# Group Learning for High-Dimensional Sparse Data

Vladimir Cherkassky[1,2], Hsiang-Han Chen[2], Han-Tai Shiao[1]
[1]Department of Electrical and Computer Engineering
[2]Bioinformatics and Computational Biology
University of Minnesota, Twin Cities
Minneapolis, Minnesota 55455, U.S.A.
{cherk001, chen4646, shiao003}@umn.edu

*Abstract*—**We describe new methodology for supervised learning with sparse data, i.e., when the number of input features is (much) larger than the number of training samples (*n*). Under the proposed approach, all available (*d*) input features are split into several (*t*) subsets, effectively resulting in a larger number (*t*n*) of labeled training samples in lower-dimensional input space (of dimensionality *d/t*). This (modified) training data is then used to estimate a classifier for making predictions in lower-dimensional space. In this paper, standard SVM is used for training a classifier. During testing (prediction), a group of *t* predictions made by SVM classifier needs to be combined via intelligent post-processing rules, in order to make a prediction for a test input (in the original d-dimensional space). The novelty of our approach is in the design and empirical validation of these post-processing rules under Group Learning setting. We demonstrate that such post-processing rules effectively reflect general (common-sense) a priori knowledge (about application data). Specifically, we propose two different post-processing schemes and demonstrate their effectiveness for two real-life application domains, i.e., handwritten digit recognition and seizure prediction from iEEG signal. These empirical results show superior performance of the Group Learning approach for sparse data, under both balanced and unbalanced classification settings**

***Keywords—binary classification, digit recognition, feature selection, histogram of projections, Group Learning, iEEG, seizure prediction, SVM, unbalanced data.***

## I. Introduction and Motivation

Sparse high-dimensional data sets are common in many machine learning problems where the dimensionality of data samples *d* is much larger than the training sample size *n*. Example applications include gene microarray analysis, image based object recognition, functional magnetic resonance imaging (fMRI), etc. In microarray data analysis, technologies have been designed to measure the gene expression levels of tens of thousands of genes in a single experiment. However, the sample size in each data set is typically small, ranging from tens to low hundreds due to high cost of measurements. Similarly, in brain imaging studies the dimensionality of the input data vector is very high (the number of voxels $d \sim 10000$), but only a few hundred of two-dimensional (2-D) or 3-D images ($n \sim 100$) are available. Such sparse high-dimensional data sets present new challenges for classification learning methods.

Most approaches for learning with high-dimensional data focus on improving learning methods by incorporating a priori knowledge about application domain [1]–[3]. Among all these approaches, using feature selection to reduce the dimensionality of data is the most common approach for modeling sparse high-dimensional data. These approaches include *filter methods* (when feature selection is performed as part of pre-processing, prior to learning) and *wrapper methods* (when feature selection is implemented as a part of learning method) [4]–[8].

Alternatively, we propose a methodology for learning with high-dimensional data *without* feature selection. Our approach is to split all features into several groups, so that a high-dimensional feature vector $x \in X$ can be represented as several lower dimensional vectors $x' \in X'$, where $x \in R^d$, $x' \in R^{d'}$, and $d' < d$. The process of learning, including model estimation and prediction, will take place in a lower dimensional space $X'$, rather than the original space *X*. In order to illustrate the proposed strategy, consider the task of handwritten digit recognition for digits 5 versus 8 in MNIST dataset. Each digit (5 or 8) is a 28×28 pixels image represented as a real-valued vector of size 28 × 28 = 784, i.e., $x \in R^{784}$. Within this vector, each of the 784 components (features) represents the pixel intensity (via 8-bit integer). The training examples of digits 5 and 8 are labeled as negative and positive.

We can partition an image into four non-overlapping patches, each of size 14×14, as illustrated in Fig. 1. More formally, each training image *x* is now represented as 4 sub-images $\{x_1', x_2', x_3', x_4'\}$, where $x' \in R^{196}, j = 1,\ldots,4$. All 4 sub-images preserve the same labeling as an original training sample. Then we estimate *one* classifier using all labeled sub-images in space $X'$. Note that pixels in space $X'$ are more highly correlated than pixels in the original space *X* as they are close to one another. Further, note that the mapping from the original pixel space onto new space (of sub-images) effectively reflects a priori knowledge about application data (e.g., local spatial correlation of pixels in natural images). This approach results in the reduction of the dimensionality of the feature space, while keeping all features. Further, the number of training samples is effectively "increased" without utilizing additional labeled training data. This may be contrasted to standard feature selection methods [1]–[7], where:

- dimensionality reduction may result in lost information (due to discarded features);

- the number of training samples remains the same.

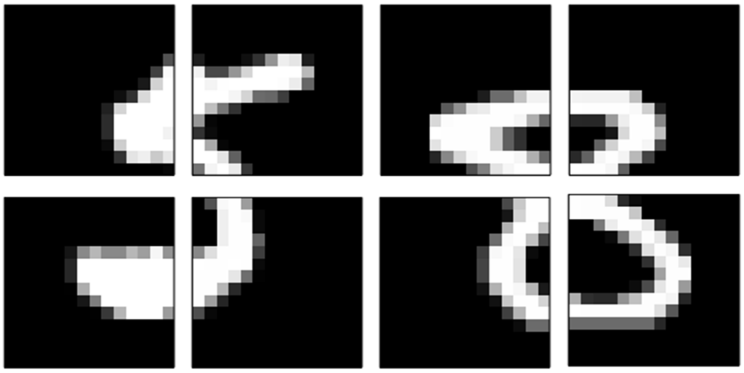

**Fig. 1.** Partitioning of digit 5 and 8 into four sub-images (patches) of size 14x14.

During testing (or prediction) stage, the goal is to make an accurate prediction for the test input x (in the original feature space), i.e., $x \in R^{784}$ in the example above. This requires combining $t$ individual predictions made by the trained classifier for 4 sub-images. That is, we need to specify a post-processing procedure (a set of rules) for combining 4 predictions. A proper post-processing procedure should provide good prediction performance. As usual, good/superior prediction performance can be achieved only for data sets with certain properties (aka a priori knowledge). We refer to this approach as *Group Learning*. The goal of this paper is to show that:

(1) The Group Learning approach provides competitive performance for sparse high-dimensional data.

(2) It is possible to provide very general statistical characterization for such data sets.

(3) It is possible to design effective post-processing rules for data sets with statistical characteristics (2). The effectiveness of these rules is demonstrated using several real-life data sets.

