increase of the number of arousals and awakens.
- There is none significant relation between characteristics such as age, gender, diagnosis and medication, and the classification performance.
- Unusual patterns of alpha activity and rapid activity affected the classification performance.

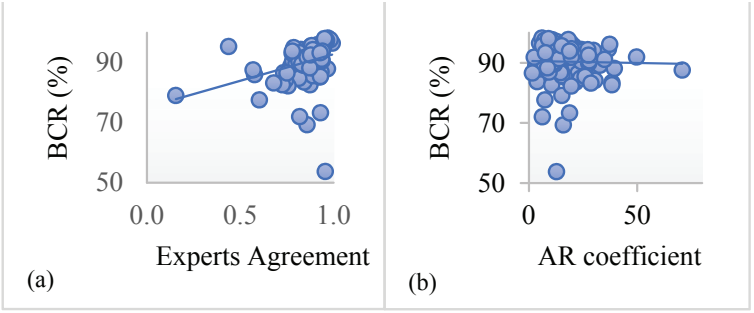

FIGURE 7.16: (a) Distribution of balanced classification rate (BCR) values concerning two experts' agreement in sleep-wake visual scoring; (b) distribution of BCR values corresponding to Auto Regressive (AR)-coefficient, in automatic sleep-wake detection.

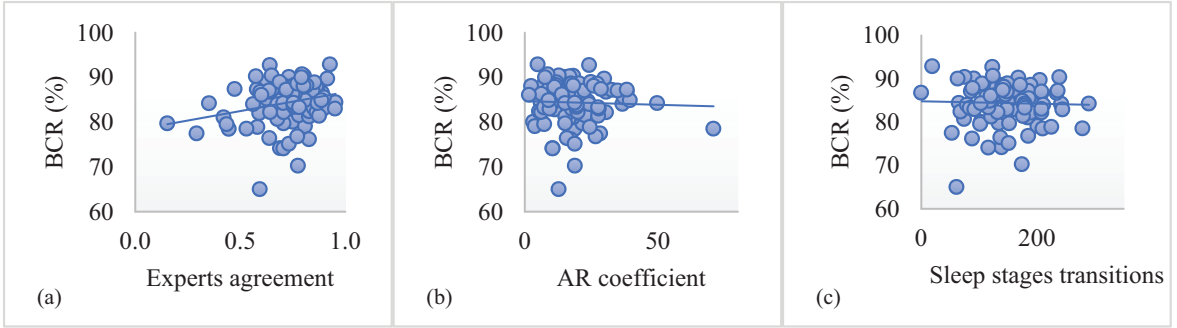

FIGURE 7.17: (a) Distribution of balanced classification rate (BCR) values concerning two experts' agreement in multistage visual scoring; (b) distribution of BCR values corresponding to Auto Regressive (AR)-coefficient, in automatic multiclass sleep staging; (c) distribution of BCR values corresponding to sleep stage transitions.

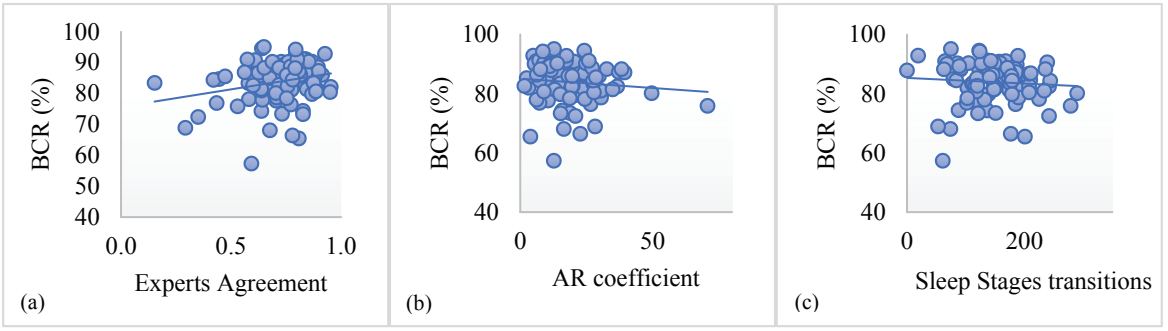

FIGURE 7.18: (a) Distribution of balanced classification rate (BCR) values concerning two experts' agreement in detection N2; (b) distribution of BCR values corresponding to Auto Regressive (AR)-coefficient, in detection N2; (c) distribution of BCR values corresponding to the number of sleep stage transitions.

7.4.2 For multiclass sleep staging

- Similar to sleep-wake detection, a direct relation between experts′ agreement, AR coefficients and the classification performance, were detected (Fig. 7.17).
- Regarding missclassification of stage N2, most of the time, this stage was misclassified as stage N1. Moreover, in most of the epochs with high AR coefficients and low level of experts' agreement, stage N2 was misclassified (Fig. 7.18-a, -b).
- In subjects with higher number of transitions between sleep stages it was verified a trend to lower performances. Since the possibility of stage transition from N2 to the other stages (awake, N1, N3 or REM) is high, once stage N2 is more prevalent in a PSG signal, the classification performance of this stage is affected by the number of stage transitions (Fig. 7.18).
- As mentioned in state of the art, the lowest classification performance was related to N1. Since stage N1 makes a link between the wakefulness and the sleep, it has common transition characteristics. Moreover, in subjects with high misclassification of stage N1, it was verified that, the EEG alterations such as artifacts, paroxystic activity and rapid activity affected the automatic classification.
- For some subjects, the standard deviation from average recognition rate of N3, is too high. For example, in subject 10 of subgroup-I, cardiac and sweat artifacts affected the results. Furthermore, muscle activities, were extensively observed in subjects with higher misclassified N3.

- For REM sleep stage, the ambiguous EEG patterns are the most influencing reason of misclassification.

7.5 Performance Evaluation of IWIVM

In this section, importance weighted import vector machine, the contribution introduced in Section 6.4 is evaluated. In order to assess the effectiveness of the proposed IWIVM method, several experiments have been carried out in classification tasks under covariate shift. In particular, we assessed the method performance on two different domain adaptation problems: 1) a two-dimensional binary toy classification problem under covariate shift, 2) a real world cross-domain object recognition task using the benchmark datasets Office [175] and Caltech-256 [267].

We focus on unsupervised domain adaptation setting. Moreover, aiming to compare with the

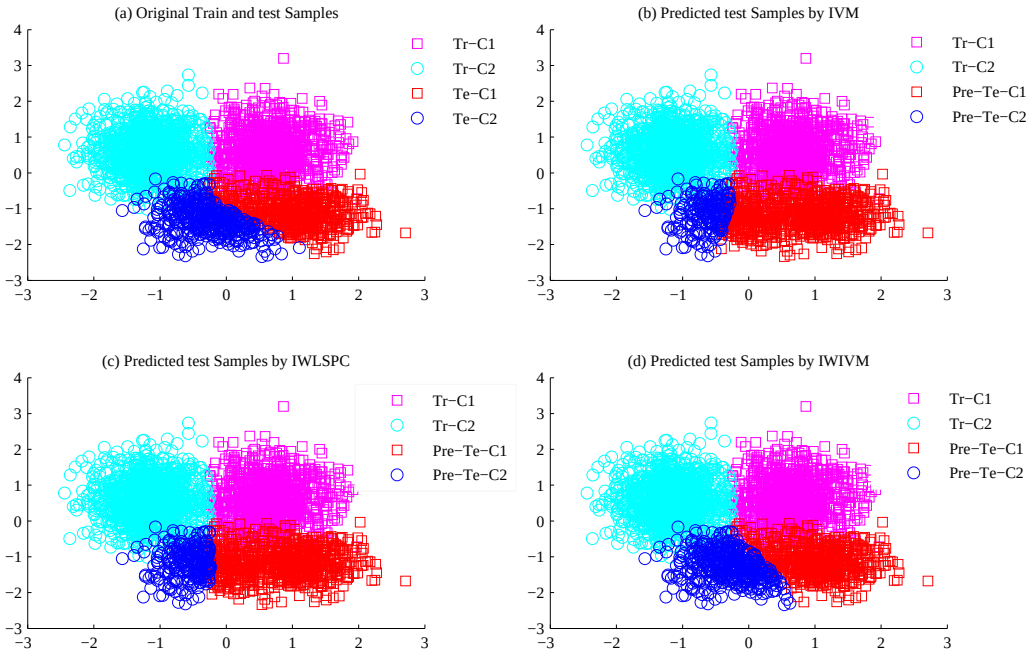

FIGURE 7.19: Performance analysis over two-dimensional two-classes toy problem corresponding to (a) Original samples, (b) Predicted test samples by IVM, (c) Predicted test samples by IWLSPC, and (d) Predicted test samples by IWIVM. Tr-Ci: Training samples of class-i, Te-Ci: Test samples of class-i, Pre-Te-Ci: Predicted test samples of class-i.

state-of the-art methods for cross domain object recognition, results of semi-supervised DA are presented. For all the datasets, true labels are available for both source- and target-domain instances. However, in case of unsupervised DA, prior information related to the target domain

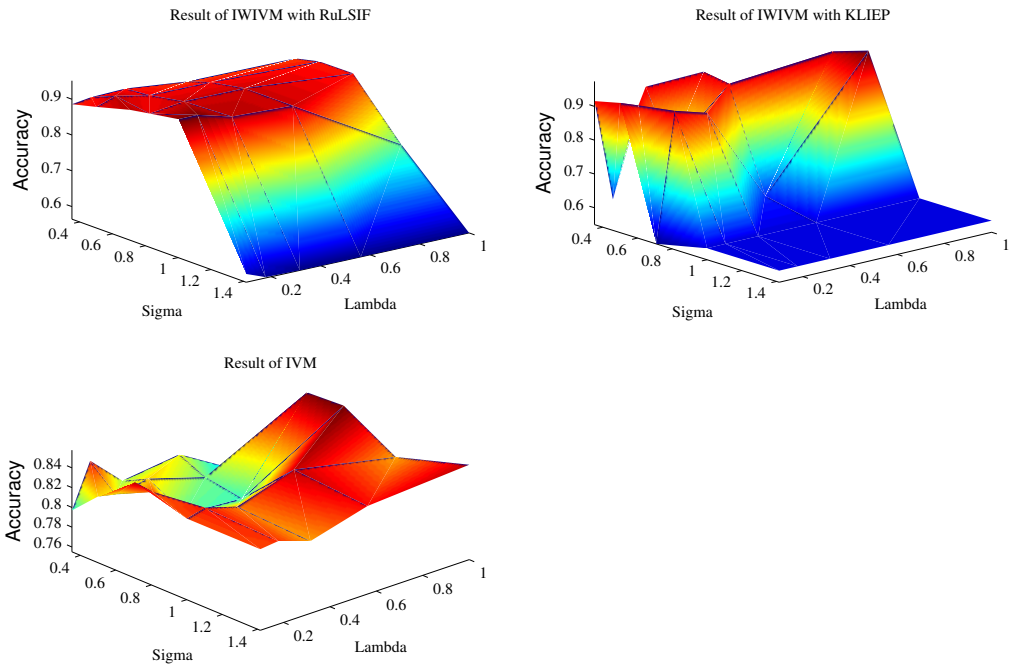

FIGURE 7.20: Classification accuracies corresponding to different values of kernel width σ and regularization factor λ.

was considered only for performance assessment of the proposed method. In order to excel the effectiveness in different kind of problems, Gaussian kernel functions

$$K\left(x, x^{'}\right) = exp\left(-\frac{\left\|x - x^{'}\right\|^2}{2\sigma^2}\right) \tag{7.1}$$

where σ denotes the Gaussian kernel width, were used in all the experiments.
IWIVM and RIWCV require the values of the importance ratio $\{w(\boldsymbol{x}_n^s)\}_{n=1}^{N_s}$, which are unknown in practice. Among the importance estimation methods explained in Section 4.5, the KLIEP and RuLSIF were employed in our experiments, since they were shown to be superior. We use the overall accuracy (*i.e.*, the percentage of correctly labeled samples over the whole number of considered samples) as reference measure for the assessment of the methods.

The predicted results of applying IVM, IWLSPC, and IWVM on the toy dataset are illustrated in Fig. 7.19-b, -c, -d, respectively. As expected, using instance weighing improves the classification accuracy against to normal supervised methods.
The quality of the proposed method compared to some of the state-of-the-art methods is shown in Table 7.8. The mean and standard deviation of error rate for two different sizes of training instances confirm that IWIVM+RIWCV outperform the other methods. Moreover,

the RIWCV strongly improves the results, which confirms that using RIWCV selection of the most suitable parameters under covariate shift. Table 7.9 shows a comparison of training computation time of the methods. IWIVM is comparable with the state-of-the-art methods in terms of mean computation time by the t-test.

In what follows, the results obtained in a real-world corss-domain object recognition problem will be presented. Table 7.10 summarize the accuracy rates for unsupervised DA, in corss-domain object recognition. The accuracy rates of the method with the optimal parameters were used for comparison. The highest accuracy were achieved by the IWIVM method where it has shown to outperform all other methods in all domain combinations. Furthermore, comparing with the other methods, the IWIVM achieved these results with a lower number of features (10 features). Noticeably, the accuracy rate was improved regardless of employing the type of source and target domains.

Table 7.11 reports recognition accuracies on the semi-supervised domain adaptation. The overall accuracy of the IWIVM on 12 domain shifts prove the effectiveness of our method on the semi-supervised setting. Looking at individual domain shifts it was verified that IWIVM outperforms all other methods in 6 out of the 12 domain shifts. The method also shows a state-of-the-art performance for the remaining domain shifts. The Geodesic flow kernel (gfk) [172] method also performs well on some of domains shift. When webcam and DSLR are the target domain, the performance of IWIVM is lower than gfk. This result may be due to the simple PCA features that are used.

TABLE 7.8: Average error rate of 50 trials for the toy dataset. Percentage mean and standard deviation (in parentheses) for two different number of training instances.

N_{ta}	LSPC	IWLSPC	LapRLS	IWKLR	IVM	IVM	IWIVM	IWIVM	IWIVM
CV	CV	IWCV	IWCV	IWCV	CV	RCV	IWCV	RIWCV	Best Params
1000	16.6(3.4)	15.2(5.5)	16.5(6.8)	15.2(5.4)	18.02(2.4)	17.59(2.5)	16.85(3.1)	**8.93(3.1)**	**8.56(0.9)**
2000	17.2(3.1)	13.3(5.8)	15.7(7.0)	14.1(6.3)	18.64(2.1)	17.06(2.9)	16.02(5.2)	**8.34(4.9)**	**8.18(1.1)**

TABLE 7.9: Average computation times of training set for 50 trial for the toy dataset. Mean and standard deviation (in parentheses) for two different number of training instances.

N_{ta}	LSPC	IWLSPC	LapRLS	IWKLR	IVM	IVM	IWIVM	IWIVM	IWIVM
CV	CV	IWCV	IWCV	IWCV	CV	RCV	IWCV	RIWCV	Best Params
1000	**0.6(0.2)**	16.0(4.5)	3.5(2.1)	3.0(1.1)	1.10(0.65)	**1.07(0.74)**	11.68(6.23)	9.04(1.97)	**1.23(0.2)**
2000	**1.2(0.3)**	14.0(5.9)	4.9(3.0)	7.6(2.3)	2.48(1.15)	**1.96(0.16)**	11.04(5.96)	10.74(3.25)	**3.7(0.24)**

TABLE 7.10: Recognition accuracy rates on target domains with unsupervised adaptation (C: Caltech, A: Amazon,W: Webcam, and D: DSLR). The left of "→" indicates the source domain and the right of "→" is the target domain.

Method	no adapt	PCA s	PCA ta	PLS s	GFS (opti.)	SGF	gfk (S,A)	gfk (A,A)	IVM+ PCA	IWIVM+ PCA
$C \to A$	20.8±0.4	34.7±0.5	37.5±0.4	26.7±0.9	36.9±0.5	36.8±0.5	40.4±0.7	36.9±0.4	43.51±0.7	**46.42±0.5**
$C \to W$	19.4±0.7	31.3±0.6	33.9±1.1	26.0±0.6	33.9±1.2	00.0±0.0	35.8±1.0	33.7±1.1	35.09±1.1	**41.45±1.1**
$C \to D$	22.0±0.6	33.6±1.2	37.8±0.9	28.2±1.3	35.2±1.0	32.6±0.7	41.1±1.3	35.2±1.0	37.68±0.9	**46.42±0.5**
$A \to C$	22.6±0.3	34.0±0.3	35.4±0.4	31.1±0.5	35.6±0.4	35.3±0.5	37.9±0.4	35.6±0.4	36.58±0.3	**40.15±0.2**
$A \to W$	23.5±0.6	31.3±0.5	34.9±1.0	29.3±0.9	34.4±0.9	31.0±0.7	35.7±0.9	34.49±0.9	38.51±0.6	**47.75±0.4**
$A \to D$	22.2±0.4	29.4±0.8	33.3±0.8	28.0±1.0	34.9±0.9	00.0±0.0	35.1±0.8	35.2±0.9	35.31±0.8	**45.64±0.6**
$W \to C$	16.1±0.4	23.4±0.6	29.6±0.5	18.3±0.5	27.3±0.5	21.7±0.4	29.3±0.4	27.2±0.5	32.24±0.5	**39.31±0.5**
$W \to A$	20.7±0.6	28.0±0.5	32.5±0.8	21.1±0.9	31.3±0.7	27.5±0.5	35.5±0.7	31.1±0.8	37.31±0.4	**42.16±0.5**
$W \to D$	37.3±1.2	68.2±1.0	67.4±0.7	42.8±1.4	70.7±0.9	00.0±0.0	**71.2±0.9**	70.6±0.9	56.34±1.0	58.51±0.9
$D \to C$	24.8±0.4	26.8±0.3	31.2±0.3	21.4±0.6	30.0±0.2	00.0±0.0	32.7±0.4	29.8±0.3	33.92±0.2	**39.67±0.3**
$D \to A$	27.7±0.4	28.1±0.3	34.4±0.3	26.5±0.6	32.6±0.5	32.0±0.4	36.2±0.4	32.5±0.5	35.11±0.3	**39.13±0.3**
$D \to W$	53.1±0.6	61.7±0.7	**79.4±0.5**	41.9±1.4	74.9±0.6	66.0±0.5	**79.1±0.7**	74.9±0.5	76.04±0.6	**79.72±0.5**

TABLE 7.11: Recognition accuracy rates on target domains with semisupervised adaptation (C: Caltech, A: Amazon,W: Webcam, and D: DSLR). The left of "→" indicates the source domain and the right of "→" is the target domain.

Method	no adapt	PCA s	PLS s	GFS (opti.)	Metric	gfk (S,A)	SGF	svms	IVM+ PCA	IWIVM+ PCA
$C \to A$	23.1±0.4	40.3±0.4	36.8±0.5	42.0±0.5	33.7±0.8	46.1±0.6	40.2±0.7	35.9±0.4	46.79±2.3	**48.39±5.5**
$C \to W$	25.2±0.8	54.2±0.9	45.9±1.0	54.2±0.9	34.7±1.0	**57.0±0.9**	00.0±0.0	30.8±1.1	52.94±2.5	56.04±1.1
$C \to D$	26.5±0.7	48.9±1.0	43.1±1.0	50.2±0.8	33.7±0.8	**55.0±0.9**	36.6±0.8	35.6±0.7	48.03±2.2	52.87±0.7
$A \to C$	24.0±0.3	35.5±0.5	31.4±0.6	37.5±0.4	27.3±0.7	39.6±0.4	37.7±0.5	35.1±0.3	42.02±1.4	**46.59±0.3**
$A \to W$	31.6±0.6	47.3±0.7	41.4±0.9	54.2±0.8	36.0±1.0	**56.9±1.0**	37.9±0.7	33.9±0.7	52.38±2.7	52.49±0.5
$A \to D$	28.1±0.6	47.8±1.0	45.5±1.1	46.9±1.1	33.7±0.9	**50.9±0.9**	00.0±0.0	35.0±0.8	46.57±1.5	46.34±0.5
$W \to C$	20.8±0.5	28.1±0.8	24.7±0.7	32.9±0.7	21.7±0.5	32.3±0.6	29.2±0.7	31.3±0.4	33.29±1.1	**38.15±0.3**
$W \to A$	30.8±0.6	38.2±0.6	32.2±0.9	43.0±0.7	32.3±0.8	**46.2±0.7**	38.2±0.6	35.7±0.4	44.41±1.0	**46.44±0.4**
$W \to D$	44.3±1.0	72.1±0.8	49.1±0.9	**75.2±0.7**	51.3±0.9	74.1±0.9	00.0±0.0	66.6±0.7	68.70±2.2	72.56±0.5
$D \to C$	22.4±0.5	27.0±0.5	26.0±0.8	32.9±0.4	22.5±0.6	33.9±0.6	00.0±0.0	31.4±0.3	37.47±0.7	**39.22±0.3**
$D \to A$	31.3±0.7	36.8±0.5	34.5±0.4	44.9±0.7	30.3±0.8	**46.2±0.6**	39.2±0.7	34.0±0.3	45.92±1.8	**46.27±0.7**
$D \to W$	55.5±0.7	64.4±0.7	49.4±1.2	78.6±0.4	55.6±0.7	80.2±0.4	69.5±0.9	74.3±0.5	76.28±3.1	**81.13±0.4**

TABLE 7.12: Average results for sleep-wake detection corresponding to (a) applying SSM4S using SVM to Subgroup-II, (b) applying SSM4S using IVM to Subgroup-II, and (c)applying adaptive-SSM4S using IWIVM to Subgroup-II. Balanced error rate(BER), Specificity (SPEC), Sensitivity (SENS), Accuracy(ACC) are calculated for each stage.

