Transfer Learning for Electroencephalogram Signals

Farah Abid[1], Ali Hassan[1*], Anum Abid[2], Imran Khan Niazi[3], Mads Jochumsen[4]

[1] College of Electrical and Mechanical Engineering National University of Sciences and Technology, Pakistan.
[2] University of Engineering and Technology Taxilla, Pakistan.
[3] Centre for Chiropractic Research, New Zealand College of Chiropractic, Auckland 1060, New Zealand.
[4] Center for Sensory-Motor Interaction, Department of Health Science and Technology, Aalborg University, Denmark.

* Corresponding author. Email: alihassan@ceme.nust.edu.pk

Manuscript submitted September 2, 2016; accepted April 1, 2017.
doi: 10.17706/ijbbb.2017.7.3.143-152

Abstract: The accessibility to Electroencephalogram (EEG) recording systems has enabled the healthcare providers to record the brain activity of patients under treatment, during multiple sessions. Thus brain changes can be observed and evaluated. It has been shown in many studies that the EEG data are never exactly the same when recordings are done in different sessions inducing a shift between the data of multiple sessions. This shift is induced due to the changes in parameters such as: the physical /mental state of the patient, the ambient environment, location of the electrodes, and impedance of the electrodes. The shift can be modelled as a covariate shift between multiple sessions. However, the algorithms that have been developed to tackle this shift assume the presence of training as well as testing data apriori to calculate the importance weights which are then used in the learning algorithm to reduce the mismatch. This major problem makes them impractical. In this paper, we tackle this, using marginalized stacked denoising autoencoder (mSDAs) while using the data from seven healthy subjects recorded over eightsessions distributed over four weeks. We compare our results with kernel mean matching, a popular approach for covariate shift adaption. Using support vector machines for classification and reduced complexity of mSDA, we get promising accuracy.

Key words: Electroencephalogram, transfer learning, marginalized stacked denoising autoencoders, covariate shift adaptation.

1. Introduction

Electroencephalography (EEG) is used for recording electrical activity of the brain through a number of electrodes placed on the scalp. The electrodes measure voltage fluxes that are produced in the neurons of the brain by ionic stream. The EEG has traditionally been used for diagnostics of various diseases such as epilepsy and monitoring of the brain during various states such as sleep. The EEG can also be used for control applications, since various components and brain potentials such as the movement-related cortical potential (MRCP) and sensorimotor rhythms, can be voluntarily controlled. Using the EEG to control an external device is known as a Brain-Computer Interface (BCI). The BCI consists of different main components; a signal acquisition block (EEG is recorded), pre-processing (the signal-to-noise ratio is enhanced), feature extraction and classification (the intent of the user is determined), and then the output of the classifier is transformed into a device command. This command could be left/right or up/down movements of a cursor on a screen or initiating electrical stimulation of paralyzed muscles to produce a movement in e.g. stroke patients or spinal cord injured patients. To detect or decode the BCI user's intention, the BCI system must be calibrated, i.e. training the classifier. This is a tedious task that requires much time and can be exhausting for the user, since this is time that has to be spent before the BCI system can be used. This has to be done at the very moment often, because the brain is constantly changing especially as a result of mental and physical fatigue, which happens during a single session where the BCI system is being used. Moreover, the EEG changes also occur, over different sessions or days due to factors such as ambience changes in the recording environment, the non-stationary nature of the signals, slight changes in electrode placement, and environmental artifacts from external distractions. These situations are often encountered and lead to discrepancies between training and testing data which casts negative effect on the BCI system performance (the classification accuracies decrease). Due to these differences in the conditions between and within EEG recording sessions, data shifts in training and test distributions occur. This is one of the basic reasons why BCI applications do not provide the expected outcome in the real world. To address this problem datasets can be regularized using transfer learning, which uses prior knowledge to address the issue of how to use labeled data in a source domain to solve relevant but different problems in a target domain, even when the training and testing distributions are different.