The rest of this paper is organized as follows. Section II presents formal description of the Group Learning method and its motivation using Vapnik-Chervonenkis (VC)-theoretical arguments. Section III describes two proposed post-processing schemes appropriate for two different characterizations of sparse high-dimensional data sets: (a) distinct/non-overlapping class distributions (~ MNIST handwritten digits), and (b) data sets with heavily overlapping class distributions (such as iEEG data for seizure prediction). Section IV presents empirical comparisons for several data sets. These comparisons include both balanced data sets and heavily unbalanced data sets. Section V relates the proposed Group Learning to Convolutional Neural Networks (CNN) or methods generally known as Deep Learning. Finally, Section VI presents summary and conclusion.

## II. Group Learning Approach

This section provides formalization of Group Learning for sparse high-dimensional data. All descriptions assume binary classification problem. Suppose every *d*-dimensional training sample is represented as *t* samples of dimensionality *d/t*. Specifically, each training sample (*x, y*), where *x* ∈ *X* and *y* ∈ {+1, −1}, is represented as $(x_1', y), (x_2', y), \dots, (x_t', y)$ where $x'_j \in X', j = 1, \dots, t$. Here, $x'_j \in R^{\frac{d}{t}}$, $x \in R^d$, and *d/t* < *d*. The transformation above is equivalent to splitting *d* features into *t* subsets or groups. Technically, it can be done by first ordering the features based on application domain knowledge (a priori knowledge), as in digit recognition example in Fig. 1.

The training and test stages for standard supervised learning are illustrated in Fig. 2. During training, the goal is to estimate a classifier *f* (*x*) using labeled training samples. This classifier should give good prediction for new unlabeled test inputs. Both training and test tasks are performed in space *X*.

Fig. 3 illustrates the Group Learning approach to the same binary classification problem. We assume that the original features (space *X*) are represented as 3 disjoint groups of features (*t* = 3). That is, a labeled training sample *x* is first transformed into three lower dimensional ones, i.e., $x'_1$, $x'_2$, and $x'_3$ . Then the Group Learning approach estimates a classifier $f(x')$ using all labeled training samples $x'$ (3*n* total). During prediction (test) stage, an unlabeled test sample *x* is also transformed from space *X* to space $X'$. Therefore, applying the estimated classifier to all "shortened" test inputs $x'_j$ , *j* = 1, 2, 3, will result in three predictions $\hat{y}_1$, $\hat{y}_2$, and $\hat{y}_3$. In order to obtain a prediction for the test sample *x*, a post-processing procedure for reconciling $\hat{y}_1, \hat{y}_2$, and $\hat{y}_3$ into $\hat{y}$ is needed. Such post-processing rules are discussed in Section III.

In summary, under Group Learning, a training data set of size *n* is transformed into *tn* labeled samples (in space *X'* of lower dimensionality *d/t*). Then a single classifier is estimated using *tn* training samples (in this low-dimensional space). In contrast, standard machine learning approach (e.g., SVM) estimates a classifier using *n* training samples, in *d*-dimensional space. Note that the objective of Group Learning is the same as for all standard supervised learning methods, i.e. to make accurate predictions for test inputs in space *X* (not accurate predictions for test inputs in space $X'$ ). Hence, during

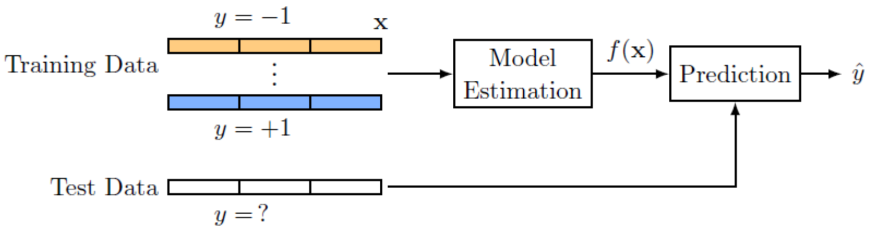

**Fig. 2.** Standard inductive learning approach: a classifier is estimated from labeled examples in space *X*, and makes prediction for test inputs in space *X*.

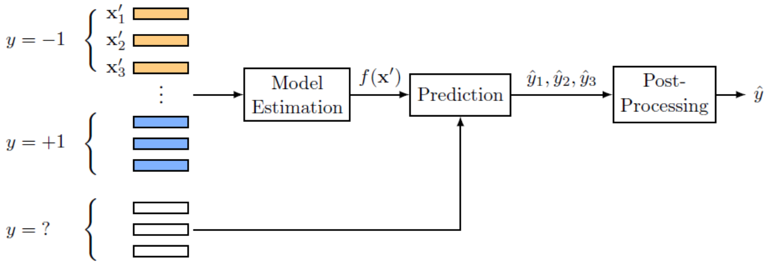

**Fig. 3.** Group Learning approach first transforms the data from space *X* to space $X'$. Both model estimation and prediction are performed in space $X'$. Prediction for test input in the original space *X* is made by combining *t* predictions (via postprocessing rule). The illustration is for *t* = 3.

testing (prediction) stage predictions for several $x'_j$ (a group of *d/t* features) are combined into a single prediction $\hat{y}$.

Next, we provide a qualitative analysis of the proposed Group Learning approach via analysis of VC-generalization bound [9], [10]. According to VC theory, for binary classification problems, the following bound for generalization (test) error holds with probability of at least $1-\eta$ for all admissible functions $f(x,\omega)$, including the function $f(x,\omega^*)$ that minimizes empirical risk:

$$R(\omega) \leq R_{emp}(\omega) + \Phi(h,n,\eta) \qquad (1)$$

where

$$\Phi(h,n,\eta) = \sqrt{\frac{h\left(log\frac{2n}{h}+1\right)-log\frac{\eta}{4}}{n}} \qquad (2)$$

Here, *h* denotes the VC-dimension, *n* is the training sample size. The term Φ is called *the confidence interval*, since it represents the difference between the training error $R_{emp}(\omega)$ and the test error $R(\omega)$ of a classifier.

Consider the behavior of Φ as a function of sample size *n*, with all other parameters fixed. Equation (2) shows strong dependency of the confidence interval Φ on *n/h*, the ratio of the number of training samples to the VC-dimension [1,4,9]. Thus, we can distinguish two extreme settings depending on *n/h* :

(1) *Large sample size*, when this ratio is very large, and the value of the confidence interval Φ becomes small. In this case, the goal of learning is minimization of empirical risk (training error), because the training error can be safely used as a measure of prediction risk (test error). This also implies that for large *n/h*, a classifier with small training error $R_{emp}(\omega)$ provides good generalization.