Method	**(a) Applying SSM4S on subgroup-II**	**(b) Applying SSM4S on subgroup-II**	**(c) Applying Adaptive-SSM4S on subgroup-II**
Classifier	LOOCV-SVM	LOOCV-IVM	LOOCV-IWIVM
Average BER	19.11±10.66	16.62±12.40	13.84±3.75
Average SENS	73.81±21.20	77.45±12.85	83.29±7.05
Average SPEC	96.14±04.97	90.29±01.28	96.39±3.37
Average ACC	92.83±06.41	88.52±09.84	94.75±4.19

7.6 Performance Assessment of The Adaptive ASSC

In order to evaluate the adaptive-ASSC method, introduced in Section 6.6, eight subjects of Subgroup-II of ISRUC-Sleep dataset, each with two different recording sessions, were used. Two types of experiments have been carried out: sleep-wake detection and multiclass sleep staging. The performance was determined using leave-one subject-out cross-validation (LOOCV). In our experiments, a fourth order Daubechies with MODWT decomposition was adopted. The extracted feature sets were normalized and the corresponding selected features were determined using the two-steps feature selector (see Section 5.2.4). The most relevant features were selected from MODWT decomposition, Harmonic parameters, Hjorth parameter, relative spectral power, percentile 75, skewness and Kurtosis as detailed in Fig. 5.1. In order to assess the performance two experiments were performed including: 1) seven subjects, each with two different recording sessions, were exploited to train the SSM4S method. IVM and SMV (Libsvm toolbox [282]) were used in the classification phase. For IVM, Gaussian kernel width 0.4, and regularization parameter 0.01, and for SVM, the sigmoid kernel degree and C parameters were set to 0.13 and 1.25 were used, as they produced the best empirical results. 2) seven subjects, each with two different recording sessions, were exploited to train the adaptive SSM4S method. Since, the LOOCV strategy was used in all experiments thus, in each iteration, two recordings of the test subject were used to obtain the weights (to make the adaptation) and to test the method, respectively. In application of sleep-wake detection, to overcome the biasness, we have randomly selected some of the sleep epochs for training of the methods, yielding an improvement in general performance. Table 7.12 and Table 7.13 summarize the details of the average performances of the SSM4S method using SVM and IVM classifiers as well as the adaptive SSM4S method using IWIVM. From the analysis of detailed

TABLE 7.13: Average results for multiclass sleep staging corresponding to (a)applying SSM4S using SVM to Subgroup-II, (b) applying SSM4S using IVM to Subgroup-II, and (c)applying adaptive-SSM4S using IWIVM to Subgroup-II. Balanced error rate(BER), Specificity (SPEC), Sensitivity (SENS), Accuracy(ACC) are calculated for each stage.

Method	**(a) Applying SSM4S on subgroup-II**	**(b) Applying SSM4S on subgroup-II**	**(c) Applying Adaptive-SSM4S on subgroup-II**
Classifier	LOOCV-SVM	LOOCV-IVM	LOOCV-IWIVM
Average BER-Awake	12.84±8.61	14.59±5.71	10.84±09.31
Average BER-N1	34.63±7.40	37.58±4.18	32.73±06.22
Average BER-N2	23.70±9.39	23.89±7.67	19.70±15.26
Average BER-N3	12.87±6.70	13.54±4.33	09.10±07.72
Average BER-REM	14.00±8.75	15.41±4.05	11.89±06.75
Average SPEC-Awake	95.60±4.12	96.63±1.22	95.46±02.96
Average SPEC-N1	91.79±6.44	91.49±3.55	92.34±4.61
Average SPEC-N2	82.65±8.71	81.13±6.04	86.31±11.13
Average SPEC-N3	91.22±3.26	89.77±4.59	97.03±2.64
Average SPEC-REM	96.03±1.07	94.12±3.52	96.47±5.33
Average SENS-Awake	87.41±11.47	77.08±16.39	82.85±17.83
Average SENS-N1	37.10±16.71	34.91±20.02	39.37±20.85
Average SENS-N2	74.85±13.93	73.63±14.21	74.69±8.76
Average SENS-N3	82.29±16.54	80.17±17.46	84.73±14.88
Average SENS-REM	76.49±22.01	75.28±20.08	80.91±16.07
Average ACC-Awake	93.37±5.34	89.66±8.94	94.12±4.08
Average ACC-N1	84.2±11.09	82.12±11.02	81.79±12.11
Average ACC-N2	79.58±4.24	80.67±12.66	82.35±7.3
Average ACC-N3	92.01±3.29	91.09±2.45	94.08±13.61
Average ACC-REM	93.8±10.21	88.97±1.29	92.77±9.5

results[5], it was verified that, the lowest balanced error values were attained with the adaptive SSM4S in both applications sleep-wake detection and multiclass sleep staging. The best and the worst results are attained in recognition of stages N1 and N3, respectively. Comparing to IVM, all of the performance measures have shown an improvement on the results (Table 7.13), which confirm that, covariate shift adaptation-based methods improve the performance of the ASSC. However, in subjects with sleep disorders (subjects 1 and 7) SVM performs better than IWIVM, which confirm that, exploiting the ambiguous patterns on PSG recordings for weighting and adaptation affect the performance of the proposed adaptive method.

[5]The detailed results of the overall performance of *Adaptive-SSM4S*, associated with the dataset is available via http://sleeptight.isr.uc.pt/ISRUC_Sleep/Results

Part III

Conclusions

Chapter 8

Conclusions and Outlook

This chapter concludes the results, reviews some of the main contributions and points out the future directions of research. Section 8.1 presents a summary of contributions and concluding remarks. Section 8.2 discusses potential directions for extending the current study.

8.1 Conclusions

8.1.1 ISRUC-Sleep Dataset: A Comprehensive Public Dataset for Researchers

The ISRUC-Sleep dataset, which contains PSG recordings of different subjects, was introduced. This dataset was created aiming to complement existing datasets by providing easy-to-apply data collection with some characteristics not covered yet. In addition, a set of scripts was developed and turn publicly available allowing to test new algorithms and experiments replication. ISRUC-Sleep dataset is useful for research: (i) in biomedical signal processing; (ii) in development of new ASSC methods; and (iii) on sleep physiology.

Even though other publicly available datasets exist, to the best of our knowledge, except Sleep-EDF dataset(expanded), which were recorded during years 1987-1991 in two subsequent day-night periods at the subjects' homes, there is no traceable public dataset providing two recordings for the same subject in different time. On the other hand, comparing to the other datasets such as *MASS*, which mostly contains data of healthy subjects, the ISRUC-Sleep dataset includes data of healthy subjects, subjects with sleep disorders, and subjects under the effect of sleep medication. This variety of data can be useful for generalization purposes. In

ISRUC-Sleep dataset, for each subject two hypnograms, created independently by two human-experts, are provided.

The details, benefits and characteristics of different subgroups of the dataset were illustrated by analyzing the performance of the *SSM4S* method. According to the results, there is a direct relation between the disagreement level of two experts and degradation of the classification performance. As expected, PSG recordings affected by artifacts and sleep related disorders, contain the most challenging patterns for sleep PSG analysis. Furthermore, some specific events and variations in usual neurophysiological patterns, such as variations in the alpha brain activities, are the main causes of performance drop in ASSC.

8.1.2 SSM4S Method: A Reliable Subject Independent ASSC Method

To discriminate the sleep stages based on AASM standard, a subject independent ASSC method SSM4S was proposed for sleep-wake detection and for multiclass sleep staging (awake, NREM (N1, N2, N3) sleep and REM sleep). The method employs the advantages of extracted features from multi-channels EEG, EOG and EMG signals according to temporal, frequency and time-frequency domains. Applying the MODWT, which omitted subsampling in the filtering process, provided the shift invariance characteristic to our method which is one of the most important properties in analysis of PSG signals. To reduce the effect of extreme values in the feature vectors the extracted feature set was transformed and normalized, which improved the overall performance. Moreover, by using the two-step feature selector it was inferred that, relative-power and percentage-of-energy are the most discriminative features for both sleep-wake detection and multiclass sleep staging (Fig. 7.12 and Fig. 7.13). The proposed method performed the best performance by combining 6 channels (C3, C4, O1, ROC, LOC and X1) for sleep-wake detection, and 9 channels (C3, C4, O1, O2, F3, F4, ROC, LOC, X1) for multiclass sleep staging (Table 7.4 and Table 7.5). The experimental study was performed using the ISRUC-Sleep dataset, which is a rich dataset composed by PSG signals from three subgroups of subjects with different characteristics (i.e. young/old, male/female, non-apnea/with apnea event and other sleep problems). The overall performance of the *SSM4S* method applied to PSG signals of different subgroups subjects reached the remarkable results for sleep-wake detection and multiclass sleep staging (Table 7.6 and Table 7.7).

8.1.3 Importance Weighting Import Vector Machine: A Unsupervised Domain Adaptation Method

In Chapter 6, we addressed a particular transfer learning task named covariate shift adaptation. The main objective of covariate shift adaptation techniques is to take advantage of the available knowledge on a given source domain in order to infer a model/classifier suitable for the classification of a related target domain. In real word applications, according to a distribution, the target data are different but related with the source domain.
We proposed an adaptive IVM, which is an adaptive sparse kernel logistic regression approach, for unsupervised and semi-supervised domain adaptation. The IWIVM approach, which is an instance adaptive method, assigns a weight to each sample of source domain based on their importance in target domain.
We illustrated the behavior of the method using two-dimensional toy problem. Moreover, we compared the proposed method on Office+Caltech datasets for a cross-domain object recognition task. IWIVM obtained the best performance even with a simple and fewer number of PCA features, and eliminated the need to make changes on features for each problem. Moreover, to optimize the parameters such as the Gaussian kernel width, the regularization and flatting parameters, the RIWCV was proposed. This stable version of IWCV, eliminates the odds of falling down in local optima by selection of the most stable values of the parameters. This method has proven to be effective in the discussed tasks.

8.1.4 Adaptive Sleep Stage Classification Method

An adaptive automatic sleep stage classification method, based on unsupervised domain adaptation, was proposed. To determine the validity of ASSC under covariate shift adaptation, importance weight import vector machine (IWIVM), which is an instance of unsupervised domain adaptation methods, compared with IWKLR, KLR, IVM and SVM. For this purpose, several feature extraction methods were applied. Features with higher positive impact in classification accuracy were the MODWT decomposition, harmonic parameters and relative power. Transformation and normalization in the feature domain were played an important role in the remarkably improvement of classification accuracy. The Adaptive ASSC method shows promising results in both applications, sleep versus wake and multiclass sleep stage classification. In order to attain the better results, RIWCV were used to find the more reliable values for the parameters.

8.2 Future Perspectives

The study conducted along this thesis led to interesting possibilities for further research works. We believe that the obtained frameworks for automatic sleep staging, unsupervised domain adaptation and IRIS recognition have the capability to be extended to different scenarios and applications. The future directions are described in the sequel:

- We propose to research, a new instance and feature domain adaptation method. Due to the effectiveness of instance adaptation and feature adaptation methods, a new unsupervised adaptive method, which consider both types of adaptation using instance and feature weighting methods, can be useful and highly desirable.
- Density ratio (weight) calculation, is one of the most effective factors in instance-based domain adaptation methods. To obtain the weights (density ratio), some of unlabeled instances form the target domain are randomly selected and used. However, a non-random selection of unlabeled target instances using clustering methods can improve the precision of the weight calculation methods such as KLIEP.
- One of the main challenges in the context of automatic sleep staging is the disagreement between the human experts in the labeling process, which yields different labels for the same epoch of the signal. These labels are used as ground through in the supervised learning methods. This issue corresponds to the labeling difference or concept drift problem, which means for the same signal we have different labels. To improve the performance of automatic sleep staging under concept drift, domain adaptation can be applied.

Appendix A

The AASM Rules for Sleep Scoring

The American academy of sleep medicine (AASM) define some characteristics and rules for sleep scoring according to the amplitude, frequency and shape of the PSG signals. The sleep stages are characterized as follows [114]:

- Wake (W)

  A. Score epochs as stage Wake when more than 50% of the epoch has alpha rhythm over the occipital region.

  B. Score epochs without visually discernible alpha rhythm as stage Wake if any of the following are present: 1) Eye blinks at frequency of 0.5-2 Hz; 2) Reading eye movements 3) Irregular conjugate rapid eye movements associated with normal or high chin muscle tone.

- N1

  A. In subjects who generate alpha rhythm, score stage N1 if alpha rhythm is attenuated and replaced by low amplitude, mixed frequency activity for more than 50% of the epoch.

  B. In subjects who do not generate alpha rhythm, score stage N1 commencing with the earliest of any of the following phenomena: 1) Activity in range of 4-7 Hz with slowing of background frequencies by $\geqslant 1Hz$ from those of stage W; 2) Vertex sharp waves; 3) Slow eye movements.

- N2

  A. Begin scoring stage N2 (in absence of criteria for N3) if 1 or both of the following occur during the first half of that epoch or the last half of the previous epoch: a) One or

more K complexes unassociated with arousals; b) One or more trains of sleep spindles.

B. Continue to score epochs with low amplitude, mixed frequency EEG activity without K complexes or sleep spindles as stage N2 if they are preceded by K complexes unassociated with arousals or sleep spindles.

C. End stage N2 sleep when one of the following events occurs: 1) Transition to stage W; 2) An arousal (change to stage N1 until a K complex unassociated with an arousal or a sleep spindle occurs); 3) A major body movement followed by slow eye movements and low amplitude mixed frequency EEG without non-arousal associated K complexes or sleep spindles (score the epoch following the major body movements as stage N1; score the epoch as stage N2 if there are no slow eye movements); 4) Transition to stage N3; 5) Transition to stage REM.

- N3

  A. Score stage N3 when 20% or more of an epoch consists of slow wave activity, irrespective of age.

- REM

  A. Score stage REM sleep in epochs with all the following phenomena: 1) Low amplitude, mixed frequency EEG; 2) Low chin EMG tone; 3) Rapid eye movements.

  B. Continue to score stage REM sleep, even in the absence of rapid eye movements, for epochs following one or more epochs of stage REM as defined in A, if the EEG continues to show low amplitude, mixed frequency activity without K complexes or sleep spindles and the chin EMG tone remains low.

  C. Stop scoring stage REM when one or more of the following occur: 1) There is a transition to stage Wake or N3; 2) An increase in chin EMG tone above the level of stage REM and criteria for stage N1 are met; 3) An arousal occurs followed by low amplitude, mixed frequency EEG and slow eye movements (score as stage N1; if no slow eye movements and chin EMG tone remains low, continue to score stage REM); 4) A major body movement followed by slow eye movements and low amplitude mixed frequency EEG without non-arousal associated K-complexes or sleep spindles (score the epoch following the major body movement as stage N1; if no slow eye movements and the EMG tone remains low, continue to score as stage REM); 5) One or more non-arousal associated K-complexes or sleep spindles are present in the first half of the epoch in the absence of rapid eye movements(score as stage N2).

Appendix B

Performance Measures

Several performance evaluation criterion were used in the evaluation of the proposed methods. The details of the performance criterion are defined as following. Considering the information of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN), the common performance measures are detailed in Table B.1.

In ASSC the classes are unbalanced, thus the common measures to calculate the accuracy and error are not suitable. Suppose a sleep versus wake scoring problem with probabilities of 0.8 and 0.2, respectively. If the classifier classifies all epochs as sleep, then the classification accuracy is 80%, despite it failed all wake epochs. Obviously, this measure is a poor indicator to calculate the performance. To deal with unbalanced classes, the balanced accuracy (BAC) measure, which treats both classes with equal importance, is used. The balanced accuracy is the average of the sensitivity and the specificity, obtained by:

$$BAC = 0.5 * (TP/(TP + FN) + TN/(TN + FP)). \tag{B.1}$$

TABLE B.1: The performance evaluation criterion.

Row	Performance Evaluation	Formula
1	TPR = sensitivity	TP/P = TP/(TP+FN)
2	TNR = specificity	TN/N = TN/(FP+TN)
3	FNR = 1-sensitivity	FN/P = FN/(TP+FN)
4	FPR = 1-specificity	FP/N = FP/(FP+TN)
5	Accuracy	(TP+TN)/(TP+TN+FP+FN)
6	Precision	TP/(TP+FP)
7	Recall	TP/(TP+FN)
8	F-Measure	2(Precision*Recall)/(Precision+Recall)= 2TP/(2TP+FN+FP)
9	ERR	Where FPR is equal to the FNR
10	AUC	(sensitivity+ specificity)/2

TPR: true positive rate, TNR: true negative rate, FPR: false positive rate and FNR: false negative rate

Equivalently, the balanced error rate is the average of the errors on each class. The balanced error rate is equal to: BER = (1-BAC).

Appendix C

Iris Recognition using Robust Localization and Nonsubsampled Contourlet Based Features

The conventional iris recognition methods do not perform well for the datasets where the eye image may contain nonideal data such as specular reflection, off-angle view, eyelid, eyelashes and other artifacts. This appendix surveys our contributions for a reliable iris recognition method using a new scale-, shift- and rotation-invariant feature-extraction method in time-frequency and spatial domains. In Section C.2, methodology and algorithm description of the proposed method is detailed. Indeed, a 2-level nonsubsampled contourlet transform (NSCT) was applied on the normalized iris images and a gray level co-occurrence matrix (GLCM) with 3 different orientations is computed on both spatial image and NSCT frequency subbands. Moreover, the effect of the occluded parts is reduced by performing an iris localization algorithm followed by a four regions of interest (ROI) selection. Next, the extracted feature set is transformed and normalized to reduce the effect of extreme values in the feature vector. Then, significant features for iris recognition were selected by a two-step method composed by a filtering stage and wrapper based selection. Finally, the selected feature set is classified using support vector machine (SVM). In Section C.3, the proposed iris identification method was evaluated on the public iris datasets CASIA Ver.1 and CASIA Ver.4-lamp showing a state-of-the-art performance.

C.1 Introduction

Iris recognition is a reliable and accurate biometric identification technology due to the uniqueness, aging invariant and noninvasive characteristics of iris. Moreover, this is a noncontact data acquisition technology.

Since, Flom and Safir [284] proposed the concept of iris recognition for first time, many research works on automatic iris recognition have been published. These approaches comprise, iris preprocessing and segmentation, iris code generation and finally, comparison and recognition [285]. An earlier automatic iris recognition method, based on multiscale Gabor wavelets and extracted phase information of iris textures, was proposed by Daugman [286]. Wildes [287] employed a gradient-based binary edge map and the Hough transform to detect the iris and pupil boundaries. Iris images were classified by using the normalized correlation. Recently, many other automatic iris recognition algorithms have been proposed which are based on the pioneered algorithms of Daugman [286] and Wildes [287]. Table C.1 summarizes the state-of the-art automatic iris recognition approaches. Preprocessing and segmentation generally consist on iris localization and iris normalization. For iris localization, which is the process of detecting the inner (iris/pupil) and the outer (iris/sclera) boundaries in the eye image, several techniques have been proposed, such as Integro-differential operator [286, 288] a combination of Hough transform and region-based active contours [289], and thresholding [290]. In iris normalization, most of the algorithms applied Daugman rubber sheet model [286, 288, 290–293]. Most of the methods performed well for ideal conditions in a very constrained environment [285, 286]. However, iris recognition under nonideal real-world conditions still presents many challenges not solved by those algorithms. A nonideal dataset of eye images may contain occlusions such as eyelids and eyelashes or low contrast, specular reflections, focus, and nonuniform illumination. Besides, the off-axis eye image (eye not oriented horizontally) that occurs frequently in real eye images is another common problem to overcome in iris recognition [294]. Recently, some methods have been proposed [295–298] to segment iris from nonideal eye images. Datasets CASIA Ver.3 and Ver.4 [299], UBIRIS Ver.2 [300] have been used to evaluate the proposed segmentation methods.