Transfer learning is an approach that inherits knowledge from one task or domain and transfers it to another related task or domain for attaining a high performance. Domain adaptation can be divided into two types based on whether the testing data in the target distribution is unlabeled or semi labeled, resulting in unsupervised and semi supervised learning respectively. In the latter, labeled target data's correspondences are used for transformation between different domains [1] whereas the former uses the techniques in which transformation between domains follows a known class of methods including discriminative and distinguishing features, minimal difference in training and testing distributions [2] and a path for mapping of one domain on another [3].

Covariate shift is the condition in which the density distribution alters between train and test phase due to mismatch of environmental circumstances or variation of devices used to obtain training and testing samples. A. Gretton *et al.* [4] have proposed distribution matching for covariate shift adaptation. They suggested Gaussian kernel based mapping to reduce the mismatch in the transformed domain. Kai Zhang *et al*. [5] suggested distribution matching in the Hilbert Space using mapping of one domain onto another employing the surrogate kernel concept. Based on Mercer's theorem a surrogate kernel is developed using different aligned kernels. S. Pan provided surveys on transfer learning in [6] and [7] in which he emphasized on reviewing the transfer learning situations for supervised and unsupervised problems. The authors focused on the relation between transfer learning and other methods for domain adaptation such as covariate shift.

Difference in source and target data distributions is a major hindrance in getting predictive models. A lot of techniques have been developed for addressing training and testing mismatch. One approach to cope with the situation is that the training samples which are similar to testing ones are provided more weights and those training samples which are different are assigned less weights. This technique of providing weights to training data, so that training data may realize a better representation of testing data is known as importance weighting which is a very famous approach for unsupervised domain adaptation. Hassan *et al.*[8] has recently used this approach in acoustic emotion recognition which has provided important improvements in emotion analysis. They have employed three algorithms as transfer learning techniques (Kernel Mean Matching (KMM) [9], KullbackLeibler Importance Estimation Procedure [10] and Unconstrained Least Squares Importance Fitting [11]. J. Deng *et al.* [1] proposed an unsupervised domain adaption procedure based on auto-encoders. They have used an adaptive autoencoder approach to learn common feature representations between training and test distributions for speech emotion recognition.

Denoising auto encoders learn representations of data by reconstructing features in the data which are corrupted by noise artificially. P. Vincent *et al.* [12] used them to extract robust features. Stacked denoising auto encoders are stacking of denoisers in a deep learning architecture. Glorot *et al.* [2] used stacked denoising autoencoders (SDAs) for domain adaptation for learning good feature representations. On sentiment analysis, a promising accuracy has been recorded using SDAs. H. Lee, *et al.* [13] used neural networks for audio classification. They used support vector machines (SVMs) for classification using the output of intermediate layers as features.Despite the promising and captivating results SDAs have certain limitations. Since they rely on iterative optimization for learning model parameters, they have high computational cost and require large training time. The challenge is to cater for the computationally intensive model selection.

In this paper, we used marginalized stacked Denoising Autoencoders (mSDAs) to address covariate shift adaptation in the EEG signals. MSDAs rely on an unsupervised domain adaptation procedure where previous information obtained from a target set is utilized to regularize the learning on source data. They constitute layer by layer training of SDAs and have denoisers as basic building blocks. MSDAs aim to marginalize the noise to eliminate the need of an optimization algorithm such as gradient descent which is required in conventional SDAs.

2. Literature Review on Importance Weighting

2.1. Verifying a Distribution Shift

To verify the presence of covariate shift in the data, we have used the Kolmogorov-Smirnov (K-S) test. This non parametric method checks whether the drawn samples are from same distribution or not. The K-S value counts a distance between the empirical density occupation of the training and testing samples. If the samples belong to the same distributions, we get the null hypothesis H_0 whereas $H_a$is obtained in the case of a covariate shift when we have different distributions. The K-S test is distribution free so it is very interesting to use it for verifying the presence of a shift in distributions. It can give authentic results when we have single dimensional data. For multi-dimensional data, we consider each jthfeature as independent of the remaining. Gretton *et al.* [14] has proposed a test for covariate shift identification of multi-dimensional data which can be used if we do not want to make the above mentioned assumption. In this, test data is mapped into a space of higher dimensions and then the difference in means of both distributions is obtained. We have used the K-S test because of the complexity of the test proposed by Gretton. The Ktest2 function is used for applying K-S test on each feature. This function is available in a Matlab Toolbox. When the test is passed, 0 is returned and it indicates that the data do not contain any shift.