(2) *Small (or finite) sample size*, when the ratio *n/h* is small, so the value of the confidence interval cannot be ignored. In this case, one needs to select optimal VC-dimension, in order to achieve best generalization (aka model selection problem in machine learning).

Obviously, for sparse high-dimensional data sets learning is performed in the *second regime*. The proposed Group Learning approach effectively increases the training sample size while reducing dimensionality, so it tends to transform the learning problem towards the *first regime* (~ larger sample size). The benefits of this strategy are twofold. First, the features (within each group) are highly correlated, so we can reduce dimensionality from *d* to *d/t* without losing important information. Second, the training sample size "increases" from *n* to *tn*. Hence, under Group Learning, the ratio of the number of training samples to VC-dimension is increased from *n/d* to $t^2n/d$. However, our analysis is purely *qualitative* because VC-theory assumes independent identically distributed (IID) samples, but samples under Group Learning are not truly IID.

## III. Proposed Post-processing Rules

This section describes two different post-processing procedures (for Group Learning) suitable for sparse data sets with two distinctly different statistical characteristics. This characterization assumes binary classification problem setting. The first type refers to sparse data sets formed by non-overlapping or slightly overlapping class distributions. The second type describes inherently more difficult sparse data sets with heavily overlapping class distributions. The difference between these two types of data sets can be easily seen when training a classifier using standard SVM. That is, for slightly overlapping class distributions, a properly trained SVM classifier will show large separation margin. Moreover, the distribution of distances between training samples and SVM decision boundary will form two separate clusters, for positive and negative training samples. This distribution of distances, aka the 'histogram-of-projections', is very useful for understanding properties of SVM models [1], [4]. On the other hand, for heavily overlapping class distributions there will be small separation between positive and negative training samples when using the histogram of projections (for trained SVM classifier). These differences motivate different post-processing procedures, as detailed next.

### *A. Lightly Overlapping Class Distributions*

First, consider sparse data sets where samples (from two different classes) can be easily discriminated. A good example is the handwritten digits from MNIST data set. The visual difference between (different) digits can be easily recognized, by both humans and computer models (see visual examples in Section I). We refer to such data as *non-overlapping* or *lightly overlapping* class distributions. For such data sets, properly trained SVM classifier will show large separation margin. For example, using digits data in Fig. 1, when Group Learning is applied for SVM training using labeled sub-images, positive and negative sub-images will be well-separated (with large margin). So during testing stage, we can apply popular post-processing, such as *majority voting*, to combine predictions for *t* sub-images and derive a single prediction for test input. That is, for *t* = 4, at least three sub-images should be predicted as 'positive' in order to derive 'positive' prediction for the full image. Further, in the event of tie, prediction for test input is chosen randomly.

### *B. Heavily Overlapping Class Distributions*

Let us consider sparse data sets where samples from different classes *cannot* be easily discriminated. For example, suppose each sample *x* is a 'matrix-of-digits' formed by *t* = 800 handwritten digits. Each digit is represented by 14x14 pixel values (downsampled from the original 28x28 pixel image). See Figs. 4 and 5. There are two types of matrices corresponding to the positive and negative class:

- *Positive class:* matrix includes 720 images of even digits ('0, 2, 4, 6, 8') and 80 images of digit '1' as illustrated in Fig. 4.
- *Negative class:* all images in the matrix are even digits as shown in Fig. 5.

The images are drawn randomly from the MNIST dataset and randomly placed in the matrices. Under Group Learning, each digit corresponds to one group, so during training the input dimensionality is reduced by factor of *t* = 800; however during testing (prediction) we need to combine 800 predictions in order to classify unlabeled test input (~ matrix-of-digits). For this data set, the class distributions are *heavily overlapping*; so this classification problem is much harder for both human and machine learning algorithms.

Note that the locations of images in the matrix do not affect the classification results under our Group Learning framework. The two classes are heavily overlapping because 90% of sub-images share the same characteristics (e.g., even digits), and only 10% of the sub-images (~ digit '1' for positive class) are different. Also, the dimensionality of image matrices (14×14×800) is very high. These properties present difficulties for standard machine learning classification methods.

### Adaptive post-processing scheme for Group Learning

Under Group Learning, each input sample (~matrix) is modeled as 800 sub-images (digits) - all having the same label (positive or negative). So SVM training is performed in 14×14-dimensional space. Then for each test input, SVM classifier makes 800 predictions that can be displayed as a histogram of projections [1,4]. A typical histogram of projections for such 'matrix-of-digits' data set is shown in Fig. 8, for standard SVM classifier trained using 40 samples (matrices) per class. As one can see, the histograms-of-projections for both training and test inputs are highly overlapping and non-separable. For such data, commonly used post-processing procedures like the majority voting scheme (over 800 predictions) are not effective. More generally, post-processing rules based on combining individual predictions $\hat{y}_j$ for each digit would not be effective. Instead, we focus on post-processing rules utilizing *global statistical characteristics* of all (800) SVM predictions. This procedure is based on assumption that global statistical properties of the classifier outputs (for two classes) are sufficiently different and robust, so they can be useful for separating the two classes. That is, the threshold (for combining *t* predictions) during *testing* stage is determined by global statistical characteristics of *training sub-images*. Specifically, this paper uses *the mean of the classifier outputs for all t training inputs (sub-images)*, defined as $\mu_i = \frac{\sum_{j=1}^{t} f(x'_{ij})}{t}$, where $f(x'_{ij})$ denotes the classifier output for each sub-images $x'_{ij}$ forming the training input $x_i$. Classification decision (for separating training inputs) depends on $\boldsymbol{\mu}$ ~ the distribution of all $\mu_i$ – values for inputs from the same class. We assume this to be *negative class*, for the sake of discussion. Specifically, we define the decision threshold $Q$ in terms of the *quantile level* $p$ of all $\mu_i$-values for negative training samples, denoted as $Q(\boldsymbol{\mu}, p) = Q$ . For example, threshold level $Q(\boldsymbol{\mu}, 0.75)$ indicates that 75% of all the mean values $\mu_i$ (for negative training samples) are below this threshold. The optimal quantile level $p*$ is determined (adaptively) using only training data. That is, optimal $p$ –value provides minimum *false-negative* (FN) error rate for training data. For unbalanced training data, we assume that the "majority" class is negative, since global statistical indices of distribution $\boldsymbol{\mu}$ will be more robust for the majority class. In the rest of the paper, we use $Q^*$ to denote this optimal threshold estimated using only training data, i.e. $Q(\boldsymbol{\mu}, p^*) = Q^*$, where $\boldsymbol{\mu}$ denotes empirical distribution of all $\mu_i$-values of negative training samples, and $p*$ is the optimal quantile level.