On the other hand, different methods have been applied to extract features from normalized iris images, such as approaches based on Gabor filters [286], Wavelet transforms [291–293, 301] Curvelet transforms [288] and 1-D circular profiles [302]. Even though, the wavelet transform is popular, powerful and familiar among the iris image processing techniques, it has its own limitations in capturing directional information such as smooth contours and the directional

TABLE C.1: State-of-the-art of IRIS recognition

Iris recognition approaches	Preprocessing and Segmentation		Feature extraction	Classification
	Iris localization	Iris normalization		
Daugman [286]	Integro-differential operator	Homogenous rubber sheet model	2D Gabor filters	Hamming distance
Wildes [287]	Image intensity gradient and Hough transform	Low pass Gaussian filter, spatial sub-sampling	Laplacian pyramid decomposition	Normalized correlation
He, Z. *et al.*[303]	Ada-boost cascade iris detector, Pulling and pushing elastic model	Daugman's rubber sheet model	Regional ordinal measure	Hamming distance
Farouk, R.M. [289]	Circular Hough transform, region-based active contours	-	Gabor wavelet	Elastic graph
Tsai, C. *et al.* [304]	Image intensity gradient and Fuzzy curve-tracing (FCT) algorithm	The nonlinear normalization	Gabor filter	Probabilistic fuzzy matching
Poursaberi and Araabi [290]	Extended-minima morphology operator with suitable threshold	Modified Daugman's rubber sheet model	Daubechies2 wavelet	Hamming distance
Roy,*et al.* [291, 292]	Level set methods with edge stopping function and energy minimization algorithm	Daugman's rubber sheet model	Daubechies wavelet	Adaptive asymmetrical SVM (AASVMs)
Belcher,C. *et al.*[298]	Rectangular images around the pupil	Daugman's rubber sheet model	Region-based SIFT features	Euclidean distance
Chen and chu [302]	2D wavelet filtering, Image intensity and Thresholding	Mapping to a fixed-size rectangular and partitioned normalized iris image into 3 and again into 2 regions	1-D circular profile	Probabilistic Neural Network (PNN)
Szewczyk *et al.* [293]	Filling the light source reflection, non-concentric circles based modelling of iris boundaries	Daugman's rubber sheet model	Reverse biorthogonal wavelet transform	Hamming distance

edges of the image. This problem is addressed by Contourlet transform (CT) [305]. In addition to multiscale and time-frequency localization properties of wavelets, CT offers directionality and anisotropy. A 4-level CT method for iris feature extraction is described in [306], in which normalized images are partitioned into multiscale and multi-directional subbands. The normalized energy of subbands are calculated as features to train a support vector machine (SVM) classifier. Due to downsampling and upsampling, the CT lacks shift-invariance. To overcome this limiting factor, Cunha *et al.* [307] proposed a shift-invariant version of CT designated nonsubsampled contourlet transform (NSCT).

Several methods for feature extraction, representing different aspects of the iris images, were reported [291, 308]. To reduce the computational cost and to improve the classification performance, a selection of the best discriminative features is highly desirable.

C.2 Proposed Approach

The proposed iris recognition method includes five major phases: iris preprocessing and segmentation, feature extraction, feature transformation and normalization, feature selection, and classification.

C.2.1 Iris Preprocessing and Segmentation

For the purpose of iris recognition, some parts of eye image such as eyelid, sclera, eyelash and pupil should be removed. In addition, even for iris of the same eye, the size may vary depending on camera-to-eye distance as well as light brightness. Therefore, the original eye image needs to be preprocessed to reduce the influence of the mentioned occlusions.

C.2.1.1 Localization

As shown in the Fig. C.1, to locate the inner (iris/pupil) and outer (iris/sclera) boundaries in an eye image, the following steps are performed: 1) reflection removal. 2) pupillary boundary detection. 3) limbic boundary detection. *Reflection removal:* Specular reflections (light spots in the eye image) can cause some problems in the localization process. As shown in Fig. C.2 (a-f), to localize the light source reflections, firstly the eye image is binarized using a thresholding technique (in the experiments, the threshold=190 was used). The binarized eye image is then

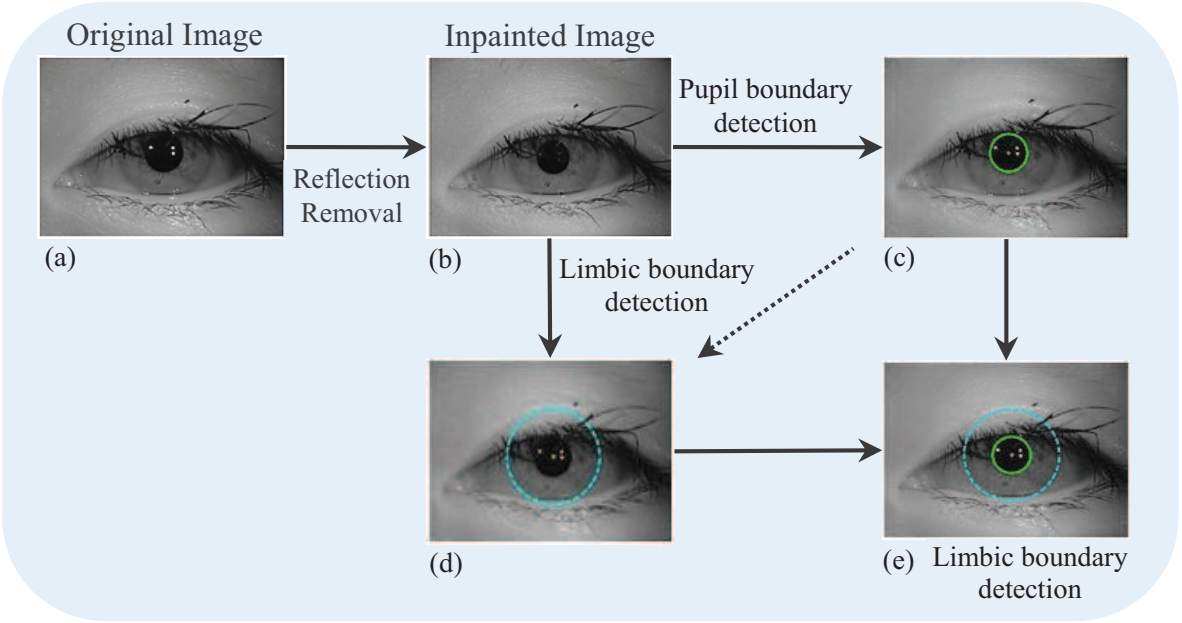

FIGURE C.1: Block diagram of the iris localization steps.

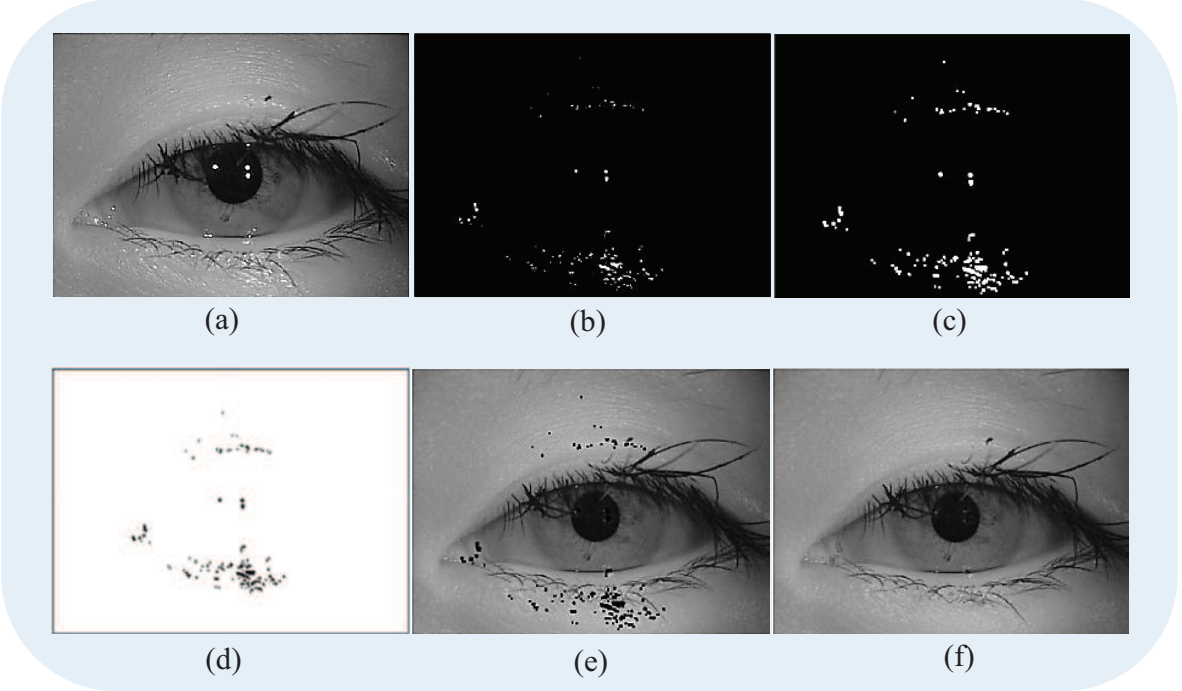

FIGURE C.2: Reflection removal steps; (a) the original eye image, (b) binarized eye image after applying the threshold, (c) dilated binarized eye image resulted from (b), (d) complement of image (c), (e) mask image resulted of applying (d) to the eye image, (f) inpainted image.

Algorithm 5: Reflection Removal(*EyeImage*)

1. Set threshold = 190.
2. ***If*** EyeImage (i, j)> threshold
   % i, j: index of each pixel of eye image.
   EyeImage(i,j) = 1.
   else
   EyeImage(i,j) = 0.
   [End of ***If*** structure.]
3. Mask=Dilate (*EyeImage*).
4. MaskedImage = EyeImage (1- Mask).
5. [Inpaint the reflections]
   (a) Find zero pixels in the MaskedImage.
   (b) ***While*** NumOfZeroPixels > 0 do:
      i. Set the average of 8 surrounding neighbours to the zero pixels in the MaskedImage.
      ii. Find the reminder of the zero pixels in the MaskedImageand Update the NumOfZeroPixels.
   [End of ***While*** structure.]

   [End of step 5.]

dilated, to consider all possible affected regions. Afterwards to fill the segmented reflection, the resulted mask is complemented and applied to the eye image for marking the reflections spots. Finally, the detected specular reflections are "inpainted" using the 8 surrounding neighbors (All steps are detailed in Algorithm 5).

Pupillary boundary detection: To detect the pupillary boundary, the inpainted eye image is first binarized (Fig. C.3(b)) using a threshold value, $M + 25$ [309] where M is a minimum fixed value of the inpainted image. In addition to the pupil, other dark regions of the eye image such as eyelashes fall below this threshold value. In order to eliminate the regions corresponding to the eyelashes a 2-D median filter with a 10×10-convolution mask is applied to the binary image. This reduces the number of candidate regions detected as a consequence of thresholding [309] (Fig. C.3(c)). The remaining regions in the median-filtered binary image are labeled and the region with the largest area and the smallest eccentricity is determined as

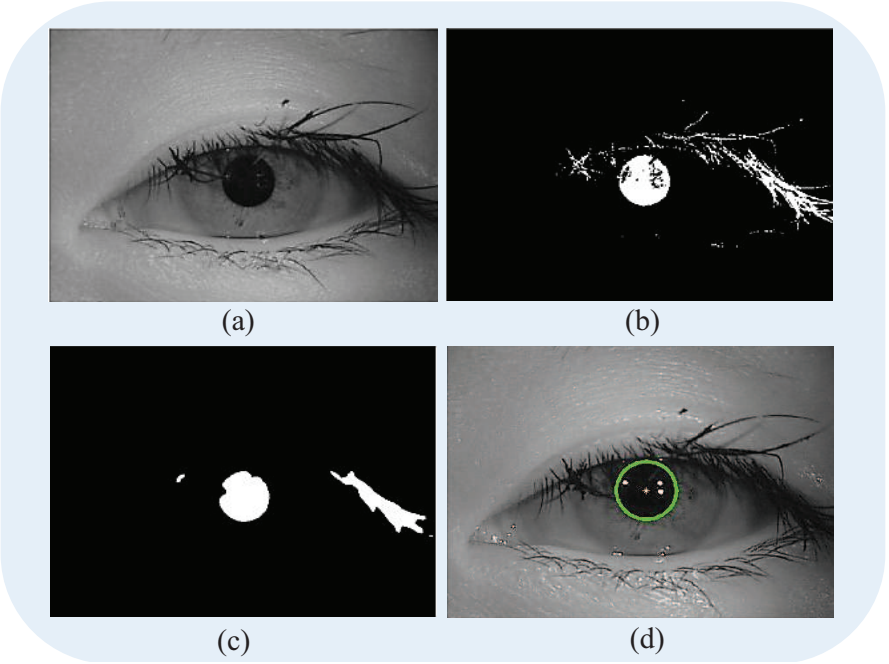

FIGURE C.3: Pupil boundary detection steps. (a) inpainted image, (b) binarized inpainted image, (c) smoothed image, (d) detected pupillary boundary.

Algorithm 6: Pupillary Boundary Detection(*InpaintedImage*)

1. Set threshold = min (*InpaintedImage*) + 25.
2. ***If*** InpaintedImage (i,j) < threshold *% i, j: index of each pixel of InpaintedImage.*
   InpaintedImage(i,j) = 1.
   else
   InpaintedImage(i,j) = 0.
   [End of ***If*** structure]
3. BinarizedPupilImage=Fill the InpaintedImageholes.
4. SmoothedImage = MedianFilter (*BinarizedPupilImage*, [10, 10]).
   % [10, 10] is window size for median filter
5. Label the exiting regions in the SmoothedImage.
6. Calculate Area and Eccentricity for each region.
7. Find the region with the largest Area and the smallest Eccentricity.
8. Calculate Centroid and Radiusof the Pupil Region.

pupil region. Finally, the pupil radius and centroid are calculated as follows:

$$pupilRadius = (\sqrt{4 \times A/\pi}) \Big/ 2 \tag{C.1}$$

$$(C_x, C_y) = (\int xdA \Big/ A, \int ydA \Big/ A) \tag{C.2}$$

where (C_x, C_y) denote the center coordinates of the pupil and A is the area of the pupil. All steps are detailed in Algorithm 6.

Algorithm 7: Limbic Boundary Detection(*InpaintedImage,CentroidofPupil*)

1. CannyEdgeImage = Canny (*InpaintedImage*).
2. AdjustedImage = AdjustedGamma (*CannyEdgeImage*).
3. NMSImage = NonMaximaSuppression (*AdjustedImage*).
4. FinalEdgeMap = HysteresisThreshold (*NMSImage*).
5. [IrisRadius, IrisCenterX, IrisCenterY] = CircularHoughTransform (*FinalEdgeMap*, *CentroidofPupil*).

Limbic boundary detection: As shown in Algorithm 7, before locating the outer boundary, a gamma threshold [310] is adjusted to the iris edge map(extracted by a Canny edge detector) to enhance the iris contrast. Then, the weak edge pixels are set to zero using non-maxima suppression; thus only the dominant edges are extracted. Finally, a hysteresis threshold is applied to the image. Having the pupil center coordinates, the radius and center coordinates of the iris boundary can be deduced using the circular Hough transform (Fig. C.4).

C.2.1.2 Regions of Interest Selection

As depicted in Fig. C.5, to disregard the iris regions occluded by the eyelid and eyelashes and to avoid loss of discriminative features, we adopt the method described in our previous work [30], in which four regions of interest (ROI) are selected:

1. Right side of the iris circle, a sector between angles -$\pi/4$ and $\pi/4$ with a radius equal to iris radius (Fig. C.5(a)).
2. Left side of the iris circle, a sector between angles $4\pi/5$ and $4\pi/3$ with a radius equal to iris radius (Fig. C.5(a)).

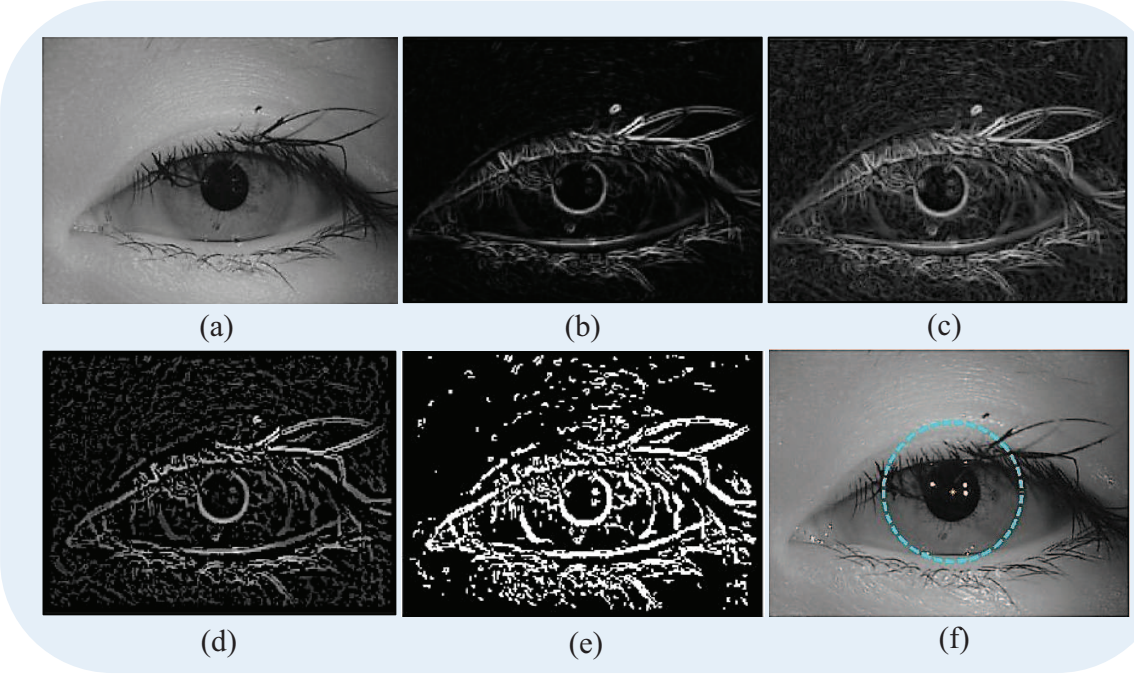

FIGURE C.4: Illustration of limbic boundary detection steps. (a) inpainted image, (b) result of applying canny edge detector, (c) result of applying gamma adjustment, (d) result of applying non-maxima suppression, (e) result of applying hysteresis thresholding, (f) result of applying circular hough transform on (e) and detected limbic boundary.

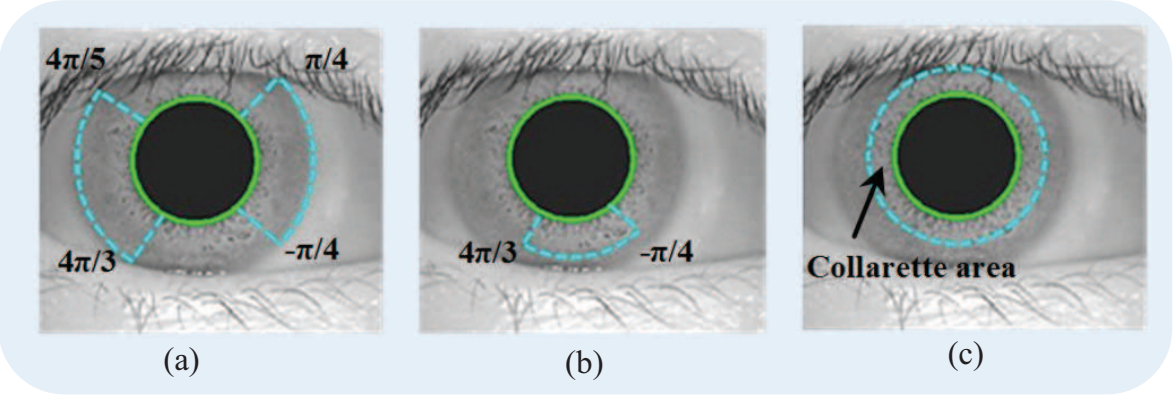

FIGURE C.5: Selected areas for normalization.