2.2. Importance Weighting

Importance weighting employed for covariate shift adaptation assigns more weights to those training samples which provide closer representation to the test data. Consider we have a given domain X and n training samples $\{x_i^{tr}\}_{i=1}^{n_{tr}}$ are independently drawn from a particular distribution $p_{tr}(x)$and n testing samples$\{x_j^{te}\}_{j=1}^{n_{te}}$are independently drawn from a distribution having density $p_{te}(x)$.

$$\{x_i^{tr}\}_{i=1}^{n_{tr}} \xleftarrow{i.i.d} p_{tr}(x)$$

$$\{x_i^{te}\}_{i=1}^{n_{te}} \xleftarrow{i.i.d} p_{te}(x)$$

For minimizing the discrepancy between the training and test distributions, the objective is to obtain importance weights β which can be defined as a ratio of testing data density to that of training data; it is non negative as per definition "$\beta = \frac{p_{te}(x)}{p_{tr}(x)}$". Actually we are driving our learning scheme towards significant regions in the available data. Dense portions in testing distributions will provide more weights hence the algorithm is pushed to important regions. So the issue is to obtain the importance weights to address covariate shift.

2.3. Kernel Mean Matching

KMM is independent of density calculation, hence huge amounts of data are not required. It reduces the difference in the means of test and training data distributions on which importance weighting has been applied in a high dimensional space. A kernel is used to induce this high dimensional space. KMM is based on maximum mean discrepancy method (MMD) given by Gretton *et al.* [14]. Let *p*represent density of one distribution and *q* represent the other distribution. $\mathrm{MMD}[\Phi, p, q]$ is zero if *p* is equal to *q* where Φ shows mapping kernel which induces a high dimensional space. The objective function is defined by (1).

$$J(\beta) = \min_{\beta} \left\| \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} \beta_i \Phi(x_i^{tr}) - \frac{1}{n_{te}} \sum_{i=1}^{n_{te}} \Phi(x_i^{te}) \right\|$$
$$subject\,to\ \beta_i \in [0, \mathrm{B}]\ and \left| \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} \beta_i - 1 \right| \le \varepsilon \qquad (1)$$

where β and ε are the parameters for optimizing the objective function. β and ε are greater than zero. The above function can be expanded as given in (2).

$$\frac{1}{n_{tr}^{2}} \beta^T K \beta - \frac{2}{n_{tr}^{2}} \kappa^T \beta + const. \qquad (2)$$

where $K_{ij} = \kappa(x_i^{tr}, x_j^{tr})$ and $\kappa_i = \frac{n_{tr}}{n_{te}} \sum_{j=1}^{n_{te}} k(x_i^{tr}, x_j^{tr})$ to get an appropriate β, the quadratic function is given in (3).

$$\min_{\beta} \frac{1}{2} \beta^T K \beta - \kappa^T \beta\ subject\,to\ \beta_i \in [0, \mathrm{B}]\ and \left| \sum_{i=1}^{n_{tr}} \beta_i - n_{tr} \right| \le n_{tr} \varepsilon \qquad (3)$$

These weights are used for weighting the training samples, so that they can realize a better representation of testing data. Appropriate tuning of the following three parameters is necessary for the algorithm to work efficiently: B, which is the upper limit of importance weights, the width σ of the Gaussian kernel and ε. Suggested values for the parameters as proposed by Gretton *et al.* [9] are: $\mathrm{B} = 1000$, $\sigma = 0.1$, $\varepsilon = (\sqrt{n_{tr}} - 1/\sqrt{n_{tr}})$. To obtain the importance weights from KMM by solving the quadratic function, we have used the Matlab toolbox for optimization. The generic form of quadratic function is given in (4).

$$\min_x \frac{1}{2} x^T H x + f^T x \; subject\,to \begin{cases} A.x \le b \\ A_{eq}.x \le b_{eq} \\ LB \le x \le UB \end{cases} \tag{4}$$

3. Proposed Methodology

We used mSDA for covariate shift adaptation in the EEG data.

