Mixed Effects Models for Single-Trial ERP Detection in Noninvasive Brain Computer Interface Design

Yonghong Huang [a], Deniz Erdogmus [b,a], Kenneth Hild II [a], Misha Pavel [a], Santosh Mathan [c]

[a] *Augmented Cognition Lab, Division of Biomedical Engineering, Oregon Health and Science University, Portland, OR, USA*
[b] *Cognitive Systems Lab, Electrical and Computer Engineering Department, Northeastern University, Boston MA, USA*
[c] *Human Centered Systems Lab, Honeywell Research Laboratories, Seattle WA, USA*

Abstract

Single-trial evoked response potential detection is a fundamental problem that needs to be solved with high accuracy before noninvasive brain computer interfaces (BCI) can become a widely used practical tool that enables seamless communication with and control of a computer and any peripheral devices that can be connected to it. While in current BCI prototypes multi-trial inference is utilized with some success to convey user's intent to the computer for various applications, speed of such communication is inherently limited by the number of stimulus repetitions the subject has to go through before one command selection can be transmitted to the computer. Consequently, number of stimulus repetitions (i.e., number of trials) is inversely proportional to the speed of communication and control that the subject can achieve.

In this chapter, we provide a review of our recent work on using mixed effects models, a parametric modeling approach to statistically model trial-responses in electroencephalography in a generative fashion. Emerging from this generative model, we also develop a Fisher kernel that is in turn utilized in the support vec-

tor machine framework to develop a discriminative model for single-trial evoked response potential detection. Our results demonstrate that across multiple subjects and multiple sessions, the Fisher kernel detector outperforms its likelihood ratio test counterpart based on the generative model as well as other benchmark classifiers, specifically support vector machines with linear and Gaussian kernels.
Key words: Brain computer interface, single-trial ERP detection in EEG, mixed effects model, fisher kernel support vector machine

---

1. Introduction

Single-trial evoked response potential (STERP) detection is a fundamental signal processing problem that needs to be solved effectively and efficiently for noninvasive brain computer interfaces (BCI) operating synchronously with some stimulus sequence to become a practical solution to assistive and augmentative communication (AAC) needs of persons with disabilities. While in assistive communication device applications the role of the BCI is to act as a keyboard-substitute, there exist many other applications of noninvasive BCIs that could utilize similar visual presentation paradigms where the displayed letters are replaced by images of interest to achieve goals related to particular tasks involved; for instance image retrieval from large databases for medical and civilian research and planning purposes, as well as for image mining in the web for recreation are potential applications that will create a broader impact on the society.

Recent research that utilize the visual evoked potentials (VEP) in various visual presentation paradigms have primarily relied on multi-trial ERP detection to achieve practically acceptable accuracy levels. For instance, the well known P300 Speller uses 8-16 repetitions typically [4], while G.tec claims accurate-enough de-

tection for brain-controlled typing using only 2-repetitions in well-trained subjects [2]. It is clear that, under the assumption of statistically independent electroencephalography (EEG) measurements in response to different presentations of the same stimulus, an exponentially diminishing ERP-detection error probability can be attained by simply multiplying the class likelihoods obtained from a STERP detector as in independent Bernoulli trials.

In statistical inference theory, the problem of signal detection in noise, when framed as a hypothesis test, one usually constructs a generative probabilistic model of the measured data, which then forms the basis for Bayesian inference - optimal in terms of expected risk minimization. Deciding the class label (in our case, ERP(or 1) or noERP(or 0)) of a novel EEG waveform obtained in response to a particular visual stimulus could be achieved using the likelihood ratio test procedure for instance: specifically, the ratio of likelihoods of the new data under the two competing generative models compared with a threshold reveals the optimal decision in this minimum expected risk framework (note that one might be interested in other statistics of the risk as well, in which case, different optimal decision rules would have been obtained) [1]. One particular strength of generative probabilistic models is that it provides an understanding of the underlying mechanisms for the process and therefore they contain more information that one can extract in addition to the optimal decision for the signal detection problem. On the negative side, this could also be interpreted as a weakness of the generative approach - the model attempts to capture more information than needed for the purpose, thus generalization might fail for complex classification boundaries with small amount of training data.

Discriminative learning in machine learning is an approach that focuses on

learning only the decision boundary in a classification problem, in an attempt to avoid the shortcomings of a restrictive parametric generative model that tries to fit to the data throughout the whole space. Linear discriminants, multilayer perceptrons, support vector machines (SVM) that are trained to estimate a scalar discriminant index (which essentially corresponds to identifying a linear or nonlinear data projection function after which simple thresholding can reveal optimal decisions under the assumed model and criterion) are among the examples of this approach [3]. Note that the Bayes minimum risk detector using true underlying class distributions is also effectively a nonlinear dimensionality reduction that allows for threshold comparison; while generative models attempt to approximate the individual class distributions, discriminative models will attempt to model the surface in the feature space which will be mapped to the threshold value. Due to the reduced complexity of the function to be modeled, these latter models generally outperform the former approach in real world problems where true distributions of underlying generative mechanisms are difficult to formulate (usually due to lack of understanding at the fundamental level or due to prohibitive computational complexity of such forward models in dynamic systems - both are influential in the case with brain signals).

Fisher kernels are proposed for data with variable lengths where generative models can be formulated but discriminative learning is desired due to the inherent penalization of longer data [6], citeJaa00, [5]. This technique provides a link between generative models and discriminative learning methods (although the technique refers to the use of a kernel, in fact, it is mathematically an alternative model-based distance metric, thus can be utilized in various discriminative methods if employed properly). Specifically, the intuition is that distances be-

tween pairs of data points should be measured using geodesics on the manifold induced by the generative probability density model; this approach is consistent with the information geometry of statistical models and mathematically superior to techniques that utilize Euclidean distances or other linear algebraic variations. The Fisher information matrix is known to form a natural Riemannian metric for a given parametric probability density model in its parameter space [8]. The Fisher kernel exploits this property of the Fisher information matrix and employs the Fisher score to construct an inner product that measures distances between datum pairs informed by the underlying generative model.

In this book chapter we will review our recent work in the area of STERP detection for the purpose of stimulus-synchronous BCI design. Specifically, we will utilize the mixed effects model (MEM) approach (a graphical hierarchical Bayesian special case) to develop a generative model for multichannel EEG signals and we will evaluate the performance of likelihood ratio test and Fisher kernel SVM detectors in comparison to linear and Gaussian-kernel SVM detectors. In this context, we will explore simple linear spatial dimensionality reduction techniques as well as techniques for incrementally updating SVMs for long-term adaptivity of the BCI as additional training data becomes available.

2. MEM for Stimulus-Synchronized EEG

Mixed effects models [9] are utilized for longitudinal sequence data in biostatistics, where each subject/sample yields a sequence of measurements and these measurement sequences are assumed to follow a temporal structure that consists of three components: (i) population contribution, which corresponds to the population average; (ii) individual variability component, which determines the ran-

dom variation of an individual from the population mean; (ii) stationary measurement noise. The population and individual components are assumed to be linear combinations of basis functions of time and the measurement noise is usually assumed to be temporally white.

In BCI applications, for each visual stimulus, the corresponding VEP waveforms do not necessarily have exactly the same shape; specifically, ERP components such as P300 may have variations in amplitude, duration, or latency from trial to trial. These variations might arise from a variety of factors including fatigue and attention, task difficulty and stimulus complexity (our unpublished results indicate that as stimulus presentation rate and stimulus image complexity increases, the latency of the P300 increases –suggesting more processing is performed by the brain– and the peak amplitude and duration of this component decreases - suggesting that the decision reached has higher uncertainty).

Due to these observations, we will employ the MEM framework to develop two generative models for EEG waveforms in response to *target* and *distractor* visual stimuli. It is assumed in BCI design that target stimuli results in ERP generation and distractor images are largely ignored by the brain.