3. Bottom side of the iris circle, a sector between angles $4\pi/3$ and $-\pi/4$ with a radius of 1/2 of the iris radius (Fig. C.5(b)).

4. A disk around the pupil with a radius of 1/3 of the iris radius to cover the collarette area (Fig. C.5(c)).

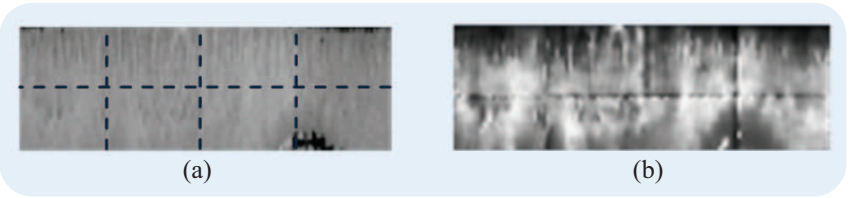

FIGURE C.6: Illustration of iris enhancement step; (a) tiled normalized image, (b) enhanced iris image resulted from histogram equalization and Wiener filtering.

C.2.1.3 Normalization and Enhancement

To compensate several external factors such as illumination variations and imaging distance, the partial iris images are normalized using "Daugman Rubber Sheet" model [286].
Since the original iris image has low contrast and may have non uniform illumination caused by the position of the light sources, some enhancements need to be applied. The histogram equalization is used to enhance the normalized iris images. The enhancement involves tessellating the normalized iris into 32×32 tiles (Fig. C.6(a)) and subjecting each tile to histogram equalization. Then the Wiener noise-removal filter is applied to the output of equalized histogram (Fig. C.6(b)).

C.2.2 Feature Extraction

A reliable iris recognition system should extract features that are invariant to scaling, shift and rotation. As we described in [30], the scale invariance is obtained by unwrapping the selected iris regions into four fixed size rectangles. To achieve shift invariance, the enhanced images are transformed into the frequency domain using the NSCT which is a shift-invariant transform and can capture the geometry of the iris texture. Finally, the GLCM is calculated on both spatial image and NSCT frequency subbands, which yields rotation invariance. The method is detailed in the following paragraphs.

C.2.2.1 Nonsubsampled Contourlet Transform

In contourlet transform, the Laplacian Pyramid (LP) is first used to capture point discontinuities, and then followed by a Directional Filter Bank (DFB) to link point discontinuities into linear structures [311]. The overall result is an image expansion using basic elements like contour segments, and thus called contourlet transform, which is implemented by a Pyramidal Directional Filter Bank (PDFB) [312]. The LP decomposition at each level generates a

downsampled lowpass version of the original image, and a difference between the original image and the prediction results in a bandpass image. As stated in [307] "*due to downsamplers and upsamplers present in both LP and DFB, contourlet transform is not shift-invariant*". To achieve the shift-invariance property, NSCT was proposed.
The NSCT is built upon nonsubsampled laplacian pyramids (NSLP) and nonsubsampled directional filter bank (NSDFB); thus, it is a fully shift-invariant, multiscale, and multidirection image decomposition that has a fast implementation.

C.2.2.2 Primary Features

The enhanced iris image is decomposed into 6 directions using NSDFB at 2 different scales. Next, some textural features are extracted from the spatial iris image and all the resultant NSCT frequency subbands. Textural features f_1-f_{22} mentioned in Table C.2 are computed on the basis of statistical distribution of pixels' intensity at a given position relative to others in a matrix of pixels called GLCM [308]. Since the GLCM is computed for different orientations, the rotation of the iris can be captured by one of the matrices. Feature extraction based on GLCM is a second-order statistic that can be employed to analyze an image as a texture. Although GLCM captures properties of a texture, it cannot be directly used for further analysis, such as the comparison of two textures; thus numeric features f_1-f_{22} which contain significant information about the textural characteristics are obtained from the GLCM in different directions [308], [313], and [314]. Moreover, numerical features f_{23}-f_{26} are calculated directly on NSCT frequency subbands and spatial iris image.

C.2.3 Feature Transformation and Normalization

The extracted features are transformed and normalized in order to reduce the influence of extreme values. The transformation methods applied to each feature are described in [3]. After a thorough experimental evaluation of each transform operator over extracted features, it was empirically verified that the best classification results were attained by applying $x_{ij} = 1/\sqrt{y_{ij}}$ where y_{ij} denotes the ij$_{th}$ element of a feature matrix $\boldsymbol{Y}$, and

$$\mathbf{X} = \{x_{ij}\}, \quad i = 1, 2, \ldots, N \quad j = 1, 2, \ldots, M \tag{C.3}$$

TABLE C.2: Textural features

Row	Feature name	Formula
f_1	Autocorrelation [313]	$\sum_j (ij)\, p(i,j)$
f_2	Contrast [307, 313]	$\sum_{n=0}^{N_{g-1}} n^2 \{\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j)\}$, cx
f_3	Correlation Haralock1991	$\sum_{i,j} \frac{(i-\mu_i)(j-\mu_j)p(i,j)}{\sigma_i \sigma_j}$
f_4	Correlation [307, 313]	$\frac{\sum_i \sum_j (ij)\, p\,(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y}$
f_5	Cluster Prominence [313]	$\sum_i \sum_j \{i+j-\mu_x-\mu_y\}^4 \times p(i,j)$
f_6	Cluster Shade [313]	$\sum_i \sum_j \{i+j-\mu_x-\mu_y\}^3 \times p(i,j)$
f_7	Dissimilarity [313]	$\sum_i \sum_j \lvert i-j\rvert .p(i,j)$
f_8	Energy [307, 313]	$\sum_i \sum_j p(i,j)^2$
f_9	Entropy [313]	$-\sum_i \sum_j p(i,j)\log(p\,(i,j))$
f_{10}	Homogeneity [313, 315]	$\sum_{i,j} \frac{p(i,j)}{1+\lvert i-j\rvert}$
f_{11}	Homogeneity [313]	$\sum_i \sum_j \frac{1}{1+(i-j)^2} p(i,j)$
f_{12}	Maximum probability [313]	$MAX\ p(i,j)$
f_{13}	Sum of squares Variance [307]	$\sum_i \sum_j (i-\mu)^2 p(i,j)$
f_{14}	Sum of average [307]	$\sum_{i=2}^{2N_g} i p_{x+y}(i)$
f_{15}	Sum of variance [307]	$\sum_{i=2}^{2N_g} (i-f_8)^2 p_{x+y}(i)$
f_{16}	Sum of entropy [307]	$-\sum_{i=2}^{2N_g} p_{x+y}\,(i) \log\,\{p_{x+y}\,(i)\}$
f_{17}	Difference variance [307]	$variance\ of\ p_{x-y}$
f_{18}	Difference entropy [307]	$-\sum_{i=0}^{N_{g-1}} p_{x-y}\,(i) \log\{p_{x-y}\,(i)\}$
f_{19}	Information measure of correlation1 [307]	$\frac{HXY - HXY1}{max\{HX, HY\}'}$ $HXY = -\sum_i \sum_j p(i,j)\log(p\,(i,j))$ $HXY1 = -\sum_i \sum_j p\,(i,j) \log\{p_x\,(i)\, p_y\,(i)\}$
f_{20}	Information measure of correlation2 [307]	$(1-\exp\,[-2.0\,(HXY2-HXY)]\,)^{1/2}$ $HXY = -\sum_i \sum_j p(i,j)\log p(i,j)$ $HXY2 = -\sum_i \sum_j p_x\,(i)\, p_y\,(j) \log\{p_x\,(i)\, p_y\,(j)\}$
f_{21}	Inverse difference normalized [314]	$\sum_{i,j=1}^{G} \frac{C_{ij}}{1+\lvert i-j\rvert^2/G^2}$
f_{22}	Inverse difference moment normalized [314]	$\sum_{i,j=1}^{G} \frac{C_{ij}}{1+(i-j)^2}$
f_{23}	Standard Deviation	$(\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\overline{x})^2)^{\frac{1}{2}}$ $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$
f_{24}	Mean	$\left(\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j)\right)/M$ $i+j=n$
f_{25}	Variance	$\sum_i \sum_j (i-\mu)^2 p(i,j)$
f_{26}	Energy of Fast Fourier-Transform	$Energy(FFT) = \sum_{i,j} p(i,j)^2$

(where N and M denote the number of samples and features, respectively) is the transformed feature matrix. Thereby this transform was adopted in the overall iris recognition system. To avoid features in greater numeric ranges dominating those in smaller numeric ranges, each feature of the transformed matrix $\mathbf{X}$ is independently normalized (scaled) by applying

$$\overline{x}_{ij} = x_{ij}/(max(\boldsymbol{x}_j) - min(\boldsymbol{x}_j)) \tag{C.4}$$

where $i = 1, 2, \ldots, N$, $j = 1, 2, \ldots, M$ and $\boldsymbol{x}_j$ is a vector of each independent feature [316].

C.2.4 Feature Selection

Selection of the features using well known automatic feature selectors is not accurate enough to get the best results; most of these feature selectors, select feature elements from all the feature-types which yield inaccurate selection. In order to obtain a more accurate selection and further reduce the number of extracted features, a new two-step feature selection process, which consists of filtering and a wrapper phases, is proposed. Filter based methods are in general faster than wrapper strategies. On the other hand, wrapper strategies are found to be more accurate [259]. In first step (as detailed in Algorithm 8), several feature-types (each feature-type consists of some feature elements) with the minimum redundancy are selected between the entire feature-types of Table C.2. As shown in Table C.3, in this step two prominent groups of the features are selected: 1) Group I composed by features f_{10}, f_{11}, f_{12}, f_{13}, f_{14} and f_{15}; and 2) Group II consists of features f_{21}, f_{22}, f_{23}, f_{24}, f_{25} and f_{26} (see Table C.2). As second step of feature selector, the minimal-redundancy and maximal-relevance (mRMR) [256] is used to select the most discriminative feature elements from these two groups of feature-types. Moreover, in the second step of feature selector, we compared the result of mRMR with sequential forward selection (SFS) [257], sequential backward selection (SBS) [257], sequential floating forward selection (SFFS) [258], sequential floating backward selection (SFBS) [258] and differential evolution based feature selection (DEFS) [259].

C.2.5 Classification

For the classification stage we used SVM [260]. Furthermore, k-nearest neighbor (KNN), naïve bayes (NB), and artificial neural network (ANN) are used to compare efficiency of the system.

TABLE C.3: Classification accuracy of each feature, some combination of two features, and best combinations resulted of Algorithm 8.

Mean accuracy for each feature (%)	Feature name																								
86.95	Autocorrelation	*																							
0.5	Contrast																								
0.5	Correlation:																								
0.5	Correlation																								
0.5	Cluster Prominence																								
0.5	Cluster Shade																								
0.5	Dissimilarity																								
88.65	Energy		*																						
0.5	Entropy																								
92.5	Homogeneity:matlab			*																					
92.4	Homogeneity				*																				
84.05	Maximum					*																			
83.45	Sum of squares Variance				*		*																		
81.1	Sum of average			*				*																	
85.4	Sum of variance								*																
0.5	Sum of entropy																								
0.5	Difference variance																								
0.5	Difference probability																								
0.5	Information measure of correlation1																								
0.5	Information measure of correlation2																								
93.5	Inverse difference normalized	*						*	*		*														*
94	Inverse difference moment normalized		*			*				*		*													*
75.05	Standard deviation										*		*	*	*										*
10.25	Mean															*									*
49.45	variance																*								*
68.2	Energy of Fast Fourier transform														*	*	*								*
	Mean accuracy (%)	96.25	95.8	96.35	96.4	94.25	96.9	96.45	96.15	96.4	96.35	84.08	38.29	68.7	95.65	96.45									
		The best combination of each 2 features													best combination of features (group1)	best combination of features (group2)									

Algorithm 8: Feature Selection (*FeatureMatrix*)

1. FeatureMatrix = $\{F_1, F_2, \ldots, F_N\}$, $F_i = \{ y_1, y_2, \ldots, y_M\}$
   % N: number of feature type, M: number of element
2. Initialize total feature set, total number of feature type and accuracy:
   (a) Set TotalFeatureSet = { }, d= 1.
   (b) Set AccuracySet = { }.
3. ***While***(d <= N) do:
   (a) Initialize FeatureSet and number of feature type:
      i. Set FeatureSet = { }, Set k=d+1.
   (b) Add F_d to FeatureSet.
   (c) Calculate accuracy of FeatureSet.
   (d) Add accuracy to AccuracySet and FeatureSet to TotalFeatureSet.
   (e) ***While*** (k <= N)
      i. Add F_k to FeatureSet.
      ii. Calculate accuracy of FeatureSet.
      iii. Add accuracy to AccuracySet and FeatureSet to TotalFeatureSet.
      iv. k= k+1.
   [End of ***While*** structure]
   (f) Set d=d+1.

[End of ***While*** structure]

C.3 Performance Assessment

To assess the performance of the proposed algorithm, several experiments were conducted using different publicly available datasets. All of the experiments were carried out in identification mode. The features of a test iris image were compared with the features of whole dataset. Left eye images of the CASIA dataset Ver.1 and Ver.4-lamp were used, which are popular iris datasets and widely adopted to evaluate the iris recognition system [299]. CASIA Ver.1 contains a total of 756 iris images from 108 subjects, in which the images were captured in two sessions, with at least one month interval. CASIA Ver.4-lamp was collected in one session using a hand-held iris sensor; a lamp was turned on/off close to the subject to make different illumination conditions. It contains 16213 iris images from 411 subjects. As stated

in a note of CASIA Ver.4-lamp [299] "*Elastic deformation of iris texture due to pupil expansion and contraction under different illumination conditions is one of the most common and challenging issues in the iris recognition*". CASIA Ver.4-lamp offers eye images in nonideal conditions, providing suitable data to assess the effects of iris image normalization and robust iris feature representation.

In our experiments, a two-level NSCT decomposition was adopted with 2 and 4 directions for each pyramidal level, respectively. Three GLCMs were calculated on all NSCT frequency subbands and the spatial image both in 0^o, 90^o and 135^o. The normalized iris images were decomposed by the NSPDFB. We have selected "pyrexc" and "pkva" as NSLP and NSDFB filter in PDFB decomposition [307] given their superior performance assessed empirically. SVM-KM [317] toolbox with Gaussian kernel was used in the classification phase. The Gaussian kernel degree and C parameters were set to 6 and 100 respectively as they produced the best empirical results. Experiments were carried out over 2000 images of 200 randomly selected classes, with 10 images per class and 756 images of 108 classes for CASIA Ver.4 and Ver.1, respectively. In order to verify reliability of the results, all the assessments were determined by leave-one out cross-validation (LOOCV). Moreover, to characterize the performance of the proposed method some well-known measures such as accuracy, area under curve (AUC), the equal-error rate (EER), sensitivity, specificity and F-measure were used. In particular, F-measure or balanced F-score is a weighted average of *precision* and *recall* where *precision* is the fraction of retrieved instances that are relevant and *recall*is the fraction of relevant instances that are retrieved.

C.3.1 Evaluation of Proposed Scheme for Iris Localization and Region of Interest Selection

To validate the performance of the proposed scheme for localization, we applied the method to the eye images with different occlusions and artifacts such as eyelids, eyelashes obstruction, specular reflections, contrast changes, non-uniform illumination, rotation and scale. As illustrated in Fig. C.7(a-i), the proposed method performs well despite of the artifacts, and it can localize the inner and the outer boundaries accurately. However, we observed that, it does not properly localize the outer boundaries due to the low contrast between the iris and sclera (see Fig. C.7(h)). To alleviate the loss of significant data, four iris ROIs were selected in our segmentation method. The detection error trade-off (DET) curves in Fig. C.8 show the comparison of different iris localization approaches on the CASIA Ver.4-lamp. Each

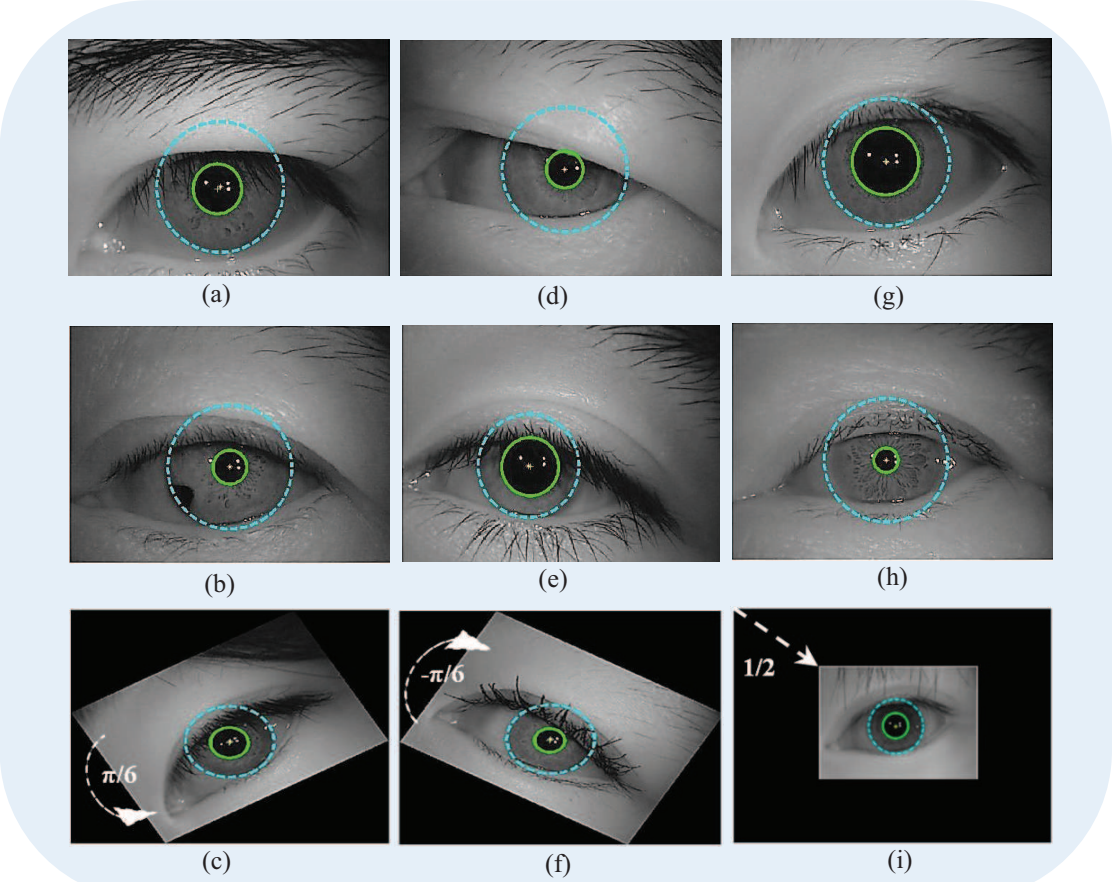

FIGURE C.7: Illustration of some randomly selected iris segmentation results for CASIA Ver.4-lamp; (a), (b), and (c) have some artifacts; moreover, (c) shows robustness of segmentation method to left rotation; (d) and (e) suffer from occlusion; (f) shows robustness to right rotation and suffers from makeup; The pupils in (g) and (h) are bigger and smaller than the normal size, respectively; (i) example of high amounts of blur and shows robustness to scaling.