3.1. Autoencoders

An autoencoder maps an input x to $h(x)$, so that the input can be reconstructed from $h(x)$ hence the target is the input itself. When training is done using theMSE criterion and number of linear hidden layers is one, having k units, then data are being represented as the first k components. When a nonlinear hidden layer is used, it does not behave like principal component analysis. An issue with this approach is that if there is an autoencoder with n input units and n or greater units in the hidden layer, then the learning function will constitute an identical mapping of input units to hidden layer, yielding no useful representation of input data. There are many ways in which autoencoders with more hidden units can be prevented from just copying the inputs into hidden units. One approach is to use a denoising autoencoder (DA) which reduces the error by reconstructing the input from its corrupted version.

3.2. Denoising Autoencoders

A simple autoencoder has two elements; one is the encoder and the other is the decoder. An input x is mapped to some hidden layer $h(x)$, using the encoder $h(.)$. The mapping of a hidden layer unit $h(x)$ to an output which reconstructed as x, is done by the decoder $g(.)$. The reconstruction error or loss is minimized by learning some parameters. A DA is a neural network constituting one layer and has a modification such that before the input data is mapped to some hidden representation, it is corrupted with certain noise. Then from this corrupted version $\tilde{x}$, input x is reconstructed by minimizing the loss. Corruptions may have different versions including additive Gaussian noise or masking noise. Vincent et al. [12] used binary masking noise in which a few input features are set to zero. This is typical in text representations where due to a difference in training and test domains or due to the author's writing style, words can be missing.

In SDA, DAs are stacked in multiple layers to develop a higher level data representation. It involves feeding the hidden representations of l_{th} DA as input to $(l+1)_{th}$ DA. Learning is done layer by layer using a greedy algorithm. In many scenarios, the features learned from stacked denoising autoencoders provide high performance accuracy. In sentiment analysis, Glorot *et al.* [2] used SDAs along with linear SVMs to learn features which provide remarkable accuracy. The SDAs have few limitations: a) It is computationally very intensive and uses long training time as it uses a stochastic gradient descent, b) Hectic cross validation is required for tuning parameters such as learning rate, noise level, structure of whole network, and batch size etc. This is computationally expensive in terms of cost and time constraints as a single run may require a long time, c) A unique solution of the optimization is not guaranteed as it depends on initialization. To solve these issues, M. Chen *et al.* [15] proposed marginalized SDA, that marginalizes the noise to calculate to the closed form solution.

3.3. Marginalized Stacked Denoising Autoencoders

We used a modified version of SDA, which is compatible with SDA in feature learning but with higher speedups, optimum computation and lowering the number of parameters which are required to be tuned. In marginalized stacked DAs the complexity is reduced and model selection also gets faster. The fundamental building unit of our approach is single layer DA. We take n inputs from a union of source and target domains and induce corruption in them by removing randomly selected features. Actually, each feature is set to zero with some probability which is greater than zero. If x is an input, its corrupted version is $\tilde{x}$.

In contrast to SDA, which usesencoder and decoder approach, reconstruction is done in single mapping, hence the reconstruction loss $\frac{1}{2n}\sum_{i=1}^{n}\left\|x_i - W\tilde{x}_i\right\|^2$ is minimized. We solve for Wto minimize the overall loss.

$$\iota_{sq}(w) = \frac{1}{2nm}\sum_{j=1}^{m}\sum_{i=1}^{n}\left\|x_i - W\tilde{x}_{i,j}\right\|^2 \qquad (5)$$

For simplification, it is assumed that a feature is added to input which is constant over all samples, $x_i = [x_i;1],$ and during mapping, the bias vector is introduced in mapping $\mathrm{W} = [\mathrm{W}, \mathrm{b}].$ Multiple runs are done on the training dataset each with different corruption level to reduce the variance in loss. To lower the variance, we perform several runs on the training data, each time with different corruption.