2.1. Model Description

In the following, we will refer to the measured (vectorized) multichannel EEG response for a particular visual stimulus as an *individual* and the group of individuals that come from the same type of stimulus (i.e., target or distractor) as a *population*. For individual i of N from population c (where $c \in \{0, 1\}$ denotes class membership: ERP or noERP), the MEM is written as:

$$\mathbf{y}_i^c = \mathbf{X}_i^c \boldsymbol{\alpha}^c + \mathbf{Z}_i^c \mathbf{b}_i^c + \boldsymbol{\varepsilon}_i^c. \tag{1}$$

In the MEM expression:

- $\mathbf{y}_i^c$ is an $n^c \times 1$ vector of observations for the i^{th} individual and n_i is the number of observations for the i^{th} individual.
- $\boldsymbol{\alpha}^c$ is the $p^c \times 1$ population effect coefficient vector.
- $\mathbf{X}_i^c$ is an $n^c \times p^c$ population design matrix (basis vectors for fixed effects).
- $\mathbf{b}_i^c$ is a $k^c \times 1$ individual random effect vector. These vectors are assumed to have a hyper-distribution (e.g., zero-mean multivariate Gaussian with covariance $\mathbf{D}^c$: $\mathbf{b}_i^c \sim N(\mathbf{0}, \mathbf{D}^c)$) that needs to be learned from data.
- $\mathbf{Z}_i^c$ is an $n^c \times k^c$ individual design matrix (basis vectors for random effects).
- $\boldsymbol{\varepsilon}_i^c$ is an $n^c \times 1$ vector of independent and identically distributed (iid) noise with zero mean and positive definite within-individual covariance (typically, $\boldsymbol{\varepsilon}_i^c \sim N(\mathbf{0}, \sigma^{c2}\mathbf{I})$).

Thus, assuming that all distributions involved are Gaussian, the density model corresponding to (1) can be written as $\mathbf{y}_i^c \sim N(\mathbf{X_i^c}\boldsymbol{\alpha}^c, \sigma^{c2}\mathbf{I} + \mathbf{Z}_i^c\mathbf{D}^c\mathbf{Z}_i^{cT})$. In (1), $\mathbf{y}_i^c$ is the vectorized spatiotemporal stimulus-time-locked EEG measurement (for instance from 32 channels over the duration 0-500ms following stimulus onset). In test mode, when class labels are not known, the superscript indicating class label is to be determined. In the same equation, $\mathbf{X}_i^c$ and $\mathbf{Z}_i^c$ are known design matrices (consisting of preselected basis vectors for population and individual effects in their columns). The parameters to be determined via model fitting using maximum likelihood estimation, for instance, are $\boldsymbol{\alpha}^c$, the covariance $\mathbf{D}^c$ of the random vectors $\mathbf{b}_i^c$, and the covariance of the additive background noise component $\boldsymbol{\varepsilon}_i^c$, specifically σ^{c2} if the noise is assumed to be spatiotemporally white for each class (ERP and noERP).

2.2. Model Parameter Estimation

The maximum likelihood estimates of MEM parameters can be identified using the available data with the Expectation-Maximization (EM) algorithm [10].

For a given class, let $\mathbf{V}_i^c = \sigma^{c2}\mathbf{I} + \mathbf{Z}_i^c\mathbf{D}^c\mathbf{Z}_i^{cT}$ denote $Cov(\mathbf{y}_i^c)$, the covariance of the measurement vectors from this class. If $\mathbf{V}_i^c$ was known, we could estimate $\boldsymbol{\alpha}^c$ and $\mathbf{b}_i^c$. Assuming that the measured vectors are independent (and identically distributed according to the Gaussian model prescribed by MEM, the joint data likelihood would be given by

$$p(\mathbf{y}^c;\boldsymbol{\theta}^c) = \prod_{i=1}^{N} \frac{\exp[-\frac{1}{2}(\mathbf{y}_i^c - \mathbf{X}_i^c\boldsymbol{\alpha}^c)^T\mathbf{V}_i^{c-1}(\mathbf{y}_i^c - \mathbf{X}_i^c\boldsymbol{\alpha}^c)]}{(2\pi)^{\frac{n_i}{2}}|\mathbf{V}_i^c|^{\frac{1}{2}}} \tag{2}$$

where $\boldsymbol{\theta}^c = (\boldsymbol{\alpha}^c; vec(\mathbf{D}^c); \sigma)$ for white noise. For simplicity of model, $\mathbf{D}^c$ could be assumed to be diagonal, in which case, the parameter vector would only include the individual variances of the individual random effect coefficients. From this expression, the log-likelihood as a function of the parameter vector is obtained as

$$\begin{aligned} l(\boldsymbol{\theta}^c) &= -\frac{1}{2}\{Nn^c ln(2\pi) + \sum_{i=1}^{N}[ln|\mathbf{V}_i^c| \\ &\quad +(\mathbf{y}_i^c - \mathbf{X}_i^c\boldsymbol{\alpha}^c)^T\mathbf{V}_i^{c-1}(\mathbf{y}_i^c - \mathbf{X}_i^c\boldsymbol{\alpha}^c)]\}. \end{aligned} \tag{3}$$

If the covariance parameter estimates $\hat{\sigma}^{c2}$ and $\hat{\mathbf{D}}^c$ were available, then the log-likelihood function could be maximized by the generalized least squares estimator. Specifically, taking the derivative of $l(\boldsymbol{\theta})$ with respect to $\boldsymbol{\alpha}^c$ and equating to zero, we get

$$\hat{\boldsymbol{\alpha}^c} = (\sum_{i=1}^{N}\mathbf{X}_i^{cT}\mathbf{V}_i^{c-1}\mathbf{X}_i^c)^{-1}\sum_{i=1}^{N}\mathbf{X}_i^{cT}\mathbf{V}_i^{c-1}\mathbf{y}_i^c. \tag{4}$$

Once an estimate for $\boldsymbol{\alpha}^c$ is available, we can obtain $\mathbf{b}_i^c$ using least square estimation as follows:

$$\hat{\mathbf{b}_i^c} = \mathbf{D}^c\mathbf{Z}_i^{cT}\mathbf{V}_i^{c-1}(\mathbf{y}_i^c - \mathbf{X}_i^c\hat{\boldsymbol{\alpha}^c}). \tag{5}$$

Next, we provide the EM estimates for σ^{c2} and $\mathbf{D}^c$.

M-step: If we were to observe $\mathbf{b}_i^c$ and $\boldsymbol{\varepsilon}_i^c$, we could easily obtain a simple closed-form solution using ML estimates of variances,

$$\hat{\sigma^c}^2 = \frac{1}{Nn^c}\sum_{i=1}^{N}\boldsymbol{\varepsilon}_i^{cT}\boldsymbol{\varepsilon}_i^c, \tag{6}$$

$$\hat{\mathbf{D}^c} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{b}_i^c\mathbf{b}_i^{cT}. \tag{7}$$

E-step: If σ^{c2} and $\mathbf{D}^c$ estimates are available, we could calculate the sufficient statistics as follows:

$$\begin{aligned}\sum_{i=1}^{N}\boldsymbol{\varepsilon}_i^{cT}\boldsymbol{\varepsilon}_i^c &= \sum_{i=1}^{N}\hat{\boldsymbol{\varepsilon}}_i^c(\hat{\boldsymbol{\theta}})^T\hat{\boldsymbol{\varepsilon}}_i^c(\hat{\boldsymbol{\theta}}) \\ &\quad + \sum_{i=1}^{N} tr\{Cov[\boldsymbol{\varepsilon}_i^c|\mathbf{y}_i^c, \hat{\boldsymbol{\alpha}}^c(\hat{\boldsymbol{\theta}}^c), \hat{\boldsymbol{\theta}}^c]\},\end{aligned} \tag{8}$$

$$\begin{aligned}\sum_{i=1}^{N}\mathbf{b}_i^{cT}\mathbf{b}_i^c &= \sum_{i=1}^{N}\{\hat{\mathbf{b}}_i^c(\hat{\boldsymbol{\theta}})^T\hat{\mathbf{b}}_i^c(\hat{\boldsymbol{\theta}}) \\ &\quad + Cov[\mathbf{b}_i^c|\mathbf{y}_i^c, \hat{\boldsymbol{\alpha}}^c(\hat{\boldsymbol{\theta}}^c), \hat{\boldsymbol{\theta}}^c]\},\end{aligned} \tag{9}$$