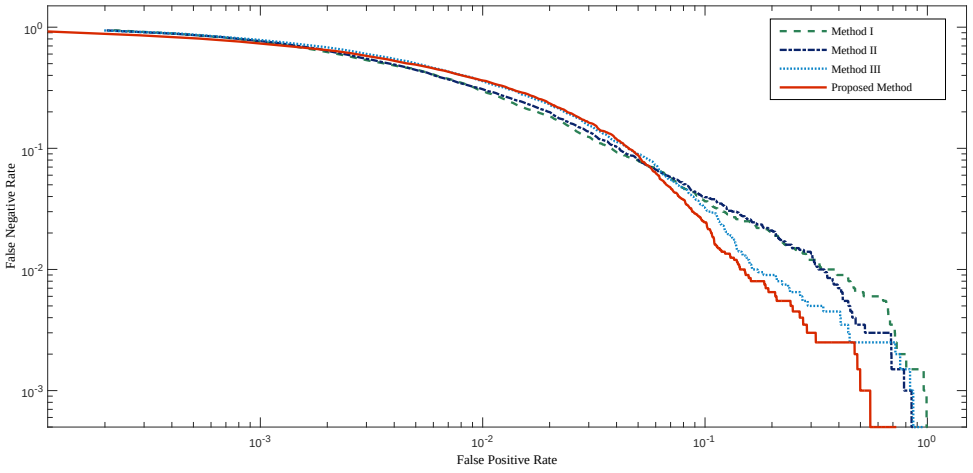

FIGURE C.8: The DET curves for comparing of different iris ROI in the localization process over the CASIAVer.4-lamp. The parameters of method I are $\Theta = (0, 2\pi)$, r = IrisR, method II $\Theta = (0, 2\pi)$, $r = 1/3 \times IrisR$ and method III are $\Theta_{Left} = (3\pi/4, 5\pi/4)$, $\Theta_{Right} = (-\pi/4, \pi/4)$, r = IrisR.

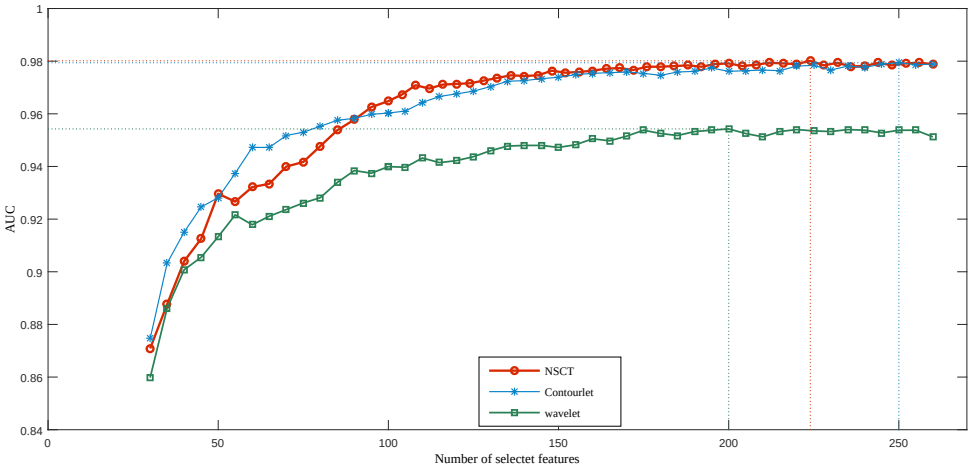

FIGURE C.9: Comparison between the AUC curves of the proposed method with NSCT, contourlet, and wavelet transforms on CASIA V.4-lamp.

curve is denoted by symbols r, Θ which represent normalized polar coordinates. Four cases are considered: 1) $\Theta = (0, 2\pi), r = IrisR$ that corresponds to a disk around the iris with iris radius, which covers the whole iris region; 2) $\Theta = (0, 2\pi)$, $r = 1/3 \times IrisR$ that corresponds to a disk around the iris with 1/3 iris radius, similar to Fig. C.5(c); 3) $\Theta_{Left} = (3\pi/4, 5\pi/4)$, $\Theta_{Right} = (-\pi/4, \pi/4), r = IrisR$ that refers to a state similar to Fig. C.5(a); and 4) our proposed method detailed in Section C.2.1.2.

The results shown in Fig. C.8 illustrate the superior performance of the proposed method over the other mentioned approaches.

C.3.2 Performance Assessment of Using Different Time-Frequency Transforms

This section is devoted to analysis the impact of different time-frequency transforms (wavelet, contourlet and NSCT), applied in feature extraction, in the overall iris recognition performance. According to the AUC curves of Fig. C.9, F-measure and average accuracy values shown in Table C.4, NSCT provided the best results, which may be correlated to its redundant structure. It gives the highest matching performance (0.9801) with the lowest number of features (224). However, for wavelet transforms the best AUC value of 0.9543 was obtained with 200 features; this AUC is lower than in the case of using NSCT. For the contourlet transform, the best average AUC of 0.9794 was attained with 250 features, which in comparison

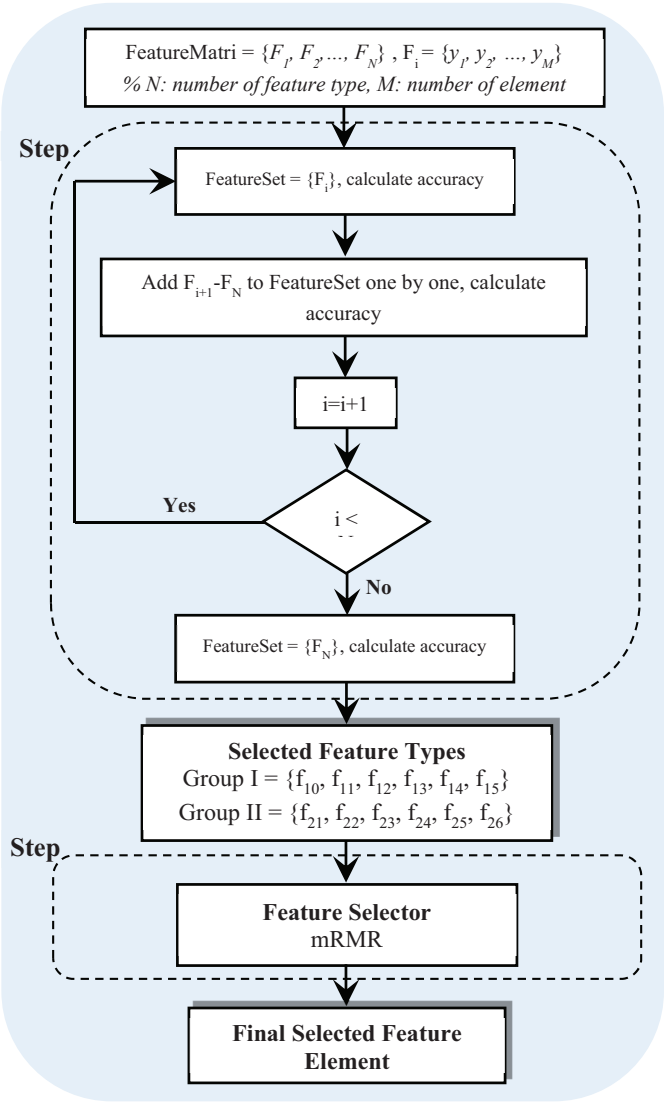

FIGURE C.10: A sample of the two-step feature selector: step1: selection of two prominent groups of features; step2: selection of feature element.

with NSCT-based feature extraction has a higher number of features and lower accuracy and F-measure values.

C.3.3 Evaluation of the Features Importance

To account with the high dimensionality problem in iris recognition, the proposed two-step feature selection method was used (Fig. C.10). As shown in Table C.3, after analysis of different combinations of features, two features groups were selected using Algorithm 8: 1) Group I composed by features f_{10}, f_{11}, f_{12}, f_{13}, f_{14} and f_{15}(see Table C.2); 2) Group II composed by features f_{21}, f_{22}, f_{23}, f_{24}, f_{25} and f_{26} (see Table C.2). Starting with 2240 extracted features from the four ROIs, a total of 896 elements (features of groups I, II) were selected in the

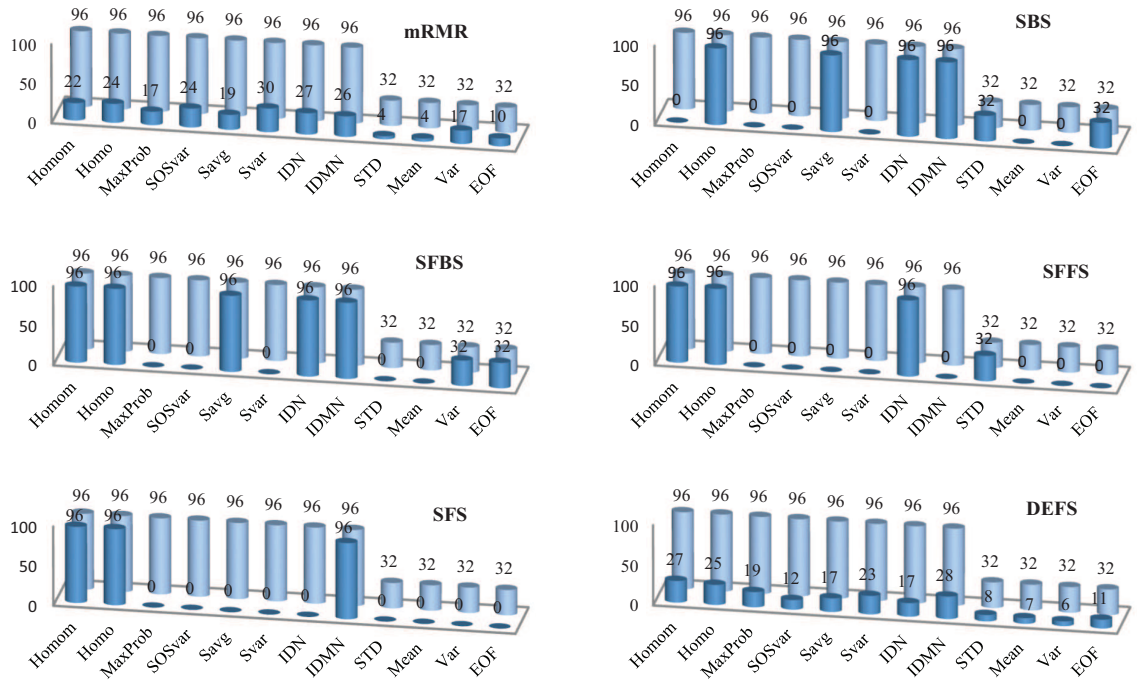

FIGURE C.11: Performance of different feature selection methods. Dark blue: selected features, light blue: total features. Homom: Homogeneity: matlab, Homo: Homogeneity, MaxProb: Maximum probability, SOSvar: Sum of squares Variance, Savg: Sum of average, Svar: Sum of variance, IDN: Inverse difference normalized, IDMN: Inverse difference moment normalized, STD: Standard deviation, Mean, Var: variance and EOF: Energy of Fast Fourier transform.

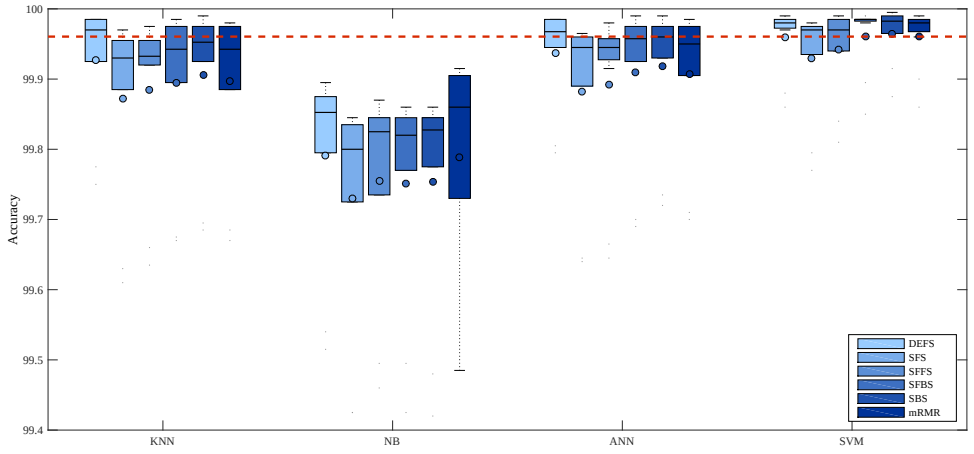

FIGURE C.12: Accuracy of the iris recognition, corresponding to 6 feature selectors (DEFS, SFS, SBS, SFBS, SFFS and mRMR,) and 4 classifiers (KNN, NB, ANN and SVM).

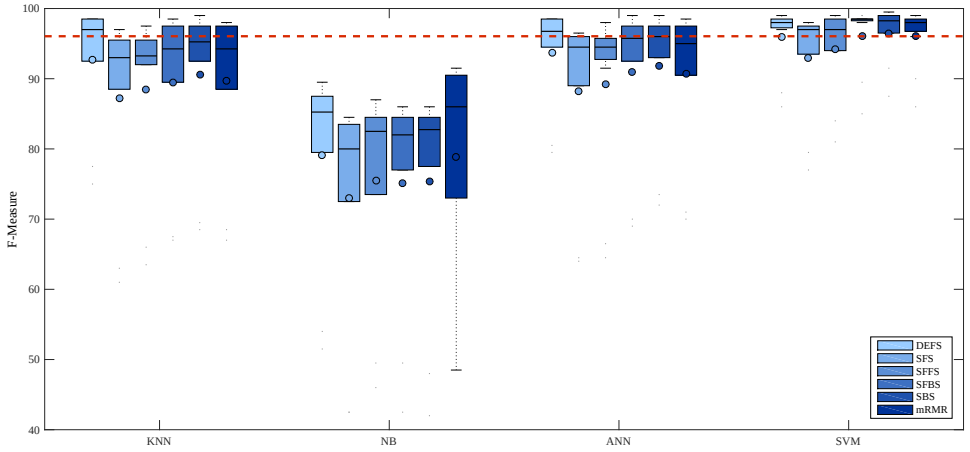

FIGURE C.13: F-measure of the iris recognition method, corresponding to 6 feature selectors (DEFS, SFS, SBS, SFBS, SFFS and mRMR,) and 4 classifiers (KNN, NB, ANN and SVM).

TABLE C.4: Comparison of feature extraction method on using different time-frequency transforms.

Feature Extraction Methods	Number of Features	Mean Accuracy	Mean Sensitivity	Mean Specificity	Mean Equal Error Rate	Mean F-Measure	Max Accuracy
NSCT	224	0.9996	0.9605	0.9998	1.98	96.05	0.9999
Contourlet	250	0.9995	0.9590	0.9998	2.06	95.90	0.9999
Wavelet	200	0.9990	0.9090	0.9995	4.57	90.90	0.9998

first step of the feature selector. Next, in second step of the feature selection strategy, these two groups of features were fed into the different feature selectors mentioned in section C.2.4. Fig. C.11 shows the proportion of the selected features from entire features resulted of the first step. It is shown that, mRMR was able to select a subset of 224 features that contained the discriminant information that gave lower EER for recognition. However, as illustrated in Fig. C.10, the highest average accuracy, performed with a combination of SBS and SVM, which in comparison with mRMR and SVM has a higher number of features. In fact, SBS selected 448 features, which is twice of selected features by mRMR.

C.3.4 Performance Evaluation of the Proposed Scheme

Fig. C.12 - C.13 compares the performance obtained by the proposed method with different combinations of mentioned feature selector/classifiers in Sections C.2.4 and C.2.5. In the experiments, four different types of classifiers were considered: NB, KNN, ANN and SVM classifiers. They are capable of handling large-scale classification problems. Moreover, six of the best feature selection approaches were used; DEFS, mRMR, and sequential methods (SBS,

TABLE C.5: Comparison with other methods for CASIA Ver.1, Ver.3-lamp and Ver.4-lamp (the results are taken from the published works.)

Method	Accuracy(%)		
	CASIA Ver.1	CASIA Ver.3-lamp	CASIA Ver.4-lamp
Daugman [286, 318]	100	96	-
Masek [310]	-	79.02	-
Basit [319]	98.94	-	-
Chen et al.[302]	99.35	-	-
Jan et al.[285]	100	98	-
Ibrahim et al.[320]	99.90	98.28	-
Khalighi et al.[30]	98.28	-	99.65
Proposed method, LOOCV (mean accuracy)	99.97	-	99.96
Proposed method, LOOCV(max accuracy)	100	-	99.99

SFBS, SFFS and SFS). The results are expressed in terms of box-whisker plots showing the average, median, the first and third quartile values of the accuracies and F-measures. The horizontal lines outside each box identify the upper and lower whiskers, and dot points denote the outliers. According to the results shown in

Fig. C.12 - C.13, the proposed combination of mRMR and SVM outperformed the others (accuracy of 0.9996 with 224 features). Although, the highest accuracy (0.9997) was attained with a combination of SBS and SVM it was obtained with the cost of requiring a higher number of features (mRMR=224, SBS=448). Moreover, some of the other combinations (e.g. combination of DEFS and ANN) also attained acceptable results.

Indeed, as shown in Fig. C.12 - C.13, SVM classifiers present the lowest interquartile of accuracies, sensitivities and specificities.

C.3.5 Comparison with State-of-the-art Methods

Table C.5 summarizes results of existing state-of-the-art iris recognition methods, tested at least on one of the following data sets: CASIA Ver.1, Ver.3-lamp and Ver.4-lamp. Regarding CASIA Ver.1, some accuracy results

on Table C.5 are higher than the average accuracy value of our method[1]. Considering that there are no reported performance results based on CASIA Ver. 4-lamp, and due to the similarity of Ver.3-lamp and Ver.4-lamp [299] wecompared our results with Ver.3-lamp. The proposed method attains better results than our previous work [30] with a lower number of features. Moreover, it performs at the state-of-the-art as can be observed from the results in Table C.5.

[1]Our reported results were obtained using the LOOCV method in the testing process.

C.4 Conclusion

In this work a new iris recognition method based on NSCT and GLCM, was proposed. This method has some advantages over other approaches. Firstly, the proposed iris localization algorithm is reliable performs well under nonconstrained conditions such as rotation, scale, and illumination conditions existing in CASIA Ver.4-lamp (see Fig. C.7). Secondly, some of the summarized works just used the upper and/or lower part of the iris image to remove the occluded regions by the eyelid and eyelashes, which results in loss of significant data. The proposed method selects four ROIs to make use of the most significant data in the iris texture. Thirdly, the extracted features are invariant to scaling, shift and rotation, which are some of the most important properties in iris recognition. Fourthly, to reduce the effect of extreme values in the feature matrix, the extracted feature set are transformed and normalized which improved the recognition rate. A two-step feature selection process that consists on a filtering and a wrapper phases was proposed. Moreover, we inferred from the proposed feature selector, that features homogeneity, inverse difference normalized, inverse difference moment normalized, sum of variance, sum of squares variance, sum of average, maximum probability, standard deviation, energy of fast Fourier transform and a their combinations are the best features for iris recognition problem. Finally, to estimate the accuracy of the proposed method LOOCV was used. The obtained average accuracies on CASIA Ver.4-lamp and Ver.1 were 99.96 %, and 99.97 % respectively.

Bibliography

[1] A. Martınez and R. Benavente, "The AR face database," *Rapport technique*, vol. 24, 1998.

[2] J. M. Plonsey and Robert, *Bioelectromagnetism - Principles and Applications of Bioelectric and Biomagnetic Fields*. New York: Oxford University Press, 1995.

[3] G. Becq, S. Charbonnier, F. Chapotot, A. Buguet, L. Bourdon, and P. Baconnier, "Comparison between five classifiers for automatic scoring of human sleep recordings," in *Studies in Computational Intelligence (SCI)*, p. 113–127, Springer, 2005.

[4] M. Ronzhina, O. Janoušek, J. Kolářová, M. Nováková, P. Honzík, and I. Provazník, "Sleep scoring using artificial neural networks," *Sleep medicine reviews*, vol. 16, no. 3, pp. 251–263, 2012.

[5] L. Zoubek, S. Charbonnier, S. Lesecq, A. Buguet, and F. Chapotot, "Feature selection for sleep/wake stages classification using data driven methods," *Biomedical Signal Processing and Control*, vol. 2, no. 3, pp. 171–179, 2007.

[6] M. Davey, "Sleep disorders: A clinical textbook," *Journal of Paediatrics and Child Health*, vol. 43, no. 1-2, pp. 96–97, 2007.

[7] Z. Cashero, *Comparison of EEG preprocessing methods to improve the performance of the P300 speller*. PhD thesis, Colorado State University. Libraries, 2007.

[8] V. C. F. Helland, A. Gapelyuk, A. Suhrbier, M. Riedl, T. Penzel, J. Kurths, and N. Wessel, "Investigation of an automatic sleep stage classification by means of multiscorer hypnogram," *Methods of Information in Medicine*, vol. 49, no. 5, pp. 467–472, 2010.