Consider the input framework $X = [x_1, \ldots x_n] \in R^{dxn}$ and its m-times repetition as $\overline{X} = [X, \ldots, X]$ where $\tilde{X}$ is the corrupted form of $\overline{X}$. Now the loss equation can be written as given in (6).

$$\iota_{sq}(w) = \frac{1}{2nm} tr[(\overline{\mathrm{X}} - W\tilde{X})^T(\overline{X} - W\tilde{X})] \qquad (6)$$

The closed form of the solution is $W = PQ^{-1}$ with $Q = \tilde{X}\tilde{X}^T$ and $P = \overline{X}\tilde{X}^T$. The solution to (6) is dependent on corrupted features and input samples.Ideally it is recommended that for all input samples, all possible corruptions should be considered so we would like $m \to \infty$. According to the weak law of large numbers, when m becomes very large and more copies of corrupted data are formed, convergence of P and Q to their expected values $\mathrm{E}[P], \mathrm{E}[Q]$ occurs. The closed form mapping of W can be expressed as $W = \mathrm{E}[P]\mathrm{E}[Q]^{-1}$. We will compute both $\mathrm{E}[P]$ and $\mathrm{E}[Q]$. First we focus on $\mathrm{E}[Q] = \sum_{i=1}^{n}\mathrm{E}[\tilde{x}\tilde{x}^T]$. If the two features i and j survived the corruption process, only then entry $[\tilde{x}\tilde{x}^T]$ is considered uncorrupted. This can only happen when probability is $(1-p)^2$.

Consider a vector $q = [1-p, \ldots 1-p, 1]^T \in R^{d+1}$ where the probability of feature i is represented by q_i. We define $S = XX^T$ as the scatter matrix of uncorrupted input data X. Now we can express $\mathrm{E}[Q] = \left\lfloor\begin{matrix} S_{ij}q_i & if\ i=j \\ S_{ij}q_i p_j & if\ i \neq j \end{matrix}\right.$ and $\mathrm{E}[P] = S_{ij}q_i$ in its closed form.

3.3.1. Stacking and induction of nonlinearity in feature generation

Now without using reconstruction of $\tilde{x}_i$, we can compute W in closed form. We refer to this closed form as Marginalized Denoising Autoencoder (MDA). The success of SDA is based on two important factors: nonlinearity and stacking of multiple DAs. Our approach is also capable of both where several MDA layers are stacked in a multiple layer architecture by feeding the hidden representations of l_{th} layer as input to $(l+1)_{th}$ layer. Closed form learning of each transformation map W^{l} is done to reconstruct the output of previous mDA h_l from its corrupted version. In SDAs the nonlinearity factor is added using a nonlinear encoder $h(.)$. Training becomes highly nonconvex when the encoder is learned with a reconstruction matrix W Parameters are to be tuned by running an iterative algorithm making this approach computationally extensive. We inject nonlinearity after computation of W by applying a nonlinear function on output of each mDA. Each layer is obtained by a nonlinear mapping $h_l = \tanh(W^l h_{l-1})$ of the preceding layer, with $h_0 = x$ showing input.

4. Dataset and Experimental Setup

To examine the performance of the proposed method we used MRCPs recorded from seven healthy subjects. Each subject was seated in a chair in a shielded room for recording EEG during dorsiflexion of the ankle. Two types of dorsiflexions were performed; fast and slow. The fast movement was 0.5s to reach a target force of 20% of maximum voluntary contraction (MVC), whereas the slow movement were 3s to reach the same target force of 20% MVC. Each of the movement types were repeated 30 times with a 10 s rest period in between. Using a Nuamp amplifier, 10 channels (C3, C4, F3, F4, P3, P4, Cz, Fz, Pz and FP1) of monopolar EEG were recorded by placing electrodes on scalp. The sampling frequency was 500 Hz, and the signals were digitized with 32 bits accuracy. The recordings were performed in different sessions with a separation of at least two days but less than a week between two consecutive sessions. Therefore, for every week, we have EEG recordings for two days as shown in Table 1. Each subject followed this procedure.

Table 1. Overview of Recording Setup

	Week 1		**Week 2**		**Week 3**		**Week 4**	
Days	1	2	1	2	1	2	1	2

Each session's data have recordings for two classes: *fast* and *slow*. The EEG signals were processed by applying a band pass filter between 0.05 & 5 Hz with 2nd order Butterworth filter. A Surrogate channel was obtained by applying a large Laplacian spatial filter on the 9 EEG channels with Cz as the center electrode (FP1 was used to register electrooculography).