Then during testing (prediction) stage, given a test input with *the mean value of SVM classifier outputs for test sub-images* denoted as $\mu_T$, the postprocessing rule (for prediction) is:

- Test input is positive if its $\mu_T > Q^*$, otherwise it is negative.

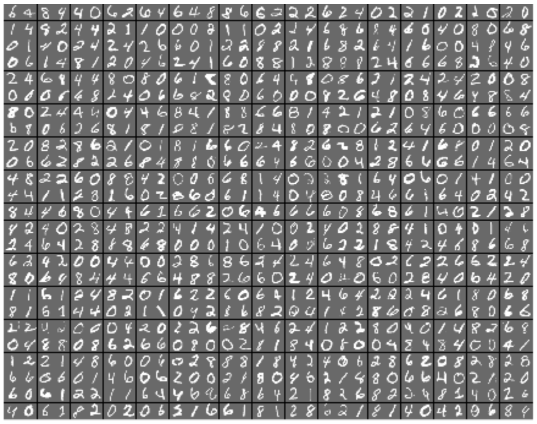

**Fig. 4.** Example of the positive class matrix with 800 images which include 720 even digits ('0, 2, 4, 6, 8') and 80 digits '1'.

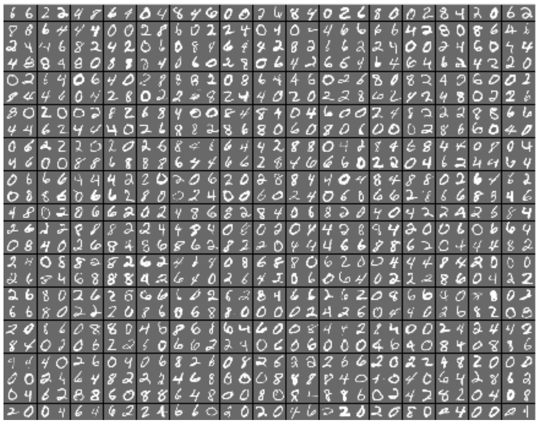

**Fig. 5.** Example of the negative class matrix with 800 images which are all even digits ('0, 2, 4, 6, 8').

For the 'matrix of digits' data set formed by 800 digits, using such post-processing rules based on the mean of SVM classifier outputs provide very robust discrimination between two heavily overlapping class distributions. Empirical results shown later in Fig. 9 illustrate good separation between the mean values (for samples from different classes) for *both* training and test data. Good separability for test data implies good prediction performance of the Group Learning approach.

The proposed adaptive post-processing (for Group Learning setting) can be used for both *balanced* and *unbalanced* data settings, assuming negative training samples correspond to majority class. This is because global statistical indices (used for proposed post-processing method) are more stable for the majority class. We will keep this in mind when presenting empirical results in Section IV.

Many applications with highly unbalanced data (e.g., fraud detection, medical diagnosis of abnormal heart conditions etc.) require *high sensitivity* for positive (minority class) predictions. In these cases, it may be reasonable to suggest the following *high-sensitivity post-processing rule* for Group Learning:

> During testing, if *any single* prediction $f\left(x'_{Tj}\right)$ for one sub-image $x'_{Tj}$ from the test input $T$ is positive, then classify the whole test input $T$ as positive, otherwise, it is negative.

We show empirical comparisons between these three post-processing methods (*majority voting*, *adaptive post-processing*, and the *high-sensitivity post-processing*) for heavily overlapping data in Section IV-B.

## IV. Empirical Comparisons

This section presents empirical results for data sets with non-overlapping and heavily overlapping class distributions to illustrate the effectiveness of the Group Learning approach. We use standard linear SVM classifier as a benchmark for proposed Group Learning. Using *linear* SVM is justified due to low-sample size and high dimensionality of the data. Introducing nonlinearity (via kernels) does not improve prediction performance for such data sets. All empirical comparisons follow the same experimental design shown in Fig. 2 (for standard SVM) and in Fig. 3 (for Group Learning). In all experiments, the classifier is linear SVM estimated using training data and a separate validation data to select optimal complexity parameter $C = [2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^{0}, 2^{1}, 2^{2}, 2^{3}, 2^{4},]$. Classification performance (test error) is evaluated using (out-of-sample) test data. For unbalanced data, we used the following experimental set-up [11]: unbalanced training data (positive vs. negative = 1:8), but balanced validation and test data sets. This setting effectively implements different misclassification costs for training (model estimation) and tuning complexity parameter *C*.

### A. *Balanced data with non-overlapping class distributions*

Handwritten digit images 5 and 8 of size 28×28 were used in this experiment. Each image is partitioned into 4 patches (t=4) as shown in Fig. 1. The goal is to compare prediction performance of standard SVM classifier (trained on the whole digit images vs. the Group Learning method (with t=4) using the same training data. See Figs. 2 and 3. Under both learning strategies, model estimation is performed using *linear* SVM.

Specifically, this experiment used the following data sets:

(1) Full image of digit 5 ~ negative class; digit 8 ~positive;

(2) number of training images: 10 (5 per class);

(3) number of validation images: 20 (10 per class);

(4) number of test images: 500 (250 per class).

All experiments used independent training, validation and test data sets. Specifically, optimally chosen parameter *C* yields the lowest error rate for validation data.

Each experiment is repeated 50 times using different (randomly selected) training and validation data sets, and the average test error is reported. We report separately false-positive (FP) and false-negative (FN) error rates for test data. Here FP error refers to incorrect prediction for test digit 5, and FN error indicates incorrect prediction for test digit 8.

Further, the number of training samples (per class) is increased from 5 to 25, in order to investigate its effect on test error rate. The size of the validation set is adjusted accordingly.