where $\hat{\boldsymbol{\varepsilon}}_i^c(\hat{\boldsymbol{\theta}}^c) = \mathbf{y}_i^c - \mathbf{X}_i^c\hat{\boldsymbol{\alpha}}_i^c(\hat{\boldsymbol{\theta}}^c) - \mathbf{Z}_i^c\hat{\mathbf{b}}_i^c(\hat{\boldsymbol{\theta}}^c)$ and $\hat{\mathbf{b}}_i^c(\hat{\boldsymbol{\theta}}^c)$ were obtained from ML estimation. Based on $\boldsymbol{\varepsilon}_i^c|\boldsymbol{\theta}^c \sim N(\mathbf{0}, \sigma^{c2}\mathbf{I})$, $\mathbf{y}_i^c|\boldsymbol{\varepsilon}_i^c; \boldsymbol{\theta}^c \sim N(\mathbf{X}_i^c\boldsymbol{\alpha}^c, \mathbf{Z}_i^c\mathbf{D}^c\mathbf{Z}_i^{cT})$, and $\mathbf{y}_i^c|\boldsymbol{\theta}^c \sim N(\mathbf{X}_i^c\boldsymbol{\alpha}^c, \sigma^{c2}\mathbf{I} + \mathbf{Z}_i^c\mathbf{D}^c\mathbf{Z}_i^{cT})$, we can derive

$$Cov[\boldsymbol{\varepsilon}_i^c|\mathbf{y}_i^c, \hat{\boldsymbol{\alpha}}^c(\hat{\boldsymbol{\theta}}^c), \hat{\boldsymbol{\theta}}^c] = [(\mathbf{Z}_i^c\mathbf{D}^c{\mathbf{Z}_i}^{cT})^{-1} + (\sigma^{c2}\mathbf{I})^{-1}]^{-1}. \tag{10}$$

Similarly, based on $\mathbf{y}_i^c|\mathbf{b}_i^c; \boldsymbol{\theta}^c \sim N(\mathbf{X}_i^c\boldsymbol{\alpha}^c + \mathbf{Z}_i^c\mathbf{b}_i^c, \sigma^{c2}\mathbf{I})$, $\mathbf{y}_i^c|\boldsymbol{\theta}^c \sim N(\mathbf{X}_i^c\boldsymbol{\alpha}^c, \sigma^{c2}\mathbf{I} + \mathbf{Z}_i^c\mathbf{D}^c{\mathbf{Z}_i}^{cT})$, and $\mathbf{b}_i^c|\boldsymbol{\theta}^c \sim N(\mathbf{0}, \mathbf{D}^c)$ we can calculate

$$Cov[\mathbf{b}_i^c|\mathbf{y}_i^c, \hat{\boldsymbol{\alpha}}^c(\hat{\boldsymbol{\theta}}^c), \hat{\boldsymbol{\theta}}^c] = ({\mathbf{Z}_i}^{cT}\mathbf{Z}_i^c/\sigma^{c2} + \mathbf{D}^{c-1})^{-1}. \tag{11}$$

Thus from (6)-(11), we obtain the variance parameter estimates as:

$$\begin{aligned}\hat{\sigma}^{c2} &= Nn^c \sum_{i=1}^{N} \hat{\boldsymbol{\varepsilon}}_i^c(\hat{\boldsymbol{\theta}}^c)^T \hat{\boldsymbol{\varepsilon}}_i^c(\hat{\boldsymbol{\theta}}^c) \\ &\quad + \frac{1}{Nn^c} \sum_{i=1}^{N} tr\{[(\mathbf{Z}_i^c \mathbf{D}^c \mathbf{Z}_i^{cT})^{-1} + (\sigma^{c2}\mathbf{I})^{-1}]^{-1}\} \qquad (12)\end{aligned}$$

$$\hat{\mathbf{D}}^c = \frac{1}{N} \sum_{i=1}^{N} \{\hat{\mathbf{b}}_i^c(\hat{\boldsymbol{\theta}}^c)^T \hat{\mathbf{b}}_i^c(\hat{\boldsymbol{\theta}}^c) + (\frac{\mathbf{Z}_i^{cT}\mathbf{Z}_i^c}{\sigma^{c2}} + \mathbf{D}^{c-1})^{-1}\}. \qquad (13)$$

Upon convergence of the EM iterations, we obtain $\hat{\sigma}^{c2}$ and $\hat{\mathbf{D}}^c$.

3. Dimension Reduction in MEM Calculations

The model parameter estimation procedure provided in the previous section involves $n^c \times n^c$ matrix inversions and determinants. These computations can be reduced to $k \times k$ where $k \ll n^c$ using the following exact rank-reduction formulas. Since these reductions apply to models of both classes, we will omit the superscript indicating class label in the following expressions throughout this section.

3.1. Simplified Formulas for Log-likelihood

Since $\mathbf{V}_i = \sigma^2\mathbf{I} + \mathbf{Z}_i\mathbf{D}\mathbf{Z}_i^T$ involves an $k \times k$ low-rank matrix inversion. We can use the following dimension-reduction formulas to exploit the relevant rank-k subspace:

$$\begin{aligned}\mathbf{V}_i^{-1} &= \sigma^{-2}\mathbf{I}_n - \sigma^{-2}\mathbf{I}_n\mathbf{Z}_i(\mathbf{D}^{-1} \\ &\quad + \mathbf{Z}_i^T\sigma^{-2}\mathbf{I}_n\mathbf{Z}_i)^{-1}\mathbf{Z}_i^T\sigma^{-2}\mathbf{I}_n \\ &= \sigma^{-2}\mathbf{I}_k - \sigma^{-4}\mathbf{Z}_i^T\mathbf{Z}_i(\mathbf{D}^{-1} + \sigma^{-2}\mathbf{Z}_i^T\mathbf{Z}_i)^{-1}, \qquad (14)\end{aligned}$$

$$|\mathbf{V}_i| = \sigma^{2(n-k)}|\sigma^2\mathbf{I}_k + \mathbf{D}\mathbf{Z}_i^T\mathbf{Z}_i|. \tag{15}$$

If matrix $\mathbf{D}$ is nonsingular, we can have the log of the determinant as a function of $\mathbf{D}^{-1}$.

$$\begin{aligned}\ln|\mathbf{V}_i| &= \ln|\sigma^2\mathbf{D}^{-1} + \mathbf{Z}_i^T\mathbf{Z}_i| \\ &\quad - \ln|\mathbf{D}^{-1}| + (n-k)\ln\sigma^2. \end{aligned}\tag{16}$$

3.2. Simplified formulas for σ^2 and $\mathbf{D}$

To avoid inverse matrices in Equation (10) and (11), by using matrix inversion lemma, we have the following simplification,

$$[(\mathbf{Z}_i\mathbf{D}{\mathbf{Z}_i}^T)^{-1} + (\sigma^2\mathbf{I}_{n_i})^{-1}]^{-1} = \sigma^2\mathbf{I}_{n_i} - \sigma^4\mathbf{I}_{n_i}\mathbf{V}_i^{-1} \tag{17}$$

$$({\mathbf{Z}_i}^T\mathbf{Z}_i/\sigma^2 + \mathbf{D}^{-1})^{-1} = \mathbf{D} - \mathbf{D}\mathbf{Z}_i^T\mathbf{V}_i^{-1}\mathbf{Z}_i\mathbf{D}. \tag{18}$$

Therefore Equation (12) and (13) can be simplified as follows

$$\hat{\sigma}^2 = \frac{1}{Nn}\sum_{i=1}^{N}\hat{\boldsymbol{\varepsilon}}_i(\hat{\boldsymbol{\theta}})^T\hat{\boldsymbol{\varepsilon}}_i(\hat{\boldsymbol{\theta}}) + \sigma^2 - \frac{1}{Nn}\sigma^4\sum_{i=1}^{N}tr(\mathbf{V}_i^{-1}) \tag{19}$$

$$\hat{\mathbf{D}} = \frac{1}{N}\sum_{i=1}^{N}[\hat{\mathbf{b}}_i(\hat{\boldsymbol{\theta}})^T\hat{\mathbf{b}}_i(\hat{\boldsymbol{\theta}})] + \mathbf{D} - \frac{1}{N}\mathbf{D}(\sum_{i=1}^{N}\mathbf{Z}_i^T\mathbf{V}_i^{-1}\mathbf{Z}_i)\mathbf{D}. \tag{20}$$

Using (14), we also obtain

$$\mathbf{Z}_i^T\mathbf{V}_i^{-1}\mathbf{Z}_i = \mathbf{Z}_i^T\mathbf{Z}_i(\sigma^2\mathbf{I}_k + \mathbf{D}\mathbf{Z}_i^T\mathbf{Z}_i)^{-1}. \tag{21}$$