5. Evaluation Setup and Results

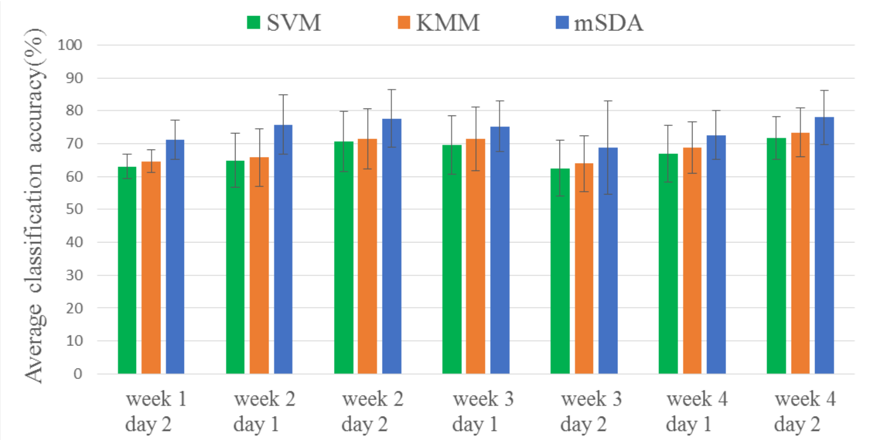

Fig. 1. Average classification accuracy for the seven subjects using test setup I.

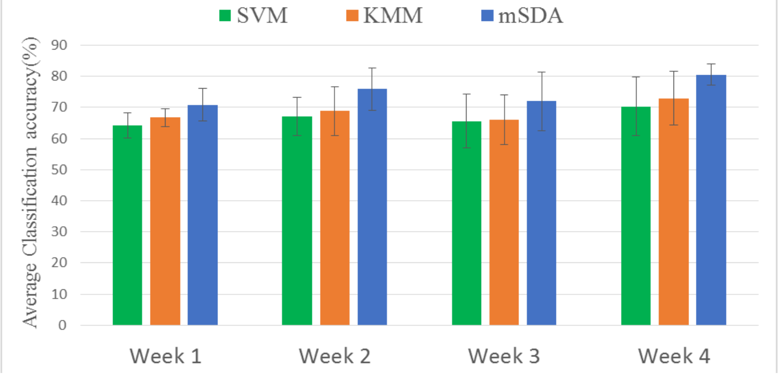

Fig. 2. Average classification accuracy for the seven subjects using test setup II.

We evaluate the performance of our system using a binary classification accuracy under two test setups; *Test Setup I*: For all subjects we use all previous data for training and current day data for testing (accumulative training), *Test Setup II:* For all subjects we worked on each week and used data of day 1 for training and day 2 for testing. Linear SVMs with fixed parameters are used for classification. **Fig**. 1 and Fig. 2 show the average classification accuracy along with the standard deviation for *Test Setup I* and *Test Setup II* respectively.

6. Discussion

The results in Fig. 1 and Fig. 2 show that the EEG recorded in multiple sessions does have a shift in the data. This shift can be removed by explicitly addressing this issue using importance weighting techniques. The standard statistical methods (KMM) developed for minimizing this shift in the data clearly outperforms the standard SVMs supporting our argument that this shift in the data needs to be addressed. However, with the advent of a new breed of algorithms like SDAs under the umbrella of deep learning are showing far better performance than the typical algorithms. In this paper, we have also shown that the EEG data that are recorded in different sessions, is a perfect problem to be tackled under the transfer learning domain.

References

[1] Deng, J., Zhang, Z., Eyben, F., & Schuller, B. (2014). Autoencoder-based unsupervised domain adaptation for speech emotion recognition. *IEEE Signal Processing Letters*, *21(9)*, 1068-1072.

[2] Glorot, X., Bordes, A., & Bengio, Y. (2011). Domain adaptation for large-scale sentiment classification: A deep learning approach. *Proceedings of the 28th International Conference on Machine Learning (ICML-11)* (pp. 513-520).

[3] Gopalan, R., Li, R., & Chellappa, R. (2014). Unsupervised adaptation across domain shifts by generating intermediate data representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36(11)*, 2288-2302.