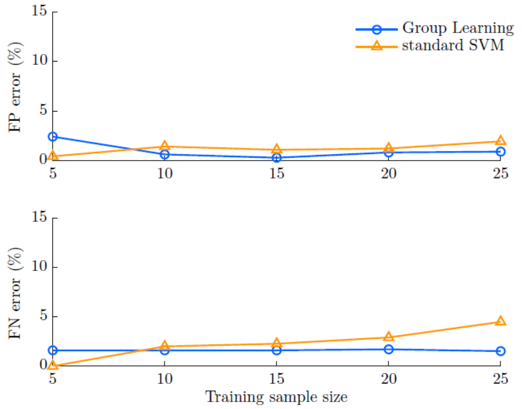

**Fig. 6.** The training FP and FN errors as a function of training sample size (per class). The Group Learning method uses 4-patch (t = 4) setting and the patch size is 14×14 pixels.

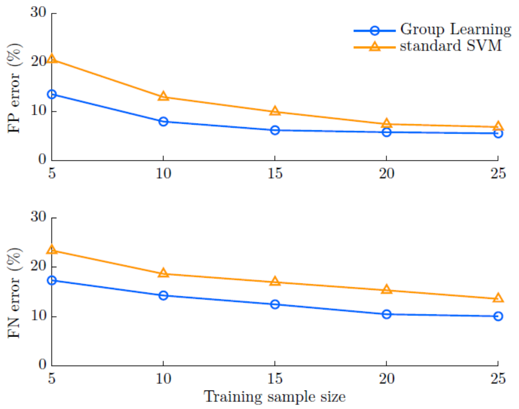

**Fig. 7.** The test FP and FN errors as a function of training sample size (per class). The Group Learning method uses 4-patch (t = 4) setting and the patch size is 14×14 pixels.

Fig. 6 shows empirical comparison for training error rates between standard SVM and the Group Learning method, as a function of training sample size (per class). Both methods achieve relatively low training error rate.

The test errors for both methods are shown in Fig. 7. The Group Learning method achieves much lower test errors compared with standard SVM, especially for small training sample size. These results suggest the effectiveness and capability of the Group Learning method (vs. standard SVM).

### *B. Data sets with heavily-overlapping class distributions*

Next, we describe performance of the Group Learning method for heavily overlapping data sets, including (a) ‘matrix-of-digits’ data introduced in Section III-B and (b) canine intracranial electroencephalography (iEEG) data set used for prediction of epileptic seizures [11]. This paper follows the same experimental setting as in [11] but applies different post-processing rules. The input dimensionality for both data sets is very high. Each ‘matrix-of-digits’ data sample has 156,800 input features, and each iEEG data sample has 69,120 input features as defined in [11]. Due to high dimensionality, along with heavily overlapping class distributions, modeling such data sets is very challenging for standard machine learning methods. Hence, we apply the Group Learning approach and compare three post-processing procedures, namely the *majority voting*, *adaptive post-processing*, and the *high-sensitivity post-processing*, as described in Section III-B.

#### Matrix of digits data: balanced training set

First, consider performance of Group Learning for balanced training data, under the following experimental setting:

(1) *positive class:* 800-digit matrix composed of 720 even digits and 80 digits ‘1’ (see Fig. 4);

(2) *negative class:* 800-digit matrix composed of 800 even digits (see Fig. 5);

(3) number of training inputs/matrices: 80(40 per class);

(4) number of validation matrices: 80 (40 per class);

(5) number of test matrices: 1000 (500 per class)

Note that dimensionality of each matrix (input sample) is very high: $d$ = 156,800 (i.e., 800 digits * 14×14 pixels).

Under Group Learning, each sample (~matrix-of-digits) consists of $t$ = 800 sub-images, so that SVM classifier training is performed in $d/t$ -dimensional space. However, predicting class label (for test input matrix) requires combining all $t$ individual predictions made by trained SVM classifier. All these $t$ = 800 predictions (for sub-images) can be represented via histogram-of-projections technique [1], [4]. Such histograms of projections for both training and test inputs (matrices) are shown in Fig. 8. The histograms (for different classes) are heavily overlapping and shown in color. This overlapping implies that simple post-processing rules (to combine $t$ = 800 predictions) will not work. As shown in Table I, using *simple majority voting* would predict every test input as *negative class* yielding 100% FN and 0% FP error rate. On the other hand, the *high-sensitivity* rule (as described in Section III-B) will predict every test input as positive class yielding 0% FN and 100% FP error rate.

Applying *adaptive* post-processing procedure (as described in Section III-B) results in accurate classification of these two heavily-overlapping class distributions (e.g., 0% FN and 5% FP test error rate, as shown in Table 1). Fig. 9 shows clear separation between the $\mu$ values for samples from positive and negative classes, for both training and test data. As evident from Fig. 9, using decision threshold, $Q(\boldsymbol{\mu}, 1.00) = Q^*$, one can separate all positive and negative samples in the training set, and make only a few errors for test data.

TABLE I. TEST ERRORS (FN AND FP) OF GROUP LEARNING USING 3 POST-PROCESSING METHODS (FOR BALANCED TRAINING DATA)

| **Post-processing** | **FN (%)** | **FP (%)** |
|---|---|---|
| **Majority voting** | 100 | 0 |
| **Adaptive** | 0 | 5 |
| **High-sensitivity** | 0 | 100 |

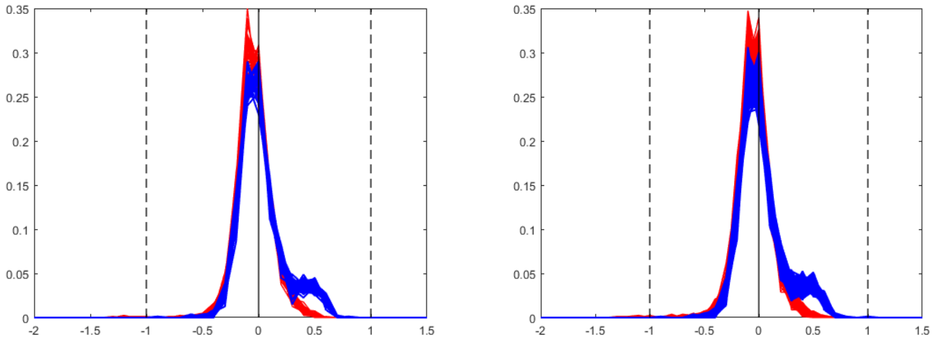

**Fig. 8.** Histograms of projections for 'matrix-of-digits' data formed by 800 handwritten digits. *Notation:* red ~ negative class, blue ~ positive class samples; SVM margin borders are labeled -1 and +1. (a) histograms for training data (left, 40 positive and 40 negative matrices); (b) histograms for test data (right, 500 positive and 500 negative matrices).