If $(\mathbf{Z}_i^T\mathbf{Z}_i)^{-1}$ exists, we can have

$$\mathbf{Z}_i^T\mathbf{V}_i^{-1}\mathbf{Z}_i = [\sigma^2(\mathbf{Z}_i^T\mathbf{Z}_i)^{-1} + \mathbf{D}]^{-1}. \tag{22}$$

Furthermore from (20),

$$\begin{aligned}\sum_{i=1}^{N}\mathbf{Z}_i^T\mathbf{V}_i^{-1}\mathbf{Z}_i = \sigma^{-2}\sum_{i=1}^{N}\mathbf{Z}_i^T\mathbf{Z}_i - \sigma^{-4}\sum_{i=1}^{N}[(\mathbf{Z}_i^T\mathbf{Z}_i) \\ (\mathbf{D}^{-1} + \sigma^{-2}\mathbf{Z}_i^T\mathbf{Z}_i)^{-1}(\mathbf{Z}_i^T\mathbf{Z}_i)^T].\end{aligned}\tag{23}$$

3.3. Simplified formulas for $\boldsymbol{\alpha}$ and $\mathbf{b}_i$

In (4) we can substitute

$$\sum_{i=1}^{N} \mathbf{X}_i^T \mathbf{V}_i^{-1} \mathbf{X}_i = \sigma^{-2} \sum_{i=1}^{N} \mathbf{X}_i^T \mathbf{X}_i - \sigma^{-4} \sum_{i=1}^{N} [(\mathbf{X}_i^T \mathbf{Z}_i) (\mathbf{D}^{-1} + \sigma^{-2} \mathbf{Z}_i^T \mathbf{Z}_i)^{-1} (\mathbf{X}_i^T \mathbf{Z}_i)^T], \tag{24}$$

$$\sum_{i=1}^{N} \mathbf{X}_i^T \mathbf{V}_i^{-1} \mathbf{y}_i = \sigma^{-2} \sum_{i=1}^{N} \mathbf{X}_i^T \mathbf{y}_i - \sigma^{-4} \sum_{i=1}^{N} [(\mathbf{X}_i^T \mathbf{Z}_i) (\mathbf{D}^{-1} + \sigma^{-2} \mathbf{Z}_i^T \mathbf{Z}_i)^{-1} (\mathbf{Z}_i^T \mathbf{y}_i)], \tag{25}$$

Similarly, in (5) we can substitute

$$\begin{aligned} \mathbf{Z}_i^T \mathbf{V}_i^{-1} \mathbf{y}_i &= \sigma^{-2} \mathbf{Z}_i^T \mathbf{y}_i - \sigma^{-4} (\mathbf{Z}_i^T \mathbf{Z}_i) (\mathbf{D}^{-1} \\ &+ \sigma^{-2} \mathbf{Z}_i^T \mathbf{Z}_i)^{-1} (\mathbf{Z}_i^T \mathbf{y}_i), \end{aligned} \tag{26}$$

$$\begin{aligned} \mathbf{Z}_i^T \mathbf{V}_i^{-1} \mathbf{X}_i &= \sigma^{-2} \mathbf{X}_i^T {\mathbf{Z}_i}^T - \sigma^{-4} (\mathbf{Z}_i^T \mathbf{Z}_i) (\mathbf{D}^{-1} \\ &+ \sigma^{-2} \mathbf{Z}_i^T \mathbf{Z}_i)^{-1} (\mathbf{X}_i^T \mathbf{Z}_i)^T, \end{aligned} \tag{27}$$

4. MEM Likelihood-Ratio Test ERP Detector

Given one trained MEM per class, whose likelihood values for a given $\mathbf{y}$ are denoted by $MEM^c(\mathbf{y})$, one can design an ERP detector using the standard likelihood ratio test (LRT) approach. The threshold for the decision boundary could be obtained using minimum Bayes risk criterion, if the relative risk of missing an ERP versus a false ERP detection can be assessed a priori together with the prior probability of a true ERP occurrence [1]. Alternatively, the Neyman-Pearson approach could be adopted, for instance by setting a maximum allowable false detection rate or a minimum desired positive detection rate. Since during the algorithm

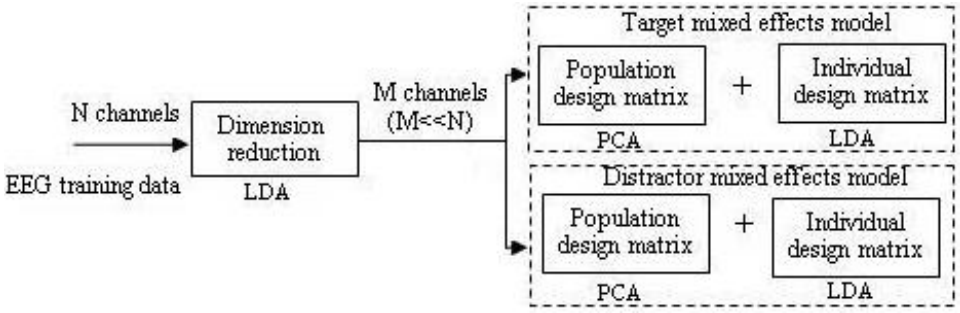

Figure 1: Training structure of the mixed effects ERP detector

assessment phase we do not wish to commit to a particular risk assumption, we utilize the area under the receiver-operator characteristic (ROC) curve (in short AUC for area under the curve) as a measure of detector performance that does not require making hard decisions.

The number of channels (spatial locations on the scalp) from which EEG is acquired might be high if denser arrays are utilized. This factor directly affects the dimensionality of the measurement vector being modeled in the MEM framework and dimensionality reduction will generally benefit the learning process from a parameter estimation variance perspective. Consequently, we employ a linear channel mixture paradigm inspired by linear discriminant analysis for preliminary dimensionality reduction followed by the training of individual class MEM parameters based on available training data for each class. Figure 1 shows the training process in block diagram format.

4.1. Channel Dimension Reduction

We employ a linear channel dimensionality reduction method inspired by the linear discriminant analysis (LDA) approach in classifier design [**?**]. Specifically, we seek to identify a set of channel linear combination coefficients that keep the

average EEG responses between ERP and noERP classes as separated as possible and also simultaneously attempt to minimize the total variance in the projected data. For each ERP sample $\mathbf{y^1}$ and noERP sample $\mathbf{y^0}$ (which are $C \times T$ matrices where C is the number of channels and T is the number of temporal samples following stimulus onset), we obtain the trial-average and trial-covariation matrices as follows:

$$\begin{aligned} \mathbf{M}^1 &= \frac{1}{N^1}\sum_{i=1}^{N^1}\mathbf{y}_i^1 \\ \mathbf{M}^0 &= \frac{1}{N^0}\sum_{i=1}^{N^0}\mathbf{y}_i^0 \end{aligned} \tag{28}$$

$$\begin{aligned} \mathbf{C}^1 &= \frac{1}{N^1}\sum_{i=1}^{N^1}(\mathbf{y}_i^1 - \mathbf{M}^1)(\mathbf{y}_i^1 - \mathbf{M}^1)^T \\ \mathbf{C}^0 &= \frac{1}{N^0}\sum_{i=1}^{N^0}(\mathbf{y}_i^0 - \mathbf{M}^0)(\mathbf{y}_i^0 - \mathbf{M}^0)^T. \end{aligned} \tag{29}$$

For a single dimensional linear channel projection of the form $\mathbf{w}^T\mathbf{y}_i^1$ and $\mathbf{w}^T\mathbf{y}_i^0$, the linear projection direction $\mathbf{w}$ is identified by maximizing the Fisher discriminant

$$J(\mathbf{w}) = \mathbf{w}^T\mathbf{S}_b\mathbf{w}/\mathbf{w}^T\mathbf{S}_w\mathbf{w}. \tag{30}$$

where the between cluster scatter matrix is $\mathbf{S}_b = (\mathbf{M}^1 - \mathbf{M}^0)(\mathbf{M}^1 - \mathbf{M}^0)^T$ and the within cluster scatter matrix is $\mathbf{S}_w = (\mathbf{C}^1 + \mathbf{C}^0)$. The solution to this is given by the generalized eigendecomposition of this symmetric matrix pair, which can also be obtained as the largest eigenvector of a nonsymmetric matrix as follows:

$$\mathbf{w} = eig(\mathbf{S}_w^{-1}\mathbf{S}_b). \tag{31}$$

For projections to higher-than-one dimension, we select the subset of largest eigenvectors with cardinality matching the desired reduced channel dimensionality. The number of eigenvectors to be retained must be determined using cross-validation or similar proper procedure from machine learning literature under the guidance of other constraints, such as computational complexity considerations.