[9] T. Lajnef, S. Chaibi, P. Ruby, P.-E. Aguera, J.-B. Eichenlaub, M. Samet, A. Kachouri, and K. Jerbi, "Learning machines and sleeping brains: Automatic sleep stage classification using decision-tree multi-class support vector machines," *Journal of Neuroscience Methods*, 2015.

[10] C. O'Reilly, N. Gosselin, J. Carrier, and T. Nielsen, "Montreal archive of sleep studies: an open-access resource for instrument benchmarking and exploratory research," *Journal of Sleep Research*, vol. 23, no. 6, pp. 628–635, 2014.

[11] V. Bajaj and R. B. Pachori, "Automatic classification of sleep stages based on the time-frequency image of EEG signals," *Computer Methods and Programs in Biomedicine*, vol. 112, no. 3, pp. 320–328, 2013.

[12] L. Fraiwan, K. Lweesy, N. Khasawneh, H. Wenz, and H. Dickhaus, "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*, 2011.

[13] D. Alvarez-Estevez and V. Moret-Bonillo, "Identification of electroencephalographic arousals in multichannel sleep recordings," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 1, pp. 54–63, 2011.

[14] J. Rafiee, M. Rafiee, N. Prause, and M. Schoen, "Biorobotics: Optimized biosignal classification using mother wavelet matrix," in *2009 IEEE 35th Annual Northeast Bioengineering Conference*, 2009.

[15] W. Karlen and D. Floreano, "Adaptive sleep-wake discrimination for wearable devices," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 4, pp. 920–926, 2011.

[16] Y. Li, H. Kambara, Y. Koike, and M. Sugiyama, "Application of covariate shift adaptation techniques in brain-computer interfaces," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 6, pp. 1318–1324, 2010.

[17] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function," *Journal of Statistical Planning and Inference*, vol. 90, no. 2, pp. 227–244, 2000.

[18] S. Khalighi, T. Sousa, D. Oliveira, G. Pires, and U. Nunes, "Efficient feature selection for sleep staging based on maximal overlap discrete wavelet transform and svm," *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 3306–3309, 2011.

[19] T. Sousa, D. Oliveria, S. Khalighi, G. Pires, and U. Nunes, "Neurophysiologic and statistical analysis of failures in automatic sleep stage classification," in *International Conference on bio-inspired systems and signal processing (Biosignals 2012)*, (Vilamoura, Portugal), 2012.

[20] S. Khalighi, T. Sousa, G. Pires, and U. Nunes, "Automatic sleep staging: A computer assisted approach for optimal combination of features and polysomnographic channels," *Expert Systems with Applications*, vol. 40, no. 17, pp. 7046–7059, 2013.

[21] L. Fraiwan, K. Lweesy, N. Khasawneh, M. Fraiwan, H. Wenz, and H. Dickhaus, "Classification of sleep stages using multi-wavelet time frequency entropy and LDA," *Methods of Information in Medicine*, vol. 49, no. 3, pp. 230–237, 2010.

[22] W. C. Tang, S. W. Lu, C. M. Tsai, C. Y. Kao, and H. H. Lee, "Harmonic parameters with HHT and wavelet transform for automatic sleep stages scoring," *Proceedings of World Academy of Science, Engineering and Technology, Vol 22*, vol. 22, pp. 414–417, 2007.

[23] M. Adnane, Z. W. Jiang, and Z. H. Yan, "Sleep-wake stages classification and sleep efficiency estimation using single-lead electrocardiogram," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1401–1413, 2012.

[24] T. Sousa, A. Cruz, S. Khalighi, G. Pires, and U. Nunes, "A two-step automatic sleep stage classification method with dubious range detection," *Computers in Biology and Medicine*, 2015.

[25] S. Khalighi, B. Ribeiro, and U. Nunes, "Importance weighted import vector machine for unsupervised domain adaptation," *submitted to IEEE Transactions on Cybernetics, June 2015*, p. under review, 2016.

[26] T. Kanamori, S. Hido, and M. Sugiyama, "A least-squares approach to direct importance estimation," *The Journal of Machine Learning Research*, vol. 10, pp. 1391–1445, 2009.

[27] S. Khalighi, T. Sousa, J. M. Santos, and U. Nunes, "ISRUC-Sleep: A comprehensive public dataset for sleep researchers," *Computer methods and programs in biomedicine*, 2015.

[28] S. Khalighi, T. Sousa, and U. Nunes, "Adaptive automatic sleep stage classification under covariate shift," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, vol. 2012, pp. 2259–62, 2012.

[29] S. Khalighi, F. Pak, P. Tirdad, and U. Nunes, "Iris recognition using robust localization and nonsubsampled contourlet based features," *Journal of Signal Processing Systems*, pp. 1–18, 2014.

[30] S. Khalighi, P. Tirdad, F. Pak, and U. Nunes, "Shift and rotation invariant iris feature extraction based on non-subsampled contourlet transform and GLCM.," in *ICPRAM (2)*, pp. 470–475, 2012.

[31] M. Sugiyama and M. Kawanabe, *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*. MIT Press, 2012.

[32] V. N. Vapnik and A. J. Chervonenkis, "Theory of pattern recognition," *Nauka*, 1974.

[33] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," tech. rep., Purdue Univ.; School of Electrical Engineering.; West Lafayette, IN, United States, 1990.

[34] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *Information Theory, IEEE Transactions on*, vol. 13, no. 1, pp. 21–27, 1967.

[35] C. Cortes and V. Vapnik, "Support vector machine," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[36] V. Vapnik, *The nature of statistical learning theory*. Springer Science & Business Media, 2013.

[37] J. Neter, M. H. Kutner, C. J. Nachtsheim, and W. Wasserman, *Applied linear statistical models*, vol. 4. Irwin Chicago, 1996.

[38] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*, vol. 1. Springer series in statistics Springer, Berlin, 2001.

[39] C. E. Rasmussen, *Gaussian processes for machine learning*. Citeseer, 2006.

[40] I. Jolliffe, *Principal component analysis*. Wiley Online Library, 2002.

[41] K. Q. Weinberger and L. K. Saul, "Unsupervised learning of image manifolds by semidefinite programming," *International Journal of Computer Vision*, vol. 70, no. 1, pp. 77–90, 2006.

[42] G. McLachlan and D. Peel, *Finite mixture models*. John Wiley & Sons, 2004.

[43] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, no. 14, pp. 281–297, Oakland, CA, USA., 1967.

[44] A. Y. Ng, M. I. Jordan, and Y. Weiss, "on spectral clustering: Analysis and an algorithm," *in Advances in neural information processing systems*, vol. 14, 2002.

[45] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains," *The annals of mathematical statistics*, vol. 41, no. 1, pp. 164–171, 1970.

[46] O. Chapelle, B. Schölkopf, A. Zien, *et al.*, *Semi-supervised learning*. MIT press Cambridge, 2006.

[47] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using EM," *Machine learning*, vol. 39, no. 2-3, pp. 103–134, 2000.

[48] M. Belkin and P. Niyogi, "Semi-supervised learning on Riemannian manifolds," *Machine learning*, vol. 56, no. 1-3, pp. 209–239, 2004.

[49] A. B. Goldberg, X. Zhu, A. Singh, Z. Xu, and R. Nowak, "Multi-manifold semi-supervised learning," in *AISTATS*, pp. 169–176, 2009.

[50] T. Joachims, "Transductive inference for text classification using support vector machines," in *ICML*, vol. 99, pp. 200–209, 1999.

[51] O. Chapelle and A. Zien, "Semi-supervised classification by low density separation," in *Tenth International Workshop on Artificial Intelligence and Statistics (AISTATS)*, pp. 57–64, 2005.

[52] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," in *Advances in neural information processing systems*, pp. 529–536, 2004.

[53] A. Blum and S. Chawla, "Learning from labeled and unlabeled data using graph mincuts," in *Proceedings of the Eighteenth international Conference on Machine Learning. C. E. Brodley and A. P. Danyluk, Eds*, pp. 19–26, 2001.

[54] O. Chapelle, J. Weston, and B. Schölkopf, "Cluster kernels for semi-supervised learning," in *Advances in neural information processing systems*, pp. 585–592, 2002.

[55] X. Zhu, Z. Ghahramani, J. Lafferty, *et al.*, "Semi-supervised learning using gaussian fields and harmonic functions," in *ICML*, vol. 3, pp. 912–919, 2003.

[56] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, pp. 189–196, Association for Computational Linguistics, 1995.

[57] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of the eleventh annual conference on Computational learning theory,ser. COLT' 98*, Series Combining labeled and unlabeled data with co-training, (New York, NY, USA), p. 92–100, ACM, 1998.

[58] B. Settles, "Active learning literature survey," *University of Wisconsin, Madison*, vol. 52, no. 55-66, p. 11, 2010.

[59] R. S. Sutton and A. G. Barto, *Introduction to reinforcement learning*, vol. 135. MIT Press Cambridge, 1998.

[60] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[61] J. Jiang, "A literature survey on domain adaptation of statistical classifiers," tech. rep., URL: http://sifaka. cs. uiuc. edu/jiang4/domainadaptation/survey, 2008.

[62] X. Wang and J. Schneider, "Flexible transfer learning under support and model shift," in *Advances in Neural Information Processing Systems*, pp. 1898–1906, 2014.

[63] R. Klinkenberg, "Learning drifting concepts: Example selection vs. example weighting," *Intelligent Data Analysis*, vol. 8, no. 3, pp. 281–300, 2004.

[64] J. G. Moreno-Torres, T. Raeder, R. Alaiz-RodríGuez, N. V. Chawla, and F. Herrera, "A unifying view on dataset shift in classification," *Pattern Recognition*, vol. 45, no. 1, pp. 521–530, 2012.

[65] T. Liu and D. Tao, "Classification with noisy labels by importance reweighting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, pp. 447–461, March 2016.

[66] H. Wang, W. Fan, P. S. Yu, and J. Han, "Mining concept-drifting data streams using ensemble classifiers," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 226–235, ACM, 2003.

[67] K. Zhang, K. Muandet, Z. Wang, *et al.*, "Domain adaptation under target and conditional shift," in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pp. 819–827, 2013.

[68] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intelligent Data Analysis*, vol. 6, no. 5, p. 429–450, 2002.

[69] J. J. Heckman, "Sample selection bias as a specification error," *Econometrica*, vol. 47, no. 1, pp. 153–161, 1979.

[70] B. Zadrozny, "Learning and evaluating classifiers under sample selection bias," in *Proceedings of the 21th Annual International Conference on Machine Learning*, Series Learning and evaluating classifiers under sample selection bias, (Banff, Canada), p. 114–121, 2004.

[71] H. D. III, "Frustratingly easy domain adaptation," in *Proceedings of the 45th Meeting of the Association for Computational Linguistics*, (Prague, Czech Republic), Association for Computational Linguistics, 2007.

[72] A. Arnold, R. Nallapati, and W. W. Cohen, "A comparative study of methods for transductive transfer learning," in *Data Mining Workshops, 2007. ICDM Workshops 2007. Seventh IEEE International Conference*, Series A comparative study of methods for transductive transfer learning, 2007.

[73] M. T. Bahadori, Y. Liu, and D. Zhang, "Learning with minimum supervision: A general framework for transductive transfer learning," in *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pp. 61–70, IEEE, 2011.

[74] M. Rohrbach, M. Stark, and B. Schiele, "Evaluating knowledge transfer and zero-shot learning in a large-scale setting," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1641–1648, IEEE, 2011.

[75] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 951–958, IEEE, 2009.

[76] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, "Self-taught learning: transfer learning from unlabeled data," in *ICML 07 Proceedings of the 24th international conference on Machine learning*, (New York, NY, USA), p. 759–766, 2007.

[77] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Self-taught clustering," in *Proceedings of the 25th international conference on Machine learning*, pp. 200–207, ACM, 2008.

[78] Z. Wang, Y. Song, and C. Zhang, "Transferred dimensionality reduction," in *Machine learning and knowledge discovery in databases*, pp. 550–565, Springer, 2008.

[79] N. Bel, C. H. Koster, and M. Villegas, "Cross-lingual text categorization," in *Research and Advanced Technology for Digital Libraries*, pp. 126–139, Springer, 2003.

[80] Y. Zhu, Y. Chen, Z. Lu, S. J. Pan, G.-R. Xue, Y. Yu, and Q. Yang, "Heterogeneous transfer learning for image classification.," in *AAAI*, 2011.

[81] Q. Yang, Y. Chen, G.-R. Xue, W. Dai, and Y. Yu, "Heterogeneous transfer learning for image clustering via the social web," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pp. 1–9, Association for Computational Linguistics, 2009.

[82] T. Evgeniou, C. A. Micchelli, and M. Pontil, "Learning multiple tasks with kernel methods," in *Journal of Machine Learning Research*, pp. 615–637, 2005.

[83] A. Argyriou, T. Evgeniou, and M. Pontil, "Convex multi-task feature learning," *Machine Learning*, vol. 73, no. 3, pp. 243–272, 2008.

[84] H. Daumé III, "Bayesian multitask learning with latent hierarchies," in *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pp. 135–142, AUAI Press, 2009.

[85] R. K. Ando and T. Zhang, "A framework for learning predictive structures from multiple tasks and unlabeled data," *The Journal of Machine Learning Research*, vol. 6, pp. 1817–1853, 2005.

[86] N. Loeff and A. Farhadi, "Scene discovery by matrix factorization," in *Computer Vision–ECCV 2008*, pp. 451–464, Springer, 2008.

[87] A. Quattoni, M. Collins, and T. Darrell, "Transfer learning for image classification with sparse prototype representations," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pp. 1–8, IEEE, 2008.

[88] A. Torralba, K. P. Murphy, and W. T. Freeman, "Sharing visual features for multiclass and multiview object detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 5, pp. 854–869, 2007.

[89] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Machine learning*, vol. 79, no. 1-2, pp. 151–175, 2010.

[90] A. Kumar, A. Saha, and H. Daume, "Co-regularization based semi-supervised domain adaptation," in *Advances in Neural Information Processing Systems*, pp. 478–486, 2010.

[91] S. J. Pan, X. Ni, J.-T. Sun, Q. Yang, and Z. Chen, "Cross-domain sentiment classification via spectral feature alignment," in *Proceedings of the 19th international conference on World wide web*, pp. 751–760, ACM, 2010.

[92] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. A. Yang, "Domain adaptation via transfer component analysis," *IEEE Transactions on Neural Networks*, vol. 22, no. 2, pp. 199–210, 2011.

[93] L. X. Duan, D. Xu, and I. W. H. Tsang, "Domain adaptation from multiple sources: A domain-dependent regularization approach," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 3, pp. 504–518, 2012.

[94] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[95] J. Zhu and E. Hovy, "Active learning for word sense disambiguation with methods for addressing the class imbalance problem," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, (Prague, Czech Republic), p. 783–790, 2007.

[96] S. Bickel and T. Scheffer, "Dirichlet-enhanced spam filtering based on biased samples," *Advances in neural information processing systems*, vol. 19, p. 161, 2007.

[97] M. Sugiyama, M. Krauledat, and K. R. Muller, "Covariate shift adaptation by importance weighted cross validation," *Journal of Machine Learning Research*, vol. 8, pp. 985–1005, 2007.

[98] M. Yamada, M. Sugiyama, and T. Matsui, "Semi-supervised speaker identification under covariate shift," *Signal Processing*, vol. 90, no. 8, pp. 2353–2361, 2010.

[99] A. Gretton, A. Smola, J. Huang, M. Schmittfull, K. Borgwardt, and B. Schlkopf, "Covariate shift by kernel mean matching," in *Dataset Shift in Machine Learning* (A. S. In J. Qui nonero Candela, M. Sugiyama and e. N. D. Lawrence, eds.), MIT Press, Cambridge,MA, 2008.

[100] M. Sugiyama and K.-R. Müller, "Input-dependent estimation of generalization error under covariate shift," *Statistics & Decisions*, vol. 23, no. 4/2005, pp. 249–279, 2005.

[101] J. Huang, A. Smola, A. Gretton, K. Borgwardt, and B. Scholkopf, "Correcting sample selection bias by unlabeled data," *In Advances in Neural Information Processing Systems*, 2007.

[102] R. Klinkenberg and T. Joachims, "Detecting concept drift with support vector machines.," in *ICML*, pp. 487–494, 2000.

[103] P. Cunningham, N. Nowlan, S. J. Delany, and M. Haahr, "A case-based approach to spam filtering that can track concept drift," in *The ICCBR*, vol. 3, pp. 03–2003, 2003.

[104] W. Tu and S. Sun, "Dynamical ensemble learning with model-friendly classifiers for domain adaptation," in *Pattern Recognition (ICPR), 2012 21st International Conference on*, pp. 1181–1184, IEEE, 2012.

[105] W. N. Street and Y. Kim, "A streaming ensemble algorithm (sea) for large-scale classification," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 377–382, ACM, 2001.

[106] K. susmakova, "Human sleep and sleep EEG," *Measurement Science Review*, no. 4(2), p. 59–74, 2004.

[107] N. C. on Sleep Disorders Research (National Heart, B. Institute), and T.-N. S. R. C. Committee, *2003 national sleep disorders research pla*. No. 3, US Dept. of Health and Human Services, National Institutes of Health, National Heart, Lung, and Blood Institute, National Center on Sleep Disorders Research, Trans-NIH Sleep Research Coordinating Committee, 2003.

[108] D. Millet, "The origins of EEG," in *7th Annual Meeting of the International Society for the History of the Neurosciences (ISHN)*, 2002.

[109] E. Newton Harvey, A. L. Loomis, and G. A. Hobart, "Cerebral states during sleep as studied by human brain potentials," *The Scientific Monthly*, vol. 45, pp. 191–192, 1937.

[110] E. Aserinsky and N. Kleitman, "Regularly occurring periods of eye motility, and concomitant phenomena, during sleep. 1953.," *The Journal of neuropsychiatry and clinical neurosciences*, vol. 15, no. 4, pp. 454–455, 2002.

[111] R. D. Ogilvie, "The process of falling asleep," *Sleep medicine reviews*, vol. 5, no. 3, pp. 247–270, 2001.

[112] H. Danker-Hopfe, D. Kunz, G. Gruber, G. Klösch, J. Lorenzo, S. Himanen, B. Kemp, T. Penzel, J. Röschke, H. Dorn, *et al.*, "Interrater reliability between scorers from eight european sleep laboratories in subjects with different sleep disorders," *Journal of sleep research*, vol. 13, no. 1, pp. 63–69, 2004.

[113] A. Rechtschaffen and A. Kales, "A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects," 1968.

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

[115] G. K. Zammit, J. Weiner, N. Damato, G. P. Sillup, and C. A. McMillan, "Quality of life in people with insomnia.," *Sleep: Journal of Sleep Research & Sleep Medicine*, 1999.

[116] A. S. D. Association, D. Committee, *et al.*, "International classification of sleep disorders: diagnostic and coding manual," *American Academy of Sleep Medicine*, 2005.

[117] M. H. Silber, S. Ancoli-Israel, M. H. Bonnet, S. Chokroverty, M. M. Grigg-Damberger, M. Hirshkowitz, S. Kapen, S. A. Keenan, M. H. Kryger, T. Penzel, M. R. Pressman,

and C. Iber, "The visual scoring of sleep in adults," *J Clin Sleep Med*, vol. 3, no. 2, pp. 121–31, 2007.

[118] S. L. Himanen, A. Joutsen, and J. Virkkala, "Visual assessment of selected high amplitude frontopolar slow waves of sleep: Differences between healthy subjects and apnea patients," *Clinical EEG and Neuroscience*, vol. 35, no. 3, pp. 125–131, 2004.

[119] A. G. Xiromeritis, A. A. Hatziefthimiou, G. M. Hadjigeorgiou, K. I. Gourgoulianis, D. N. Anagnostopoulou, and N. V. Angelopoulos, "Quantitative spectral analysis of vigilance EEG in patients with obstructive sleep apnoea syndrome," *Sleep and Breathing*, vol. 15, no. 1, pp. 121–128, 2011.

[120] T. Al-Ani, Y. Hamam, D. Novak, P. P. Mendoza, L. Lhotska, F. Lofaso, D. Isabey, and R. Ffodil, "Noninvasive automatic sleep apnea classification system, modeling and simulation in biology," in *Medicine and Biomedical Engineering Conference, Biomedsim 2005*, 2005.

[121] N. Ohisa, H. Ogawa, N. Murayama, and K. Yoshida, "A novel EEG index for evaluating the sleep quality in patients with obstructive sleep apnea-hypopnea syndrome," *Tohoku Journal of Experimental Medicine*, vol. 223, no. 4, pp. 285–289, 2011.