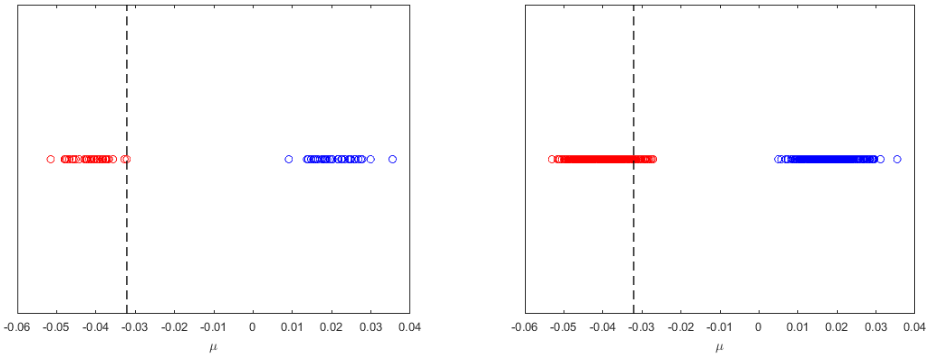

**Fig. 9.** Clustering of the mean values of 800 outputs of SVM classifier under Group Learning setting. Dashed line indicates optimal adaptive decision threshold, $Q^*$ which separates 100% of negative training samples ($in\ the\ \mu$ –space). (a) histogram of $\mu$ –values for training data (left, 40 positive and 40 negative samples), (b) histogram of $\mu$ –values for test data (right, 500 positive and 500 negative samples).

## Matrix of digits data: unbalanced training set

Next, consider performance of Group Learning for unbalanced training data. The experimental setting is the same as before, except for unbalanced training set, as detailed below:

(1) positive class: 800-digit matrix composed of 720 even digits and 80 digits '1' (see Fig. 4);

(2) negative class: 800-digit matrix composed of 800 even digits (see Fig. 5);

(3) number of training samples (matrices): 5 examples from positive class and 40 from negative class;

(4) number of validation matrices: 80 (40 per class);

(5) number of test matrices: 1000 (500 per class)

The histograms of projections for training and test data are shown in Fig. 10 – indicating heavily overlapping class distributions. Due to heavily unbalanced training data, most SVM predictions for both classes will be negative, even for positive test inputs (see Fig. 10). Hence, simple majority voting is expected to fail for this data set. Table II shows prediction performance for Group Learning under 3 post-processing rules. As shown in Table II, using *simple majority voting* would predict every test input as *negative class* yielding 100% FN and 0% FP error rate. The *high-sensitivity rule* correctly classifies 48% of positive samples and all negative samples. Adaptive post-processing method yields robust prediction (0% FN, 0.2% FP). Notably, its adaptive threshold was estimated using *only* training data – its selection is illustrated in Fig. 11.

TABLE II. TEST ERRORS (FN AND FP) OF GROUP LEARNING USING 3 POST-PROCESSING METHODS (FOR UNBALANCED TRAINING DATA)

| **Post-processing** | **FN (%)** | **FP (%)** |
|---|---|---|
| **Majority voting** | 100 | 0 |
| **Adaptive** | 0 | 0.2 |
| **High-sensitivity** | 52 | 0 |

## Canine iEEG data with unbalanced training data set

Seizure prediction from iEEG signal is commonly formalized as binary classification of preictal and interictal segments of iEEG signal [11]. This iEEG data is naturally unbalanced since seizures are rare events. In this experiment, we use the same data and experimental setting as in [11]:

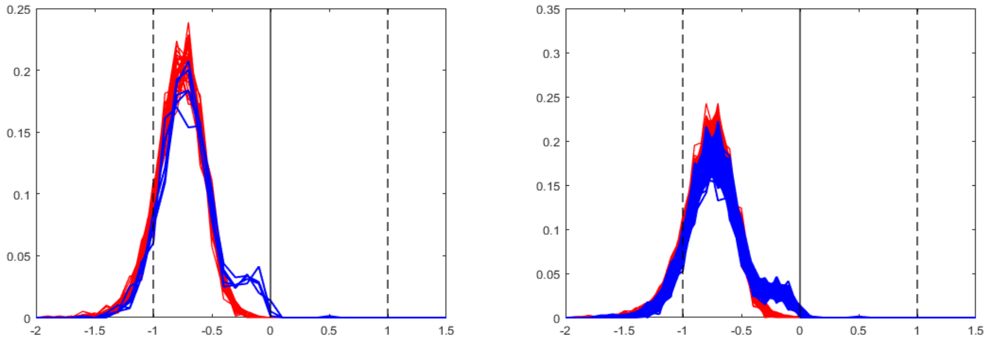

**Fig. 10.** Histograms of projections for 'matrix-of-digits' data set formed by 800 handwritten digits. *Notation:* red ~ negative class, blue ~ positive class samples; SVM margin borders are labeled -1 and +1. (a) histograms for training data (left, 5 positive and 40 negative matrices); (b) histograms for test data (right, 500 positive and 500 negative matrices).

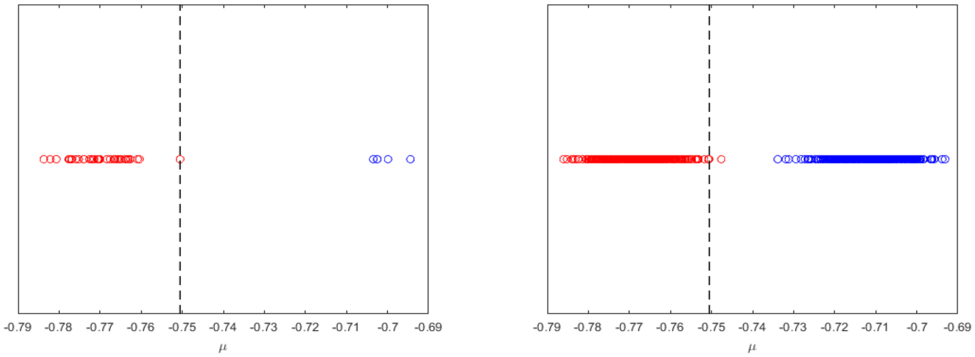

**Fig. 11.** Clustering of the mean values of 800 outputs of SVM classifier under Group Learning setting. Dashed lines indicate the adaptive decision threshold (estimated from training data) in the decision space ($\mu$ –space). (a) histogram for training data (left, 5 positive and 40 negative samples), (b) histogram for test data (right, 500 positive and 500 negative samples).