4.2. Population and Individual Design Matrices

After some experimentation and cross validation, we have decided to develop the population and individual design matrices for both ERP and noERP MEMs using PCA and LDA, respectively [1]. This also intuitively means that the population components in the models will attempt to capture the average large power trends in the signals of each class while the individual ERP variations will be modeled trying to exploit discriminative patterns.

The population design matrices for the ERP and noERP classes are obtained as the largest few eigenvectors of the corresponding class sample covariation matrices (without subtracting the class averages as one would do in covariance calculations). Specifically, the number of eigenvectors retained for use as columns in the population design matrix is selected such that a user-defined percentage of the total variation (sum of eigenvalues, or equivalently trace of the covariation matrix). The same percentage is used as the threshold for minimum retained energy for both classes/models.

The individual design matrices are developed using LDA. Specifically the largest generalized eigenvectors of the within and between class scatter matrices are retained. Since the LDA approach uses data from both classes to select the projection directions, both models use the same individual design basis vectors. Our experiments using cross-validation showed that in most datasets, using

the largest generalized eigenvector gave optimal generalization capability, while adding more basis vectors did not improve performance significantly.

Once the population and individual design matrices are selected, the maximum likelihood MEM parameters for each class can be obtained using the EM procedure provided earlier. Fig. 1 illustrates the overall block diagram of the MEM model for each class. In our experiments, for model order selection and parameter regularization we employ 10-fold cross-validation [1] within the datasets for each subject. The optimal number of channel-LDA generalized eigenvectors (N_eigs1_LDA) in initial dimension reduction as well as in the individual design matrices, and the percentage of energy retained (Perc_eigs_PCA) in the population design matrices are selected using exhaustive search within discrete sets of values. Cross-validation performance measure utilized for these assessments is the average of the the AUC estimates within the 10-fold validation framework. [1]

4.3. MEM Operation in Testing Mode

In testing mode, for each incoming sample $\mathbf{y}_i^{Test}$ the MEM still needs to identify the best individual effect coefficient vector $\mathbf{b}_i^{c,Test}$. Specifically, for each test pattern it is assumed that under MEM^c, the following generative model is accurate:

$$\mathbf{y}_i^{Test} = \mathbf{X}_i^c \boldsymbol{\alpha}^c + \mathbf{Z}_i^c \mathbf{b}_i^{c,Test} + \boldsymbol{\varepsilon}_i^c \tag{32}$$

[1] We also employ the same approach for parameter regularization in support vector machine training when obtaining baseline performance results for comparisons. These parameters include the kernel width for the isotropic Gaussian kernel (σ^2) and the overlap penalty parameter in its training (C) [11], [12].

where $\mathbf{b}_i^{c,Test} \sim N(\mathbf{0}, \mathbf{D}^c)$ and $\boldsymbol{\varepsilon}_i^c \sim N(\mathbf{0}, \sigma^{c2}\mathbf{I})$. Since we have

$$p(\mathbf{y}_i^{Test}|\boldsymbol{\alpha}^c, \mathbf{b}_i^{c,Test}) \sim N(\mathbf{X}_i^c\boldsymbol{\alpha}^c + \mathbf{Z}_i^c\mathbf{b}_i^{c,Test}, \sigma^{c2}\mathbf{I}) \tag{33}$$

we can maximize this posterior for each class and obtain the optimal individual random effect parameter $\mathbf{b}_i^{c,Test}$ for the test pattern. This yields:

$$\mathbf{b}_i^{c,Test^*} = \mathbf{D}^c\mathbf{Z}_i^{cT}\mathbf{V}_i^{c-1}(\mathbf{y}_i^{Test} - \mathbf{X}_i^c\boldsymbol{\alpha}^c). \tag{34}$$

After we obtain $\mathbf{b}_i^{1,Test^*}$ for the ERP model and $\mathbf{b}_i^{0,Test^*}$ for the noERP model using appropriate design eigenvectors in (34), we can employ the likelihood ratio test using the respective model log-likelihood estimates:

$$\begin{aligned} l(\mathbf{b}_i^{Test^*}) &= \ln[N(\mathbf{X}_i\boldsymbol{\alpha} + \mathbf{Z}_i\mathbf{b}_i^{Test^*}, \sigma^2\mathbf{I}_{n_i})] + \ln[N(\mathbf{0}, \mathbf{D})] \\ &= -\frac{1}{2\sigma^{c2}}\|\mathbf{y}_i^{Test} - (\mathbf{X}_i^c\boldsymbol{\alpha}^c + \mathbf{Z}_i^c\mathbf{b}_i^{c,Test^*})\|_2^2 \\ &\quad -\frac{1}{2}\mathbf{b}_i^{c,Test^*,T}\mathbf{D}^{c-1}\mathbf{b}_i^{c,Test^*} + ln(C_{\sigma^{c2}}) + ln(C_{\mathbf{D}^c}) \end{aligned}$$

where $C_{\sigma^{c2}}$ and $C_{\mathbf{D}^c}$ are normalization constants for noise and prior Gaussian densities. The discriminant value of the MEM (the estimates of target likelihood) is the difference between the log-likelihood values of the ERP and noERP models.

5. Fisher Kernels for SVM

The operation of SVM (and any other nonparametric approach) relies heavily on the distance metric used in assessing how close or far two data points are. The distance is then monotonically related to an assumed similarity kernel (which represents an inner product in a corresponding high dimensional space determined by the eigenfunctions of the kernel selected). The Fisher kernel is a particular

similarity measure that is constructed using an underlying generative probabilistic model for the data. It is informed by the information geometry induced by this generative model and provides a local approximation based on the Riemannian geometry of the model. This distance metric is a natural choice for pairs of samples that are close to each other – for farther pairs, the distance is a coarse approximation, but in practice seems to provide sufficient performance.

The Fisher kernel operates in the parameter-gradient space of the generative model; specifically the gradient of the log-likelihood with respect to the model parameters. It utilizes information on how sensitive the parameters are to the parameters of the generative model. For any data vector $\mathbf{y}_i$ and model parameters $\boldsymbol{\theta}$, the Fisher score is a row vector and which is defined as

$$\mathbf{U}_{\mathbf{y}_i} = \nabla_{\boldsymbol{\theta}} \log p(\mathbf{y}_i|\boldsymbol{\theta}). \tag{35}$$

The Fisher Information matrix is defined as

$$\mathbb{I} = E_{p(\mathbf{y}_i|\boldsymbol{\theta})}\{\mathbf{U}_{\mathbf{y}_i}^T \mathbf{U}_{\mathbf{y}_i}\}, \tag{36}$$

where $E_{\mathbf{y_i}}\{\ \}$ is the expectation over $p(\mathbf{y}_i|\boldsymbol{\theta})$. The Fisher kernel is defined as

$$K_F(\mathbf{y}_i, \mathbf{y}_j) = \mathbf{U}_{\mathbf{y}_i} \mathbb{I}^{-1} \mathbf{U}_{\mathbf{y}_j}^T, \tag{37}$$

where $\mathbf{y}_i$ and $\mathbf{y}_j$ are two data samples. Detailed information and properties of the Fisher kernel can be found in Jaakkola's and Tsuda's papers [6], [13].

6. Fisher Kernel Derived From The MEM

The ERP and noERP generative models offered by the MEM paradigm can be utilized. Since in test mode the class label is not known, one option is to utilize

a mixture of MEM models to derive the Fisher kernel. Another approach is to put the emphasis on similarities as measured by the ERP model (or the noERP model) depending on under which model we would like the similarities to be accurate. The Fisher kernel will then be utilized in the SVM formalism to achieve ERP detection. The Fisher information matrix in (36) is approximated by sample averaging over the training dataset.