[122] D. Z. Carvalho, G. J. L. Gerhardt, G. Dellagustin, E. L. de Santa-Helena, N. Lemke, A. Z. Segal, and S. V. Schonwald, "Loss of sleep spindle frequency deceleration in obstructive sleep apnea," *Clinical Neurophysiology*, vol. 125, no. 2, pp. 306–312, 2014.

[123] C. Stepnowsky, D. Levendowski, D. Popovic, I. Ayappa, and D. M. Rapoport, "Scoring accuracy of automated sleep staging from a bipolar electroocular recording compared to manual scoring by multiple raters," *Sleep Medicine*, vol. 14, no. 11, pp. 1199–1207, 2013.

[124] C. J. Lauer, D. Riemann, M. Wiegand, and M. Berger, "From early to late adulthood - changes in EEG sleep of depressed-patients and healthy-volunteers," *Biological Psychiatry*, vol. 29, no. 10, pp. 979–993, 1991.

[125] R. Armitage, "Sleep and circadian rhythms in mood disorders," *Acta Psychiatrica Scandinavica*, vol. 115, pp. 104–115, 2007.

[126] M. Dresler, V. I. Spoormaker, P. Beitinger, M. Czisch, M. Kimura, A. Steiger, and F. Holsboer, "Neuroscience-driven discovery and development of sleep therapeutics," *Pharmacology & Therapeutics*, vol. 141, no. 3, pp. 300–334, 2014.

[127] C. F. Reynolds, D. H. Shaw, T. F. Newton, P. A. Coble, and D. J. Kupfer, "EEG sleep in outpatients with generalized anxiety - a preliminary comparison with depressed outpatients," *Psychiatry Research*, vol. 8, no. 2, pp. 81–89, 1983.

[128] G. N. Papadimitriou, M. Kerkhofs, C. Kempenaers, and J. Mendlewicz, "EEG sleep studies in patients with generalized anxiety disorder," *Psychiatry Research*, vol. 26, no. 2, pp. 183–190, 1988.

[129] K. Spiegelhalder, W. Regen, B. Feige, J. Holz, H. Piosczyk, C. Baglioni, D. Riemann, and C. Nissen, "Increased EEG sigma and beta power during NREM sleep in primary insomnia," *Biological Psychology*, vol. 91, no. 3, pp. 329–333, 2012.

[130] C. H. Bastien, M. LeBlanc, J. Carrier, and C. M. Morin, "Sleep EEG power spectra, insomnia, and chronic use of benzodiazepines," *Sleep*, vol. 26, no. 3, pp. 313–317, 2003.

[131] G. A. Kerkhof and H. P. A. Van Dongen, "Measurement of sleep," *Human Sleep and Cognition, Part I: Basic Research*, vol. 185, pp. 21–35, 2010.

[132] M. E. Thase, "Depression, sleep, and antidepressants," *J Clin Psychiatry*, vol. 59 Suppl 4, pp. 55–65, 1998.

[133] H. P. Landolt, E. B. Raimo, B. J. Schnierow, J. R. Kelsoe, M. H. Rapaport, and J. C. Gillin, "Sleep and sleep electroencephalogram in depressed patients treated with phenelzine," *Archives of General Psychiatry*, vol. 58, no. 3, pp. 268–276, 2001.

[134] A. Steiger, O. Benkert, and F. Holsboer, "Effects of long-term treatment with the mao-a inhibitor moclobemide on sleep EEG and nocturnal hormonal secretion in normal men," *Neuropsychobiology*, vol. 30, no. 2-3, pp. 101–105, 1994.

[135] U. Vonbardeleben, A. Steiger, A. Gerken, and F. Holsboer, "Effects of fluoxetine upon pharmacoendocrine and sleep-EEG parameters in normal controls," *International Clinical Psychopharmacology*, vol. 4, pp. 1–5, 1989.

[136] H. E. Kuenzel, H. Murck, K. Held, M. Ziegenbein, and A. Steiger, "Reboxetine induces similar sleep-EEG changes like SSRI's in patients with depression," *Pharmacopsychiatry*, vol. 37, no. 5, pp. 193–195, 2004.

[137] M. Kluge, P. Schussler, and A. Steiger, "Duloxetine increases stage 3 sleep and suppresses rapid eye movement (REM) sleep in patients with major depression," *European Neuropsychopharmacology*, vol. 17, no. 8, pp. 527–531, 2007.

[138] D. A. Schmid, A. Wichniak, M. Uhr, M. Ising, H. Brunner, K. Held, J. C. Weikel, A. Sonntag, and A. Steiger, "Changes of sleep architecture, spectral composition of sleep EEG, the nocturnal secretion of cortisol, ACTH, GH, prolactin, melatonin, ghrelin, and leptin, and the DEX-CRH test in depressed patients during treatment with mirtazapine," *Neuropsychopharmacology*, vol. 31, no. 4, pp. 832–844, 2006.

[139] R. Luthringer, M. Toussaint, N. Schaltenbrand, P. Bailey, P. G. Danjou, D. Hackett, J. Y. Guichoux, and J. P. Macher, "A double-blind, placebo-controlled evaluation of the effects of orally administered venlafaxine on sleep in inpatients with major depression," *Psychopharmacology Bulletin*, vol. 32, no. 4, pp. 637–646, 1996.

[140] G. M. Saletu-Zyhlarz, M. H. Abu-Bakr, P. Anderer, G. Gruber, M. Mandl, R. Strobl, D. Gollner, W. Prause, and B. Saletu, "Insomnia in depression: Differences in objective and subjective sleep and awakening quality to normal controls and acute effects of trazodone," *Progress in Neuro-Psychopharmacology & Biological Psychiatry*, vol. 26, no. 2, pp. 249–260, 2002.

[141] M. Ferrara and L. De Gennaro, "Going local: Insights from EEG and stereo-EEG studies of the human sleep-wake cycle," *Current Topics in Medicinal Chemistry*, vol. 11, no. 19, pp. 2423–2437, 2011.

[142] P. Anderer, G. Gruber, S. Parapatics, M. Woertz, T. Miazhynskaia, G. Klosch, B. Saletu, J. Zeitlhofer, M. J. Barbanoj, H. Danker-Hopfe, S. L. Himanen, B. Kemp, T. Penzel, M. Grozinger, D. Kunz, P. Rappelsberger, A. Schlogl, and G. Dorffner, "An E-health solution for automatic sleep classification according to rechtschaffen and kales: Validation study of the somnolyzer 24 x 7 utilizing the siesta database," *Neuropsychobiology*, vol. 51, no. 3, pp. 115–133, 2005.

[143] O. R. Pacheco and F. Vaz, "Integrated system for analysis and automatic classification of sleep EEG," in *Engineering in Medicine and Biology Society, 1998. Proceedings of the 20th Annual International Conference of the IEEE*, vol. 4, pp. 2062–2065, IEEE, 1998.

[144] J. Fell, J. Röschke, K. Mann, and C. Schäffner, "Discrimination of sleep stages: a comparison between spectral and nonlinear EEG measures," *Electroencephalography and clinical Neurophysiology*, vol. 98, no. 5, pp. 401–410, 1996.

[145] A. Akin and T. Akgul, "Detection of sleep spindles by discrete wavelet transform," in *Bioengineering Conference, 1998. Proceedings of the IEEE 24th Annual Northeast*, pp. 15–17, IEEE, 1998.

[146] T. Akgül, M. Sun, R. Sclahassi, *et al.*, "Characterization of sleep spindles using higher order statistics and spectra," *Biomedical Engineering, IEEE Transactions on*, vol. 47, no. 8, pp. 997–1009, 2000.

[147] D. Gorur, U. Halici, H. Aydin, G. Ongun, F. Ozgen, and K. Leblebicioglu, "Sleep spindles detection using short time fourier transform and neural networks," in *Neural Networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference on*, vol. 2, pp. 1631–1636, IEEE, 2002.

[148] A. Erdamar, F. Duman, and S. Yetkin, "A wavelet and teager energy operator based method for automatic detection of k-complex in sleep EEG," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1284–1290, 2012.

[149] V. Pohl and E. Fahr, "Neuro-fuzzy recognition of K-complexes in sleep EEG signals," in *Engineering in Medicine and Biology Society, 1995., IEEE 17th Annual Conference*, vol. 1, pp. 789–790, IEEE, 1995.

[150] S. Roberts and L. Tarassenko, "New method of automated sleep quantification," *Medical and Biological Engineering and Computing*, vol. 30, no. 5, pp. 509–517, 1992.

[151] D. Alvarez-Estevez, J. M. Fernandez-Pastoriza, E. Hernandez-Pereira, and V. Moret-Bonillo, "A method for the automatic analysis of the sleep macrostructure in continuum," *Expert Systems with Applications*, vol. 40, p. 1796–1803, 2013.

[152] H. G. Jo, J. Y. Park, C. K. Lee, S. K. An, and S. K. Yoo, "Genetic fuzzy classifier for sleep stage identification," *Computers in Biology and Medicine*, vol. 40, no. 7, pp. 629–634, 2010.

[153] S. Gunes, K. Polat, and S. Yosunkaya, "Efficient sleep stage recognition system based on EEG signal using k-means clustering based feature weighting," *Expert Systems with Applications*, vol. 37, no. 12, pp. 7922–7928, 2010.

[154] M. E. Tagluk, N. Sezgin, and M. Akin, "Estimation of sleep stages by an artificial neural network employing EEG, EMG and EOG," *Journal of Medical Systems*, vol. 34, no. 4, pp. 717–725, 2010.

[155] F. Chapotot and G. Becq, "Automated sleep-wake staging combining robust feature extraction, artificial neural network classification, and flexible decision rules," *International Journal of Adaptive Control and Signal Processing*, vol. 24, no. 5, pp. 409–423, 2010.

[156] E. Niedermeyer and F. L. da Silva, *Electroencephalography: basic principles, clinical applications, and related fields*. Lippincott Williams & Wilkins, 2005.

[157] M. Teplan, "Fundamentals of EEG measurement," *Measurement science review*, vol. 2, no. 2, pp. 1–11, 2002.

[158] I. Gutberlet, S. Debener, T. Jung, S. Makeig, S. Tong, and N. Thakor, "Techniques of EEG recording and preprocessing," *Quantative EEG Analysis Methods and Clinical Applications*, pp. 23–49, 2009.

[159] J. Knight, *Signal fraction analysis and artifact removal in EEG*. Master thesis, Colorado State University, Colorado, 2003.

[160] B. Şen, M. Peker, A. Çavuşoğlu, and F. V. Çelebi, "A comparative study on classification of sleep stage based on EEG signals using feature selection and classification algorithms," *Journal of medical systems*, vol. 38, no. 3, pp. 1–21, 2014.

[161] M. Ronzhina, O. Janousek, J. Kolarova, M. Novakova, P. Honzik, and I. Provaznik, "Sleep scoring using artificial neural networks," *Sleep Medicine Reviews*, vol. 16, no. 3, pp. 251–263, 2012.

[162] P. Van Hese, W. Philips, J. De Koninck, R. Van de Walle, and I. Lemahieu, "Automatic detection of sleep stages using the EEG," in *Engineering in Medicine and Biology Society, 2001. Proceedings of the 23rd Annual International Conference of the IEEE*, vol. 2, pp. 1944–1947, IEEE, 2001.

[163] H. Daume III and D. Marcu, "Domain adaptation for statistical classifiers," *Journal of Artificial Intelligence Research*, pp. 101–126, 2006.

[164] J. Blitzer, R. McDonald, and F. Pereira, "Domain adaptation with structural correspondence learning," in *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, (Sydney, Australia), p. 120–128, 2006.

[165] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," *Advances in Neural Information Processing Systems*, vol. 19, 2007.

[166] H. Daumé III, "Frustratingly easy domain adaptation," *arXiv preprint arXiv:0907.1815*, 2009.

[167] J. Jiang and C. Zhai, "Instance weighting for domain adaptation in nlp," in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Series Instance weighting for domain adaptation in NLP, (Prague, Czech Republic), p. 254–271, 2007.

[168] J. Hoffman, T. Darrell, and K. Saenko, "Continuous manifold based adaptation for evolving visual domains," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pp. 867–874, IEEE, 2014.

[169] S. Bickel, *Learning under Differing Training and Test Distributions*. PhD thesis, 2009.

[170] B. M. Hill, "The estimation of probabilities - an essay on modern bayesian methods - good,ij," *Journal of the American Statistical Association*, vol. 60, no. 312, pp. 1217–1218, 1965.

[171] E. Agirre and O. L. de Lacalle, "Supervised domain adaption for wsd," in *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 42–50, Association for Computational Linguistics, 2009.

[172] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 2066–2073, IEEE, 2012.

[173] R. Gopalan, R. Li, and R. Chellappa, "Domain adaptation for object recognition: An unsupervised approach," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 999–1006, IEEE, 2011.

[174] A. Bergamo and L. Torresani, "Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach," in *Advances in Neural Information Processing Systems*, pp. 181–189, 2010.

[175] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Computer Vision–ECCV 2010*, pp. 213–226, Springer, 2010.

[176] Q. Li, "Literature survey: Domain adaptation algorithms for natural language processing," 2012.

[177] B. Geng, D. C. Tao, and C. Xu, "Daml: Domain adaptation metric learning," *IEEE Transactions on Image Processing*, vol. 20, no. 10, pp. 2980–2989, 2011.

[178] O. Beijbom, "Domain adaptations for computer vision applications," *arXiv preprint arXiv:1211.4860*, 2012.

[179] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain adaptation with multiple sources," in *Advances in neural information processing systems*, pp. 1041–1048, 2009.

[180] Q. Sun, R. Chattopadhyay, S. Panchanathan, and J. Ye, "A two-stage weighting framework for multi-source domain adaptation," in *Advances in Neural Information Processing Systems 24* (J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, eds.), pp. 505–513, Curran Associates, Inc., 2011.

[181] L. Duan, D. Xu, and S.-F. Chang, "Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1338–1345, IEEE, 2012.

[182] R. Chattopadhyay, N. C. Krishnan, and S. Panchanathan, "Hierarchical domain adaptation for semg signal classification across multiple subjects," *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 7853–7856, 2011.

[183] S.-L. Sun and H.-L. Shi, "Bayesian multi-source domain adaptation," in *Machine Learning and Cybernetics (ICMLC), 2013 International Conference on*, vol. 1, pp. 24–28, IEEE, 2013.

[184] Z. Xu and S. Sun, "Multi-source transfer learning with multi-view adaboost," in *Neural Information Processing*, pp. 332–339, Springer, 2012.

[185] W. Li, L. Duan, D. Xu, and I. W. Tsang, "Learning with augmented features for supervised and semi-supervised heterogeneous domain adaptation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 6, pp. 1134–1148, 2014.

[186] L. Duan, D. Xu, and I. Tsang, "Learning with augmented features for heterogeneous domain adaptation," *arXiv preprint arXiv:1206.4660*, 2012.

[187] M. Sugiyama, S. Nakajima, H. Kashima, P. v. Bunau, and M. Kawanabe, "Direct importance estimation with model selection and its application to covariate shift adaptation," *Advances in Neural Information Processing Systems*, 2008.

[188] C. Widmer, *Domain Adaptation in Sequence Analysis*. PhD thesis, Universit at Tubingen, 2008.

[189] X. Li and J. Bilmes, "A bayesian divergence prior for classifier adaptation," in *Proceedings of the 11th International Conference on Artificial Intelligence and Statistics*, Series A Bayesian divergence prior for classifier adaptation, (San Juan, Puerto, Rico), 2007.

[190] B. Gong, K. Grauman, and F. Sha, "Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation," in *Proceedings of The 30th International Conference on Machine Learning*, pp. 222–230, 2013.

[191] S. Satpal and S. Sarawagi, "Domain adaptation of conditional probability models via feature subsetting," in *Knowledge Discovery in Databases: PKDD 2007*, pp. 224–235, Springer, 2007.

[192] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification," in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, Series Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification, (Prague, Czech Republic), Association for Computational Linguistics, 2007.

[193] E. C. David McClosky and M. Johnson, "Effective self-training for parsing," in *HLT-NAACL 2006*, Series Effective self-training for parsing, pp. 152–159, 2006.

[194] M. Chen, K. Q. Weinberger, and J. Blitzer, "Co-training for domain adaptation," in *Advances in neural information processing systems*, pp. 2456–2464, 2011.

[195] J. Zheng, M.-Y. Liu, R. Chellappa, and P. J. Phillips, "A grassmann manifold-based domain adaptation approach," in *21st International Conference on Pattern Recognition (ICPR),*, pp. 2095–2099, IEEE, 2012.

[196] Z. Cui, W. Li, D. Xu, S. Shan, X. Chen, and X. Li, "Flowing on riemannian manifold: Domain adaptation by shifting covariance," *Cybernetics, IEEE Transactions on*, vol. 44, no. 12, pp. 2264–2273, 2014.

[197] D. Tuia, J. Munoz-Mari, L. Gomez-Chova, and J. Malo, "Graph matching for adaptation in remote sensing," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 51, no. 1, pp. 329–341, 2013.

[198] Z. H. Zhang and J. Zhou, "Multi-task clustering via domain adaptation," *Pattern Recognition*, vol. 45, no. 1, pp. 465–473, 2012.

[199] S. Shekhar, V. M. Patel, H. V. Nguyen, and R. Chellappa, "Generalized domain-adaptive dictionaries," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pp. 361–368, IEEE, 2013.

[200] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: An overview of recent advances," *IEEE Signal Processing Magazine*, 2014.

[201] C. Cortes, Y. Mansour, and M. Mohri, "Learning bounds for importance weighting," in *Advances in neural information processing systems*, pp. 442–450, 2010.

[202] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for transfer learning," in *Proceedings of the 24th international conference on Machine learning*, pp. 193–200, ACM, 2007.

[203] E. Eaton and M. DesJardins, "Set-based boosting for instance-level transfer," in *Data Mining Workshops, 2009. ICDMW'09. IEEE International Conference on*, pp. 422–428, IEEE, 2009.

[204] C. Chelba and A. Acero, "Adaptation of maximum entropy capitalizer: Little data can help a lot," *Computer Speech & Language*, vol. 20, no. 4, pp. 382–399, 2006.

[205] K. Shah, L. Barrault, and H. Schwenk, "Translation model adaptation by resampling," in *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pp. 392–399, Association for Computational Linguistics, 2010.

[206] B. Gong, F. Sha, and K. Grauman, "Overcoming dataset bias: An unsupervised domain adaptation approach," in *NIPS Workshop on Large Scale Visual Recognition and Retrieval*, 2012.

[207] S. J. Pan, J. T. Kwok, and Q. Yang, "Transfer learning via dimensionality reduction," in *Proc. of the Twenty-Third AAAI Conference on Artificial Intelligence*, Series Transfer learning via dimensionality reduction, 2008.

[208] H. Guo, H. Zhu, Z. Guo, X. Zhang, X. Wu, and Z. Su, "Domain adaptation with latent semantic association for named entity recognition," in *Proc. HTL-NAACL*, Series Domain adaptation with latent semantic association for named entity recognition, p. 281–289, 2009.

[209] J. Jiang and C. Zhai, "A two-stage approach to domain adaptation for statistical classifiers," in *Proceedings of the ACM 16th Conference on Information and Knowledge Management*, p. 401–410, 2007.

[210] B. Chen, W. Lam, I. Tsang, and T. T.-L. Wong, "Extracting discriminative concepts for domain adaptation in text mining," in *International Conference on Knowledge Discovery and Data Mining*, Series Extracting discriminative concepts for domain adaptation in text mining, 2009.

[211] F. Huang and A. Yates, "Distributional representations for handling sparsity in supervised sequence-labeling," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, p. 495–503, 2009.

[212] F. Huang and A. Yates, "Exploring representation-learning approaches to domain adaptation," in *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing.*, 2010.

[213] M. Ciaramita and O. Chapelle, "Adaptive parameters for entity recognition with perceptron hmms," in *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, pp. 1–7, Association for Computational Linguistics, 2010.

[214] A. Margolis, "A literature review of domain adaptation with unlabeled data," 2011.

[215] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," *Computer Speech & Language*, vol. 9, no. 2, pp. 171–185, 1995.

[216] R.Reichart and A.Rappoport, "Self-training for enhancement and domain adaptation of statistical parsers trained on small datasets," in *Proceedings of the 45th Meeting of the Association for Computational Linguistics*, p. 616–623, 2007.