(1) *Preictal segments* (~positive class): correspond to 4-hr consecutive iEEG segments from the period of 0.5-4.5 hrs before *lead seizures*.

(2) *Interictal segments* (~negative): correspond to any other available 4-hr consecutive iEEG segments which are sufficiently far away from seizures.

(3) *Problem formalization:* standard binary classification of 4-hr consecutive iEEG segments (preictal vs. interictal) under unbalanced setting.

Following the same Group Learning formalization as shown in Figure 3, each 4-hr iEEG segment is considered as many (720) consecutive 20s windows. Each 20s window is represented by 6 features corresponding to energy in each of six standard Berger frequency bands (0.1–4 Hz, 4–8 Hz, 8–12 Hz, 12–30 Hz, 30–80 Hz, and 80–180 Hz). Since iEEG data contains 16 channels, each 20s window is represented by 6*12=96 features. So under Group Learning each 4-hr iEEG segment is represented as 720 groups, in lower-dimensional space $x'_j \in R^{96}$, $j = 1, . . . , 720$.

Following [11], we include four dogs with 5-18 preictal segments and 8 times more interictal segments selected from random regions of year-long recordings (see details in Table III). We apply the same data-analytic modeling as in [11], using dog-L7 dataset as an example. This dataset has seven lead seizures, i.e., 7 preictal segments, and 56 interictal segments.

According to the modeling setting used in [11], we have 7 experiments with 7 different models and each model is tested using its own hold-out test set (containing 1 preictal and 1 interictal). The performance indices (FN, FP) are estimated based on these seven test sets (total 7 preictal and 7 interictal).

For linear SVM, the complexity parameter $C$ is estimated via $M$-fold cross-validation on the training set ($M$ is the number of preictal segments), so that balanced validation set (~one interictal and one preictal segment) is always used.

TABLE III. SEIZURE DATA FOR FOUR CANINES

| **Dog** | **# preictal segments** | **# interictal segments** |
|---|---|---|
| **L2** | 6 | 48 |
| **L7** | 7 | 56 |
| **M3** | 18 | 144 |
| **P2** | 5 | 40 |

Modeling this unbalanced data set yields highly overlapping class distributions, for both training and test data. Additionally, due to non-stationarity of iEEG data, the classifier outputs exhibit high variability (see typical modeling results for dog-L7 in Fig. 12). Hence, standard learning methods work poorly for such data, and this motivates the Group Learning approach [11]. For data set in Table III, Group Learning using *majority*

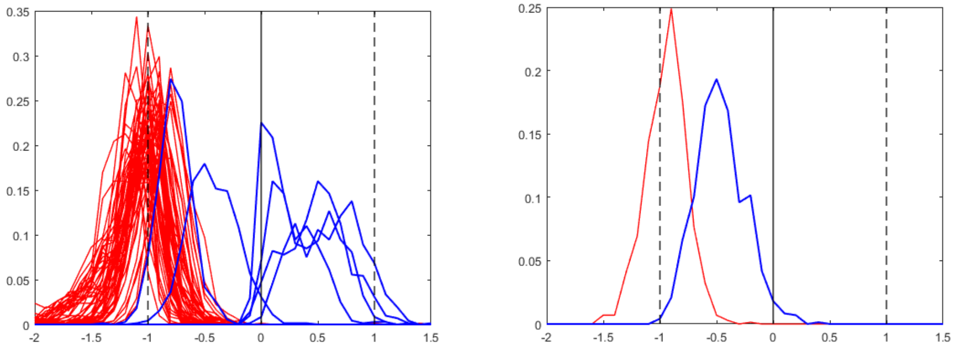

**Fig. 12.** Experimental results for modeling dog L7 data using Group Learning. Histograms of projections for training data (left) and test data (right). *Notation:* the solid line indicates SVM decision boundary and the two dashed lines indicate the margin borders (for linear SVM).

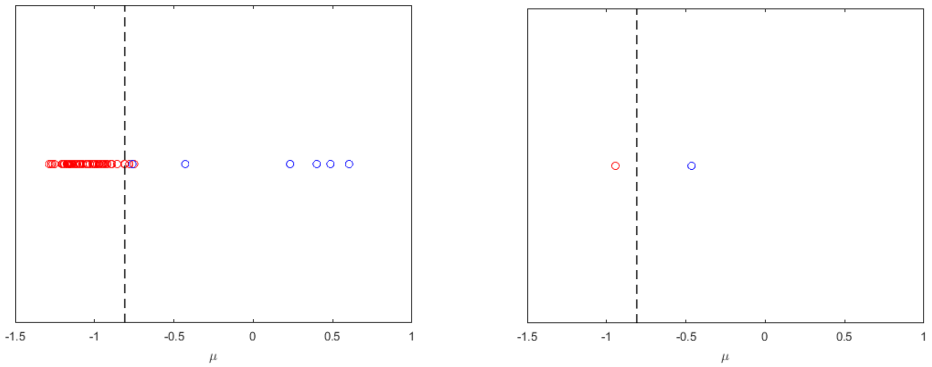

**Fig. 13.** Modeling dog L7 data using Group Learning and adaptive post-processing. Clustering of the mean values of 720 outputs of SVM classifier. Dashed lines indicate the adaptive decision threshold, estimated from training data, in the decision space ($\mu$ –space). (a) distribution of $\mu$ -values for training data (left), (b) separability of $\mu$ - values for two test inputs (right).

*voting* post-processing shows good FP error rate (0%) but large FN rate (17%-60%). On the other hand, using *high-sensitivity rule* yields good FN rate (0% - 17%), but very poor FP error rate (0%-71%). However, proposed *adaptive method* achieves both good FN rate (0%-20%) and excellent FP rate (0%-11%) – see results in Table IV. Selection of adaptive threshold for this data set is complicated by non-stationarity of iEEG signal, as illustrated next using dog-L7 data. This non-stationarity negatively affects separability of the two classes in the decision space (or $\mu$ -space). Fig. 13 shows experiment 3 for dog-L7 data, illustrating non-separability of positive and negative class distributions for training data. Hence, adaptive decision threshold, $Q(\boldsymbol{\mu}, 0.95)$, is chosen as the 95% quantile, i.e. the maximum quantile level that yields correct classification of all positive (~majority class) training samples.