6.1. Fisher scores derived from the MEM

Given the parametric density model of the observation from MEM (we use the ERP model MEM^1) the Fisher score is calculated from the corresponding log-likelihood as follows:

$$\begin{aligned} \mathbf{U}_{\mathbf{y}_i} &= \nabla_{\boldsymbol{\theta}} \log p(\mathbf{y}_i|\boldsymbol{\theta}) \\ &= \left[\nabla_{\boldsymbol{\alpha}} \log p(\mathbf{y}_i|\boldsymbol{\alpha}), \nabla_{vec(\mathbf{D})}) \log p(\mathbf{y}_i|\mathbf{D}), \nabla_{\sigma} \log p(\mathbf{y}_i|\sigma^2)\right], \end{aligned} \tag{38}$$

where the model parameters of the MEM $\boldsymbol{\theta} = (\boldsymbol{\alpha}; vec(\mathbf{D}), \sigma^2)$ and data samples obey $\mathbf{y}_i \sim N(\mathbf{X}_\mathbf{i}^\mathbf{1}\boldsymbol{\alpha}^1, {\sigma^1}^2\mathbf{I} + \mathbf{Z}_i^1\mathbf{D}^1\mathbf{Z}_i^{1T})$.

6.1.1. Fisher scores of parameter $\boldsymbol{\alpha}$

Fisher scores respective to the fixed effect parameter of the MEM $\boldsymbol{\alpha}$ is a $1 \times p$ row vector

$$\frac{\partial l}{\partial \boldsymbol{\alpha}} = \left[\frac{\partial l}{\partial \boldsymbol{\alpha}_1}, ..., \frac{\partial l}{\partial \boldsymbol{\alpha}_m}, ..., \frac{\partial l}{\partial \boldsymbol{\alpha}_p}\right] \tag{39}$$

where $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1, ..., \boldsymbol{\alpha}_m, ..., \boldsymbol{\alpha}_p]^T$ is a column vector. Based on the log-likelihood expression, if we let $\mathbf{a} = \mathbf{y} - \mathbf{X}\boldsymbol{\alpha}$, where $\mathbf{y}$ is a concatenated column vector of all

training samples and $\mathbf{X}$ is a concatenated population design matrix, we have

$$\begin{aligned} \frac{\partial l}{\partial \boldsymbol{\alpha}_m} &= -\frac{1}{2}\frac{\partial(\mathbf{a}^T\mathbf{V}^{-1}\mathbf{a})}{\partial \boldsymbol{\alpha}_m} \\ &= \mathbf{a}^T\mathbf{V}^{-1}\mathbf{X}_{:m} \end{aligned} \tag{40}$$

where $\mathbf{X}_{:m}$ denotes the m^{th} column of basis vectors and $\mathbf{V}^{-1}$ is the symmetric blockwise covariance matrix consisting of all covariances in the the Gaussian distributions $p(\mathbf{y}_i|\boldsymbol{\theta})$, we have

$$\frac{\partial l}{\partial \boldsymbol{\alpha}} = (\mathbf{y} - \mathbf{X}\boldsymbol{\alpha})^T\mathbf{V}^{-1}\mathbf{X} \tag{41}$$

6.1.2. Fisher scores of parameter D

The covariance matrix $\mathbf{D}$ of the random effects of the MEM is a $k \times k$ matrix, where k is the number of basis vectors used in individual random effect modeling. The Fisher scores with respect to the entries of $\mathbf{D}$ are given by:

$$\frac{\partial l}{\partial \mathbf{D}} = -\frac{1}{2}\left[\frac{\partial \ln|\mathbf{V}|}{\partial \mathbf{D}} + \frac{\partial(\mathbf{a}^T\mathbf{V}^{-1}\mathbf{a})}{\partial \mathbf{D}}\right]. \tag{42}$$

For each entry (m, l) of $\mathbf{D}$, we have

$$\frac{\partial l}{\partial \mathbf{D}_{ml}} = -\frac{1}{2}\left[\frac{\partial \ln|\mathbf{V}|}{\partial \mathbf{D}_{ml}} + \frac{\partial(\mathbf{a}^T\mathbf{V}^{-1}\mathbf{a})}{\partial \mathbf{D}_{ml}}\right]. \tag{43}$$

The first part of (43) is

$$\begin{aligned} \frac{\partial \ln|\mathbf{V}|}{\partial \mathbf{D}_{ml}} &= \sum_{ij}\frac{\partial \ln|\mathbf{V}|}{\partial \mathbf{V}_{ij}}\cdot\frac{\partial \mathbf{V}_{ij}}{\partial \mathbf{D}_{ml}} \\ &= \sum_{ij}(\mathbf{V}^{-1})_{ij}\cdot(\mathbf{Z}\cdot\mathbf{E}_{ml}\cdot\mathbf{Z}^T)_{ij}, \end{aligned} \tag{44}$$

where $\mathbf{E}_{ij}$ is an elementary matrix with only nonzero entry of 1 occurring at location (i,j). The second part of (43) is

$$\frac{\partial(\mathbf{a}^T\mathbf{V}^{-1}\mathbf{a})}{\partial \mathbf{D}_{ml}} = -\mathbf{a}^T(\mathbf{V}^{-1}\cdot\mathbf{Z}\cdot\mathbf{E}_{ml}\cdot\mathbf{Z}^T\cdot\mathbf{V}^{-1})\mathbf{a}. \tag{45}$$

Based on (44) and (45), (43) can be written as

$$\begin{aligned}\frac{\partial l}{\partial \mathbf{D}_{ml}} &= -\frac{1}{2}[\sum_{ij}(\mathbf{V}^{-1})_{ij}\cdot(\mathbf{Z}\cdot\mathbf{E}_{ml}\cdot\mathbf{Z}^T)_{ij} \\ &\quad -\mathbf{a}^T(\mathbf{V}^{-1}\cdot\mathbf{Z}\cdot\mathbf{E}_{ml}\cdot\mathbf{Z}^T\cdot\mathbf{V}^{-1})\mathbf{a}]. \end{aligned} \tag{46}$$

Therefore (42) can be written as

$$\frac{\partial l}{\partial \mathbf{D}} = \sum_{ml}\mathbf{E}_{ml}\frac{\partial l}{\partial \mathbf{D}_{ml}} \tag{47}$$

6.1.3. Fisher scores of parameter σ^2

Under the white spatiotemporal noise assumption, the noise covariance matrix is determined by the scalar σ^2, which is the noise variance in any spatiotemporal sample value. The Fisher score for this parameter is

$$\frac{\partial l}{\partial \sigma^2} = -\frac{1}{2}\left[\frac{\partial \ln|\mathbf{V}|}{\partial \sigma^2} + \frac{\partial(\mathbf{a}^T\mathbf{V}^{-1}\mathbf{a})}{\partial \sigma^2}\right]. \tag{48}$$

The first term is explicitly given by

$$\frac{\partial \ln|\mathbf{V}|}{\partial \sigma^2} = tr(\mathbf{V}^{-1}). \tag{49}$$

The second term is

$$\frac{\partial(\mathbf{a}^T\mathbf{V}^{-1}\mathbf{a})}{\partial \sigma^2} = -\mathbf{a}^T\cdot(\mathbf{V}^{-1})^2\cdot\mathbf{a} \tag{50}$$

Therefore we can write (48) as

$$\frac{\partial l}{\partial \sigma^2} = -\frac{1}{2}[tr(\mathbf{V}^{-1}) - \mathbf{a}^T\cdot(\mathbf{V}^{-1})^2\cdot\mathbf{a}] \tag{51}$$

Concatenating all of these terms, we obtain the Fisher score with respective to the overall parameter vector as

$$\mathbf{U}_{\mathbf{y}_i} = \left[\frac{\partial l}{\partial \boldsymbol{\alpha}}, \frac{\partial l}{\partial vec(\mathbf{D})}, \frac{\partial l}{\partial \sigma^2}\right]. \tag{52}$$

6.2. Fisher Kernel from MEM

Once the Fisher scores are available, they can be used to construct the (linear) Fisher kernel using the Mahalanobis inner product with the Fisher information matrix as the scaling matrix as in (54). The exact analytical calculation of the Fisher information matrix under the expectation with respect to the MEM might be infeasible or cumbersome. Assuming that the MEM is an accurate approximation of the true underlying data distribution, we employ sample averaging over the training data to obtain an approximate expression for this matrix. Other simplifications in the literature (also suggested by Jaakkola) include simply using the identity matrix in place of the Fisher information matrix.

$$\hat{\mathbb{I}}_{tr} = \frac{1}{N_{tr}} \sum_{k=1}^{N_{tr}} \mathbf{U}_{\mathbf{y_k^{tr}}} \mathbf{U}_{\mathbf{y_k^{tr}}}^T. \tag{53}$$

The Fisher kernel between any two samples $\mathbf{x}$ and $\mathbf{y}$, where in training both of these samples are training samples and in testing one is a support vector sample and the other is a test sample, is finally given by

$$K(\mathbf{x}, \mathbf{y}) = \frac{1}{s} \mathbf{U}_{\mathbf{x}} \hat{\mathbb{I}}_{tr}^{-1} \mathbf{U}_{\mathbf{y}}^T, \tag{54}$$

where s is scaling constant. The Fisher kernels above can be used in the SVM formalism as a replacement for the commonly used Euclidean/Mahalanobis similarity measures.