[217] D. McClosky and E.Charniak, "Self-training for biomedical parsing," in *Proceedings of the 46th Meeting of the Association for Computational Linguistics*, Series Self-Training

for Biomedical Parsing, (Columbus, Ohio: Association for Computational Linguistics), p. 101–104, 2008.

[218] D. McClosky, E. Charniak, and M. Johnson, "Reranking and self-training for parser adaptation," *Coling/Acl 2006, Vols 1 and 2, Proceedings of the Conference*, pp. 337–344, 2006.

[219] J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman, "Learning bounds for domain adaptation," in *Advances in Neural Information Processing Systems*, Cambridge, Massachusetts,USA: MIT Press, 2008.

[220] X. Wan, "Co-training for cross-lingual sentiment classification," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, Series Co-Training for Cross-Lingual sentiment classification, (Suntec, Singapore: Association for Computational Linguistics), p. 235–243, 2009.

[221] W. Wang, "Combining discriminative re-ranking and co-training for parsing mandarin speech transcripts," in *Proc. ICASSP*, Series Combining discriminative re-ranking and co-training for parsing mandarin speech transcripts, 2009.

[222] B. Kulis, K. Saenko, and T. Darrell, "What you saw is not what you get: Domain adaptation using asymmetric kernel transforms," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1785–1792, IEEE, 2011.

[223] M. Baktashmotlagh, M. T. Harandi, B. C. Lovell, and M. Salzmann, "Unsupervised domain adaptation by domain invariant projection," in *Computer Vision (ICCV), 2013 IEEE International Conference on*, pp. 769–776, IEEE, 2013.

[224] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars, "Unsupervised visual domain adaptation using subspace alignment," in *Computer Vision (ICCV), 2013 IEEE International Conference on*, pp. 2960–2967, IEEE, 2013.

[225] J. H. Ham, D. D. Lee, and L. K. Saul, "Learning high dimensional correspondences from low dimensional manifolds," 2003.

[226] C. Wang and S. Mahadevan, "Manifold alignment using procrustes analysis," in *Proceedings of the 25th international conference on Machine learning*, pp. 1120–1127, ACM, 2008.

[227] C. Wang and S. Mahadevan, "A general framework for manifold alignment.," in *AAAI Fall Symposium: Manifold Learning and Its Applications*, 2009.

[228] C. Wang and S. Mahadevan, "Heterogeneous domain adaptation using manifold alignment," in *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, vol. 22, p. 1541, 2011.

[229] N. Ponomareva and M. Thelwall, "Do neighbours help?: an exploration of graph-based algorithms for cross-domain sentiment classification," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 655–665, 2012.

[230] Q. Wu, S. Tan, and X. Cheng, "Graph ranking for sentiment transfer," in *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pp. 317–320, Association for Computational Linguistics, 2009.

[231] A. B. Goldberg and X. Zhu, "Seeing stars when there aren't many stars: graph-based semi-supervised learning for sentiment categorization," in *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing*, pp. 45–52, 2006.

[232] J. Yang, R. Yan, and A. G. Hauptmann, "Cross-domain video concept detection using adaptive svms," in *Proceedings of the 15th international conference on Multimedia*, pp. 188–197, ACM, 2007.

[233] Y. Aytar and A. Zisserman, "Tabula rasa: Model transfer for object category detection," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 2252–2259, IEEE, 2011.

[234] W. Jiang, E. Zavesky, S.-F. Chang, and A. Loui, "Cross-domain learning methods for high-level visual concept classification," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pp. 161–164, IEEE, 2008.

[235] B. A. Olshausen *et al.*, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.

[236] Q. Qiu, V. M. Patel, P. Turaga, and R. Chellappa, "Domain adaptive dictionary learning," in *Computer Vision–ECCV 2012*, pp. 631–645, Springer, 2012.

[237] J. Ni, Q. Qiu, and R. Chellappa, "Subspace interpolation via dictionary learning for unsupervised domain adaptation," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pp. 692–699, IEEE, 2013.

[238] S. Wang, D. Zhang, Y. Liang, and Q. Pan, "Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 2216–2223, IEEE, 2012.

[239] T. Kanamori, S. Hido, and M. Sugiyama, "Efficient direct density ratio estimation for non-stationarity adaptation and outlier detection," in *Advances in neural information processing systems*, pp. 809–816, 2009.

[240] Y. Tsuboi, H. Kashima, S. Hido, S. Bickel, and M. Sugiyama, "Direct density ratio estimation for large-scale covariate shift adaptation.," *Information and Media Technologies*, vol. 4, no. 2, pp. 529–546, 2009.

[241] M. Yamada, T. Suzuki, T. Kanamori, H. Hachiya, and M. Sugiyama, "Relative density-ratio estimation for robust distribution comparison," *Neural computation*, vol. 25, no. 5, pp. 1324–1370, 2013.

[242] H. Adeli, Z. Zhou, and N. Dadmehr, "Analysis of EEG records in an epileptic patient using wavelet transform," *Journal of Neuroscience Methods*, vol. 123, no. 1, pp. 69–87, 2003.

[243] D. Walden and A. Percival, "Wavelet methods for time series analysis," 2000.

[244] S. Ahmad, *Temporal Pattern Identification and Summarization Method for Complex Time Serial Data*. PhD thesis, Uni.of Surrey Guildford, 2007.

[245] I. Omerhodzic, S. Avdakovic, A. Nuhanovic, and K. Dizdarevic, "Energy distribution of EEG signals: EEG signal wavelet-neural network classifier," *International Journal of Biological and Life Sciences*, vol. 6, 2010.

[246] A. S. Yilmaz, A. Alkan, and M. H. Asyali, "Applications of parametric spectral estimation methods on detection of power system harmonics," *Electric Power Systems Research*, vol. 78, no. 4, pp. 683–693, 2008.

[247] E. Arbabi, M. B. Shamsollahi, and R. Sameni, "Comparison between effective features used for the bayesian and the svm classifiers in bci," *2005 27th Annual International*

Conference of the IEEE Engineering in Medicine and Biology Society, Vols 1-7, pp. 5365–5368, 2005.

[248] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 4, pp. 623–656, 1948.

[249] A. Renyi, "On measures of entropy and information," in *Fourth Berkeley Symp. Math. Stat. Prob.*, vol. 1, (Berkely), p. 547, University of california press, 1960.

[250] T. Maszczyk and W. Duch, "Comparison of shannon, renyi and tsallis entropy used in decision trees," *Artificial Intelligence and Soft Computing - ICAISC 2008, Proceedings*, vol. 5097, pp. 643–651, 2008.

[251] C. Tsallis, R. S. Mendes, and A. R. Plastino, "The role of constraints within generalized nonextensive statistics," *Physica A*, vol. 261, no. 3-4, pp. 534–554, 1998.

[252] F. Mormann, R. G. Andrzejak, C. E. Elger, and K. Lehnertz, "Seizure prediction: the long and winding road," *Brain*, vol. 130, pp. 314–333, 2007.

[253] R. Agarwal and J. Gotman, "Computer-assisted sleep staging," *IEEE Transactions on Biomedical Engineering*, vol. 48, no. 12, pp. 1412–1423, 2001.

[254] K. Ansari-Asl, G. Chanel, and T. Pun, "A channel selection method for EEG classification in emotion assessment based on synchronization likelihood," in *15th European Signal Processing Conference (EUSIPCO 2007)*, (Poznan, Poland), 2007.

[255] S. Devuyst, T. Dutoit, T. Ravet, P. Stenuit, M. Kerkhofs, and E. Stanus, "Automatic processing of EEG-EOG-EMG artifacts in sleep stage classification," *13th International Conference on Biomedical Engineering, Vols 1-3*, vol. 23, no. 1-3, pp. 146–150, 2009.

[256] H. C. Peng, F. H. Long, and C. Ding, "Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.

[257] A. W. Whitney, "A direct method of nonparametric measurement selection," *IEEE Trans. on Comput.*, vol. 20, no. 9, p. 1100 – 1103, 1971.

[258] P. Pudil, J. Novovičová, and J. Kittler, "Floating search methods in feature selection," *Pattern recognition letters*, vol. 15, no. 11, pp. 1119–1125, 1994.

[259] R. N. Khushaba, A. Al-Ani, and A. Al-Jumaily, "Feature subset selection using differential evolution and a statistical repair mechanism," *Expert Systems with Applications*, vol. 38, no. 9, pp. 11515–11526, 2011.

[260] J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, 1998.

[261] J. Zhu and T. Hastie, "Kernel logistic regression and the import vector machine," *Journal of Computational and Graphical Statistics*, vol. 14, no. 1, pp. 185–205, 2005.

[262] G. S. Kimeldorf and G. Wahba, "A correspondence between bayesian estimation on stochastic processes and smoothing by splines," *The Annals of Mathematical Statistics*, vol. 41, no. 2, pp. 495–502, 1970.

[263] F. Dinuzzo and B. Schölkopf, "The representer theorem for hilbert spaces: a necessary and sufficient condition," in *Advances in neural information processing systems*, pp. 189–196, 2012.

[264] J. Gui, T. Liu, D. Tao, Z. Sun, and T. Tan, "Representative vector machines: A unified framework for classical classifiers," *IEEE Transactions on Cybernetics*, vol. PP, no. 99, pp. 1–1, 2015.

[265] R. Roscher, W. Förstner, and B. Waske, "I2VM: incremental import vector machines," *Image and Vision Computing*, vol. 30, no. 4, pp. 263–278, 2012.

[266] B. D.Percival and T. A.Walden, *Wavelet Methods for Time Series Analysis*. Cambridge University Press, 2000.

[267] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," 2007.

[268] A. Krakovska and K. Mezeiova, "Automatic sleep scoring: A search for an optimal combination of measures," *Artificial Intelligence in Medicine*, vol. 53, no. 1, pp. 25–33, 2011.

[269] H. Koch, J. A. Christensen, R. Frandsen, M. Zoetmulder, L. Arvastson, S. R. Christensen, P. Jennum, and H. B. Sorensen, "Automatic sleep classification using a data-driven topic model reveals latent sleep states," *Journal of Neuroscience Methods*, vol. 235, pp. 130–137, 2014.

[270] A. M. Koupparis, V. Kokkinos, and G. K. Kostopoulos, "Semi-automatic sleep EEG scoring based on the hypnospectrogram," *Journal of Neuroscience Methods*, vol. 221, pp. 189–195, 2014.

[271] A. Brignol, T. Al-Ani, and X. Drouot, "Phase space and power spectral approaches for EEG-based automatic sleep–wake classification in humans: A comparative study using short and standard epoch lengths," *Computer Methods and Programs in Biomedicine*, vol. 109, no. 3, pp. 227–238, 2013.

[272] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C. K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet - components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. E215–E220, 2000.

[273] S. F. Quan, B. V. Howard, C. Iber, J. P. Kiley, F. J. Nieto, G. T. O'Connor, D. M. Rapoport, S. Redline, J. Robbins, J. Samet, *et al.*, "The sleep heart health study: design, rationale, and methods.," *Sleep*, no. 20, pp. 1077–85, 1998.

[274] N. Nicolaou and J. Georgiou, "Towards automatic sleep staging via cross-recurrence rate of EEG and ECG activity," *2013 IEEE Biomedical Circuits and Systems Conference (Biocas)*, pp. 198–201, 2013.

[275] B. Kemp, A. H. Zwinderman, B. Tuk, H. A. C. Kamphuisen, and J. J. L. Oberye, "Analysis of a sleep-dependent neuronal feedback loop: The slow-wave microcontinuity of the EEG," *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 9, pp. 1185–1194, 2000.

[276] F. Yaghouby and S. Sunderam, "Quasi-supervssssised scoring of human sleep in polysomnograms using augmented input variables," *Computers in Biology and Medicine*, 2015.

[277] M. G. Terzano, L. Parrino, A. Smerieri, R. Chervin, S. Chokroverty, C. Guilleminault, M. Hirschkowitz, M. Mahowald, H. Moldofsky, A. Rosa, R. Thomas, and A. Walters, "Atlas, rules, and recording techniques for the scoring of cyclic alternating pattern (CAP) in human sleep (vol 2, pg 537, 2001)," *Sleep Medicine*, vol. 3, no. 2, pp. 185–185, 2002.

[278] F. Ebrahimi, S.-K. Setarehdan, J. Ayala-Moyeda, and H. Nazeran, "Automatic sleep staging using empirical mode decomposition, discrete wavelet transform, time-domain,

and nonlinear dynamics features of heart rate variability signals," *Computer Methods and Programs in Biomedicine*, vol. 112, no. 1, pp. 47 – 57, 2013.

[279] A. Tsanas and G. Clifford, "Stage-independent, single lead EEG sleep spindle detection using the continuous wavelet transform and local weighted smoothing," *Frontiers in Human Neuroscience*, vol. 9, no. 181, 2015.

[280] H. Jasper, "Appendix to report to committee on clinical examination in EEG: the ten–twenty electrode system of the international federation," *Electroencephalography and Clinical Neurophysiology*, pp. 371–375, 1958.

[281] B. Kemp and J. Olivan, "European data format 'plus' (EDF+), an EDF alike standard format for the exchange of physiological data," *Clinical Neurophysiology*, vol. 114, no. 9, pp. 1755–1761, 2003.

[282] C. Chang and C. Lin, "LIBSVM: a library for support vector machines,," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, p. 27, 2011.

[283] S. L. Himanen and J. Hasan, "Limitations of Rechtschaffen and Kales," *Sleep Medicine Reviews*, vol. 4, no. 2, pp. 149–167, 2000.

[284] L. Flom and A. Safir, "Iris recognition system," Feb. 3 1987. US Patent 4,641,349.

[285] F. Jan, I. Usman, and S. Agha, "Iris localization in frontal eye images for less constrained iris recognition systems," *Digital Signal Processing*, vol. 22, no. 6, pp. 971–986, 2012.

[286] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 15, no. 11, pp. 1148–1161, 1993.

[287] R. P. Wildes, "Iris recognition: an emerging biometric technology," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1348–1363, 1997.

[288] A. Ahamed and M. I. H. Bhuiyan, "Low complexity iris recognition using curvelet transform," in *Informatics, Electronics & Vision (ICIEV), 2012 International Conference on*, pp. 548–553, IEEE, 2012.

[289] R. Farouk, "Iris recognition based on elastic graph matching and gabor wavelets," *Computer Vision and Image Understanding*, vol. 115, no. 8, pp. 1239–1244, 2011.

[290] A. Poursaberi and B. N. Araabi, "Iris recognition for partially occluded images: methodology and sensitivity analysis," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, pp. 1–12, 2006.

[291] K. Roy, P. Bhattacharya, and C. Y. Suen, "Towards nonideal iris recognition based on level set method, genetic algorithms and adaptive asymmetrical SVMs," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 3, pp. 458–475, 2011.

[292] K. Roy, P. Bhattacharya, and C. Y. Suen, "Iris segmentation using variational level set method," *Optics and Lasers in Engineering*, vol. 49, no. 4, pp. 578–588, 2011.

[293] R. Szewczyk, K. Grabowski, M. Napieralska, W. Sankowski, M. Zubert, and A. Napieralski, "A reliable iris recognition algorithm based on reverse biorthogonal wavelet transform," *Pattern Recognition Letters*, vol. 33, no. 8, pp. 1019–1026, 2012.

[294] F. Jan, I. Usman, and S. Agha, "Reliable iris localization using hough transform, histogram-bisection, and eccentricity," *Signal Processing*, vol. 93, no. 1, pp. 230–241, 2013.

[295] R. D. Labati and F. Scotti, "Noisy iris segmentation with boundary regularization and reflections removal," *Image and vision computing*, vol. 28, no. 2, pp. 270–277, 2010.

[296] D. S. Jeong, J. W. Hwang, B. J. Kang, K. R. Park, C. S. Won, D.-K. Park, and J. Kim, "A new iris segmentation method for non-ideal iris images," *Image and vision computing*, vol. 28, no. 2, pp. 254–260, 2010.

[297] P. Li, X. Liu, L. Xiao, and Q. Song, "Robust and accurate iris segmentation in very noisy iris images," *Image and Vision Computing*, vol. 28, no. 2, pp. 246–253, 2010.

[298] C. Belcher and Y. Du, "Region-based sift approach to iris recognition," *Optics and Lasers in Engineering*, vol. 47, no. 1, pp. 139–147, 2009.

[299] I. D. Set, "Comments on the CASIA version 1.0 iris data set," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 10, p. 1, 2007.

[300] H. Proença, S. Filipe, R. Santos, J. Oliveira, and L. A. Alexandre, "The UBIRIS. v2: A database of visible wavelength iris images captured on-the-move and at-a-distance," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 8, pp. 1529–1535, 2010.

[301] J. Kim, S. Cho, J. Choi, and R. J. Marks Ii, "Iris recognition using wavelet features," *Journal of VLSI signal processing systems for signal, image and video technology*, vol. 38, no. 2, pp. 147–156, 2004.

[302] C.-H. Chen and C.-T. Chu, "High performance iris recognition based on 1-D circular feature extraction and PSO–PNN classifier," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10351–10356, 2009.

[303] Z. He, T. Tan, Z. Sun, and X. Qiu, "Toward accurate and fast iris segmentation for iris biometrics," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 9, pp. 1670–1684, 2009.

[304] C.-C. Tsai, H.-Y. Lin, J. Taur, and C.-W. Tao, "Iris recognition using possibilistic fuzzy matching on local features," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 42, no. 1, pp. 150–162, 2012.

[305] M. N. Do and M. Vetterli, "The contourlet transform: an efficient directional multiresolution image representation," *Image Processing, IEEE Transactions on*, vol. 14, no. 12, pp. 2091–2106, 2005.

[306] M. Li, M. Jiang, M. Han, and M. Yang, "Iris recognition based on a novel multiresolution analysis framework," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pp. 4101–4104, IEEE, 2010.

[307] A. L. Da Cunha, J. Zhou, and M. N. Do, "The nonsubsampled contourlet transform: theory, design, and applications," *Image Processing, IEEE Transactions on*, vol. 15, no. 10, pp. 3089–3101, 2006.

[308] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. SMC-3, no. 6, pp. 610–621, 1973.

[309] S. Shah and A. Ross, "Iris segmentation using geodesic active contours," *Information Forensics and Security, IEEE Transactions on*, vol. 4, no. 4, pp. 824–836, 2009.

[310] L. Masek *et al.*, "Recognition of human iris patterns for biometric identification," *The University of Western Australia*, vol. 2, 2003.

[311] D. D. Po and M. N. Do, "Directional multiscale modeling of images using the contourlet transform," in *Statistical Signal Processing, 2003 IEEE Workshop on*, pp. 262–265, IEEE, 2003.

[312] M. N. Do and M. Vetterli, "Pyramidal directional filter banks and curvelets," in *Image Processing, 2001. Proceedings. 2001 International Conference on*, vol. 3, pp. 158–161, IEEE, 2001.

[313] L.-K. Soh and C. Tsatsoulis, "Texture analysis of sar sea ice imagery using gray level co-occurrence matrices," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 37, no. 2, pp. 780–795, 1999.

[314] D. A. Clausi, "An analysis of co-occurrence texture statistics as a function of grey level quantization," *Canadian Journal of remote sensing*, vol. 28, no. 1, pp. 45–62, 2002.

[315] R. M. Haralock and L. G. Shapiro, *Computer and robot vision*. Addison-Wesley Longman Publishing Co., Inc., 1991.

[316] S. Aksoy and R. M. Haralick, "Feature normalization and likelihood-based similarity measures for image retrieval," *Pattern Recognition Letters*, vol. 22, no. 5, pp. 563–582, 2001.

[317] S. Canu, Y. Grandvalet, V. Guigue, and A. Rakotomamonjy, "Svm and kernel methods matlab toolbox," *Perception Systemes et Information, INSA de Rouen, Rouen, France*, vol. 2, no. 2, 2005.

[318] J. Daugman, "New methods in iris recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 37, no. 5, pp. 1167–1175, 2007.

[319] A. Basit and M. Javed, "Localization of iris in gray scale images using intensity gradient," *Optics and Lasers in Engineering*, vol. 45, no. 12, pp. 1107–1114, 2007.

[320] M. T. Ibrahim, T. M. Khan, S. A. Khan, M. A. Khan, and L. Guan, "Iris localization using local histogram and other image statistics," *Optics and Lasers in Engineering*, vol. 50, no. 5, pp. 645–654, 2012.