TABLE IV. TEST ERROR RATES (FN AND FP) FOR THREE POST-PROCESSING METHODS

| **Dog** | **Majority voting** | | **Adaptive** | | **High-sensitivity** | |
|---|---|---|---|---|---|---|
| | **FN(%)** | **FP (%)** | **FN(%)** | **FP(%)** | **FN(%)** | **FP(%)** |
| **L2** | 17 | 0 | 0 | 0 | 17 | 0 |
| **L7** | 29 | 0 | 14 | 0 | 14 | 71 |
| **M3** | 33 | 0 | 11 | 11 | 11 | 44 |
| **P2** | 60 | 0 | 20 | 0 | 0 | 60 |

## V. CONNECTIONS TO DEEP LEARNING: SIMILARITY AND DIFFERENCES

The Group Learning method performs mapping of the original high-dimensional data onto lower-dimensional space in a manner similar to processing in Convolutional Neural Networks (CNN) also known as Deep Learning networks [12], [13]. That is, our representation of the original feature vector as a collection of (disjoint) subsets of features is similar to the convolutional layer, and our post-processing rules are similar to the pooling layer in CNNs. However, our method incorporates these two layers only once, vs. using these two layers multiple times in CNNs.

Several additional differences between Group Learning and Deep Learning are discussed next:

*Amount of available data:* Deep Learning (DL) networks usually demonstrate competitive performance only for *very large data sets*, whereas the Group Learning approach is applicable to small/ very sparse data sets.

*Theoretical understanding:* There is little theoretical understanding of DL networks. In contrast, we have shown the connection between Group Learning and VC-theory, which helps to explain improved generalization performance.

*Post-processing rules:* Under Group Learning approach, post-processing rules are applied *only* during prediction (test) stage, whereas in DL networks pooling layers are used during *both* training and testing stages. Also, pooling operation in DL is introduced as a number of heuristic rules. In contrast, post-processing rules under Group Learning are clearly related to statistical characteristics of real-world data.

*Uniqueness of estimated models:* DL networks represent a set of ad hoc algorithmic recipes, so that selection of multiple design parameters and thresholds is left to the discretion of human users. Hence, for a given data set, there may be multiple predictive models estimated by DL (and typically only the best models are published). In contrast, the Group Learning methodology always results in a single model (since it has very few tuning parameters and there is better understanding of how these parameters affect generalization).

## VI. Summary

This paper introduced a new learning paradigm: Group Learning and adaptive post-processing for high-dimensional data. Empirical results show improvement in prediction accuracy for two well-known data sets: MNIST digits and iEEG data for seizure prediction.

The Group Learning method initially transforms each *d*-dimensional training/test sample into *t* training/test samples of lower dimensionality *d/t*. This step effectively performs dimensionality reduction and increases the number of training samples. Then the Group Learning method estimates a single classifier using all training samples (in a lower dimensional space). During testing stage, *t* predictions are made for *t* test inputs (considered as a group), and these predictions are combined via post-processing rules to obtain final prediction.

We discuss two different post-processing rules for Group Learning: standard majority voting and new adaptive post-processing (utilizing global statistical characteristics of application data), and relate these post-processing rules to statistical characteristics of (unknown) class distributions. In particular, standard majority voting rule is appropriate for simple data sets (with non-overlapping class distributions), such as digit recognition data. On the other hand, for many 'difficult' applications, such as seizure prediction, fraud detection etc., class distributions may be overlapping and non-stationary. For such data sets, we propose new adaptive post-processing (for Group Learning), where combining *t* predictions (for given test input) is based on global statistical indices for positive and negative class distributions. In particular, we propose such a post-processing scheme based on the distribution of the mean values of SVM predictions (for *t* test inputs). This post-processing strategy is new for machine learning methods. However, it may be conceptually related to the idea of Learning Using Statistical Invariants (LUSI) recently introduced by Vapnik [14]. That is, proposed adaptive post-processing is based on assumption that adaptive threshold (for the mean of *t* classifier outputs) providing good separation between two classes for training data (with known labels) is also likely to yield good separation of the means for test inputs (when labels are unknown). In other words, adaptive threshold (estimated from training data) is used as 'statistical invariant'. Further work may be needed to investigate possible connections between the Group Learning method and LUSI framework. Empirical results in this paper illustrate the effectiveness of Group Learning, for both balanced and unbalanced data sets.

## Acknowledgment

This work was supported, in part, by NIH grant UH2NS095495, and NIH grant R01NS092882.

## References

[1] V. Cherkassky and F. Mulier, *Learning From Data: Concepts, Theory and Methods*, 2nd ed. Hoboken, New Jersey: John Wiley & Sons, 2007.

[2] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. New York, NY, USA: Springer series in statistics, 2001.

[3] B. Schölkopf and A. Smola, *Learning with Kernels: Support Vector Machines, regularization, optimization, and beyond.* MIT press, 2001.

[4] V. Cherkassky, *Predictive Learning*. VCtextbook.com, 2013.

[5] I. Guyon and A. Elisseef, "An introduction to variable and feauture selection," *J. Mach. Learn. Res.*, 2003.

[6] G. John, R. Kohavi, and K. Pfleger, "Irrelevant features and the subset selection problem," in *Machine Learning Proceedings 1994*, 1994, pp. 121–129.

[7] Q. Gu, Z. Li, and J. Han, "Generalized fisher score for feature selection," *arXiv Prepr. arXiv1202.3725*, 2012.

[8] F. Fleuret, "A weighted least-squares approach to clusterwise regression," *AStA Adv. Stat. Anal.*, vol. 95, no. 2, pp. 205–217, 2011.

[9] V. Vapnik, *The nature of statistical learning theory*. New York, NY, USA: Springer-Verlag New York, Inc., 1995.

[10] V. Vapnik, *Statistical learning theory*. New York: Wiley, 1998.

[11] H. T. Shiao, V. Cherkassky, J. Lee, B. Veber, E. E. Patterson, B. H. Brinkmann, and G. A. Worrell, "SVM-based system for prediction of epileptic seizures from iEEG signal," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 5, pp. 1011–1022, 2017.

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[13] A. Krizhevsky, I. Sutskever, and H. Geoffrey, "Imagenet classification with deep convolutional neural networks," *Adv. Neural Inf. Process. Syst.*, 2012.

[14] V. Vapnik and R. Izmailov, "Rethinking statistical learning theory: learning using statistical invariants," *Mach. Learn.*, vol. 108, no. 3, pp. 381–423, 2018.