7. Data Acquisition and Preprocessing

Throughout the study a large number of healthy subjects with normal or corrected vision were recruited to identify target objects of various types, qualities, and difficulty levels. In all experiments, the nominal image duration in RSVP

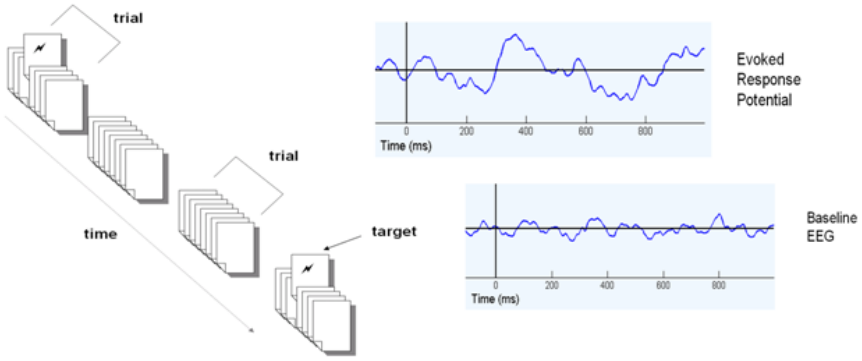

Figure 2: Illustration of RSVP paradigm (left) and typical EEG responses for target and distractor stimuli (right).Images of ERP and non-ERP signals associated with targets (left) and distractors (right). Time-zero corresponds to stimulus onset in each trial.

paradigm was 100ms/image (different subjects preferred varying rates around this speed). The RSVP paradigm consists of a sequence of blocks of images (containing multiple trials) where each block has a probability of containing a target image (in training this is set to 0.5 or 0.75) and in each block a single target image occurs (while in real testing scenarios, we did not control for this, due to the rarity of targets in most tasks, it is unlikely that multiple distinct targets occur consecutively or very close to each other in the sequence; also in some applications such as a BCI typewriter, the sequence can be controlled to achieve this property). This paradigm is illustrated in Fig. 2. For confirmation purposes, the subjects were asked to click a button as quickly as possible when they detect a target. The EEG signals used in the classification of each stimulus image were limited to the post-stimulus interval from 0ms to 500ms under the supposition that motor response corresponding to the button press would occur after 500ms (our experience shows that most subjects press the button in the interval 350-1000ms with the average of each subject falling in 475-620ms). This time limitation ensures that our ERP

detectors do not exploit the energy contained in the motor response activation process to achieve falsely high performance results - levels that one would not obtain if such motor response was not requested from or initiated by the subject.

The EEG is recorded using a 32-channel Biosemi ActiveTwo system at a sampling rate of 256Hz. To evaluate session-to-session generalization performance, we employed some subjects in multiple sessions (for instance we had four subjects attend 10 sessions distributed over 5 days - one morning and one afternoon session). Each session typically lasted about 2 hours in which 200 blocks of images containing 50 stimuli each were shown. Each block lasting only 5 seconds, the subjects were instructed to start a block of trials by a button press at will and try to avoid eye blinks and other muscle movements during the block duration. The subjects were given as much time as they wanted between blocks and they had complete control over the speed at which they complete the session.

The EEG signals were filtered using a Butterworth bandpass filter with passband set to $1 - 45$Hz. In training, target images that received a button response within 1.5sec of image onset were designated as ERP samples and those that did not receive a button response were removed. In an attempt to reduce the correlations and cross-contamination, between EEG samples, nonoverlapping samples were selected in training phase; that is, if a target image is designated as an ERP, the EEG responses for distractor images preceding and following this target image in the sequence up to 1000ms were omitted - specifically it is undesirable to have stimulus-locked responses spanning $[-100, 500]$ms response intervals that overlap with that of the selected target image (ERP sample) during training and these are omitted from training data since they might contain traces of the ERP at a shifted location. A similar nonoverlapping window constraint is applied to all

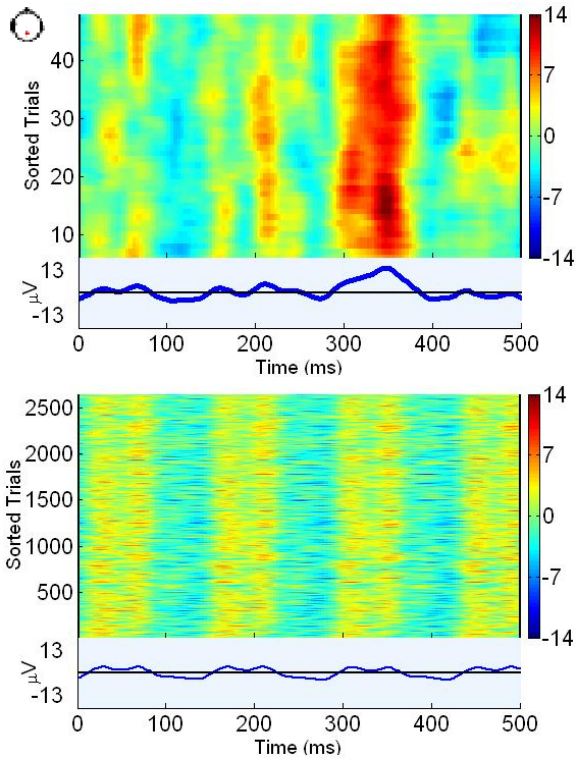

Figure 3: EEG responses recorded at Pz for target images (top) and distractor images (bottom) from stimulus onset to 500ms.

distractor images among themselves as well to avoid correlated samples. Fig. 3 shows sample EEG responses at Pz in response to a number of target and distractor trials.

8. Results

We demonstrate a selection of results from our application of the classifiers developed above to STERP detection in RSVP paradigm for BCI-based image retrieval form a database. Data from seven subjects will be used for this purpose. Dataset 1 contains 10 sessions of data from 4 subjects (acquired at OHSU).

Dataset 2 contains 8 sessions of data from 3 subjects and 5 sessions from 2 additional subjects (acquired at Honeywell). First, we demonstrate the superiority of an SVM classifier over the simpler logistic linear classifier (LLC) scheme. on these datasets. This prompts the use of SVM classifiers with generic linear or Gaussian kernels as a benchmark for performance analysis and comparison for future STERP detector designs.

8.1. Gaussian Kernel SVM as a Benchmark STERP Detector

Linear/Gaussian kernel SVMs (LKSVM and GKSVM) and LLC are trained on one session of data for each subject using 10-fold cross-validation to select optimal values for each applicable model and optimization parameter (if applicable) using exhaustive search on a discretized grid. The optimal parameter values are then employed in training these classifiers on the complete training session data and they are then tested on the remaining sessions for each subject (test set consists of 9 sessions for Dataset 1 subjects and 7 sessions for Dataset 2 subjects 1-3). The aggregated ROC curves and AUC values for each subject are shown in Fig. 4. While LLC and other linear classifiers have been used in the literature extensively in the past, our experience here demonstrates that a GKSVM outperforms an LLC significantly (both in the literally and in the statistical sense: comparison of the SVM AUC values with those of LLC via DeLong et al's method [**?**], we find that the hypothesis that GKSVM AUC is greater than that of LLC using a two-tailed t-test is accepted with $p = 4 \times 10^{-4}$).