[4] Gretton, A., Borgwardt, K. M., Rasch, M., Schölkopf, B., & Smola, A. J. (2006). A kernel method for the two-sample-problem. *Advances in Neural Information Processing Systems*, 513-520.

[5] Zhang, K., Zheng, V. W., Wang, Q., Kwok, J. T. Y., Yang, Q., & Marsic, I. (2013). Covariate shift in hilbert space: A solution via sorrogate kernels. *ICML, (3)*, 388-395.

[6] Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, *22(10)*, 1345-1359.

[7] Pan, S. J., Tsang, I. W., Kwok, J. T., & Yang, Q. (2011). Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, *22(2)*, 199-210.

[8] Hassan, A., Damper, R., & Niranjan, M. (2013). On acoustic emotion recognition: compensating for

covariate shift. *IEEE Transactions on Audio, Speech, and Language Processing*, *21(7)*, 1458-1468.

[9] Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., & Schölkopf, B. (2009). Covariate shift by kernel mean matching. *Dataset Shift in Machine Learning*, *3(4)*, 5.

[10] Sugiyama, M., Nakajima, S., Kashima, H., Buenau, P. V., & Kawanabe, M. (2008). Direct importance estimation with model selection and its application to covariate shift adaptation. *Advances in Neural Information Processing Systems*, 1433-1440.

[11] Kanamori, T., Hido, S., & Sugiyama, M. (2009). Efficient direct density ratio estimation for non-stationarity adaptation and outlier detection. *Advances in Neural Information Processing Systems*, 809-816.

[12] Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P. A. (2008, July). Extracting and composing robust features with denoising autoencoders. *Proceedings of the 25th International Conference on Machine Learning* (pp. 1096-1103). ACM.

[13] Lee, H., Pham, P., Largman, Y., & Ng, A. Y. (2009). Unsupervised feature learning for audio classification using convolutional deep belief networks. *Advances in Neural Information Processing Systems,* 1096-1104.

[14] Gretton, A., Borgwardt, K. M., Rasch, M., Schölkopf, B., & Smola, A. J. (2006). A kernel method for the two-sample-problem. *Advances in Neural Information Processing Systems,* 513-520.

[15] Chen, M., Xu, Z., Weinberger, K., & Sha, F. (2012). Marginalized denoising autoencoders for domain adaptation.

Farah Abid received her B.E. degree in computer engineering from Mirpur University of Science and Technology (MUST), AJK, Pakistan in 2014. She has done the M.S. at National University of Science and Technology, College of Electrical and Mechanical Engineering, Pakistan in 2016. Her research interests include applications of machine learning to signal processing, image processing and brain computer interface.

Ali Hassan received his B.E. and M.S. degrees in computer engineering from the National University of Sciences and Technology (NUST), College of Electrical and Mechanical Engineering, Pakistan, in 2004 and 2007, respectively. He received the Ph.D. degree in electrical engineering from the University of Southampton, UK, in 2012. He is currently working at NUST College of Electrical and Mechanical Engineering as an Assistant Professor at the Department of Computer Engineering. His research interests include application of machine learning to speech and image processing in the domains of speech, texture classification and biomedical engineering.

Anum Abid received her B.E. degree in electrical engineering from University of Engineering And Technology, Taxila (UETT), Pakistan in 2016. She is currently doing her M.S. at University of Engineering and Technology, Taxila (UETT), Pakistan. Her research interests include applications of Machine Learning to signal processing, brain computer interface, image processing and electrical power.

Imran Khan Niazi's research interests focus on rehabilitation engineering. During his PhD, he was part of a multidisciplinary team headed by Prof. Dario Farina team where focus of our group was use of brain computer interface (BCI) system for stroke rehabilitation using specifically movement related cortical potential. During his PhD studies, his interest is in the field of neural plasticity grew, especially as he worked with stroke patients and saw the importance of neural plasticity in rehabilitation. Since his PhD he has been working on the novel use of signal processing methods for early detection of movement intention and also classifying which kind of movement subjects are intending.

Mads Jochumsen did his MSc and PhD in biomedical engineering from Aalborg University, Denmark. He is currently working as an assistant professor at Aalborg University. His research interests include brain–computer interfacing.