8.2. Channel Dimensionality Reduction

An important aspect in any classifier design is dimensionality reduction, since this is a process that enables robustness and better generalization when carefully

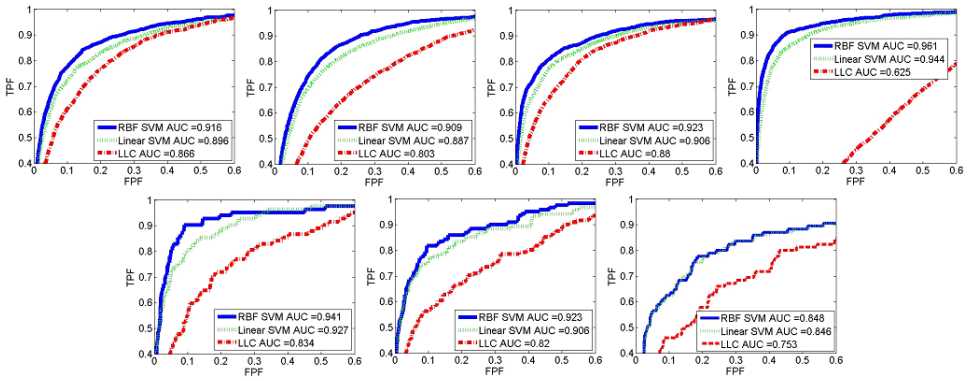

Figure 4: ROC curves and AUC values for Linear and Gaussian kernel SVMs as well as LLC for (top row) fours subjects in Dataset 1 (training with one session, testing on nine sessions each) and (bottom row) three subjects in Dataset 2 (training on one session, testing on seven sessions each).

implemented. Earlier, we described an LDA-inspired channel dimensionality reduction procedure. This procedure is implemented to all data prior to all classifier designs following this section. Specifically, using the GKSVM benchmark performance, one can easily observe that channel dimensionality reduction improves classifier generalization performance. We demonstrate this using Dataset 2 subjects 1-3 in Fig. 5. It has been our experience that optimal linear channel projections of dimensions 2-5 are typical with our data and setup.

8.3. Comparison of Classifiers

The data for each subject is projected from 32-channels to the appropriate optimal dimension using the method and in correspondance with the results presented above. For each subject, using the reduced dimension data, four classifiers are trained using one training session: MEM Likelihood Ratio Test (MEMLRT), Linear Kernel SVM (LKSVM), Gaussian Kernel SVM (GKSVM), and Fisher Kernel SVM (FKSVM). The ROC of each classifier as well as the correspond-

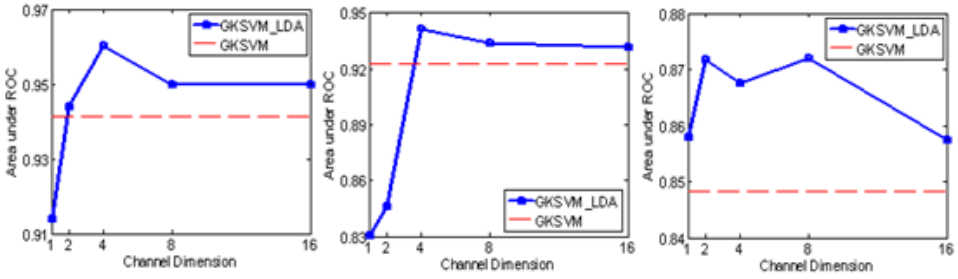

Figure 5: Channel dimension reduction using the LDA-inspired method is demonstrated here using GKSVM benchmark performance on Dataset 2 subjects where training is done using one session via 10-fold cross-validation and test performance is depicted above in the form of AUC values for various number of dimensions retained after LDA dimension reduction. The GKSVM performance with all 32-channels is provided as the red-dashed line to provide a reference that illustrates the case of no dimension reduction.

ing AUC values over the corresponding test sessions for each subject in Dataset 2 are shown in Fig. 6. The significance levels of the hypothesis comparing the mean of FKSVM to others are shown on the title of each subgraph. On average, FKSVM outperforms the other classifiers: mean MEMLRT AUC is 0.846, mean LKSVM AUC is 0.846, mean GKSVM AUC is 0.874, and mean FKSVM AUC is 0.892. These results indicate that FKSVM significantly outperforms MEMLRT and provides better performance than LKSVM and GKSVM.

9. Conclusions and Future Work

Discriminative classifiers have been more successful than their generative counterparts in many tasks since the task of the former is to model the lower dimensional classification boundary while the generative model needs to distribute its accuracy effort across the whole data space. In BCI literature, discriminative classifiers are quite popular for this reason, however, generative models of EEG

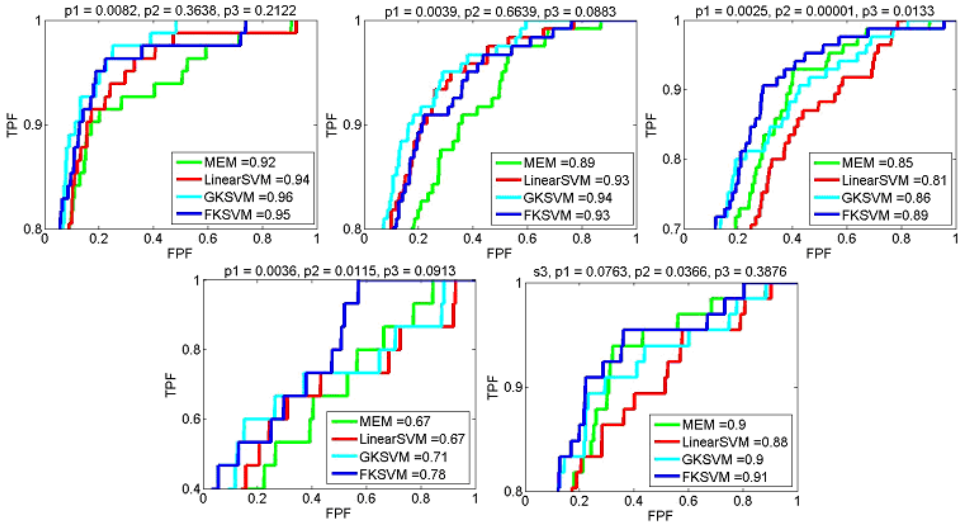

Figure 6: Comparison of ROC and AUC between four classifiers (The p-values in the titles are for AUC difference comparison t-test between FKSVM and the following, respectively: (p1) MEMLRT, (p2) LKSVM, (p3) GKSVM.

signals also offer additional opportunities for future BCI algorithm development, separation of muscle and other artifacts from components of interest while detecting relevant brain signals within one generative model framework being the most important one.

Fisher kernel formalism provides a way to incorporate information obtained from a generative model into the design of a kernel that can be utilized by an SVM classifier. In this chapter, we have developed a generative model for single trial ERP responses using a relatively simple hierarchical Bayesian model, referred to as an MEM. The likelihood ratio test based on this model, as expected did not outperform a well designed Gaussian kernel SVM - however, upon introducing Fisher kernels obtained from this model, we have obtained improvements in single trial ERP detection accuracy of SVMs, especially in subjects where the overall

performance is lower. Clearly, for subjects where the performance is on the high end, improvements are also more difficult to obtain.

This work indicates the potential of the Fisher kernel formalism in SVM design and while our underlying MEM has been relatively simple, we believe that future work in this direction where the generative model will be developed using more rigorous signal propagation models such as those used in source localization could yield Fisher kernels that dramatically outperform generic kernels such as Gaussian in BCI design.

10. Acknowledgments

Funded by DARPA (NBCHC080030). The views, opinions, and/or findings contained in this presentation are those of the author and should not be interpreted as representing the official views or policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the Department of Defense. DE is also partially funded by NSF grants IIS-0914808, IIS-0934509, and ECCS-0934506.

References

[1] R. Duda, P. Hart, D. Stork, *Pattern Classification*, Wiley, NY, 2001.

[2] C. Guger, S. Daban, E. Sellers, C. Holzner, G. Krausz, R. Carabalona, F. Gramatica, G. Edlinger, "How Many People Are Able To Control a P300-based Brain-computer Interface (BCI)?," Neuroscience Letters, vol. 462, no. 1, pp. 94-98, 2009.

[3] V.N. Vapnik, *Statistical Learning Theory*, Wiley, NY, 1998.

[4] J.R. Wolpaw, D.J. McFarland, T.M. Vaughan, G. Schalk, "The Wadsworth Center Brain-Computer Interface (BCI) Research and Development Program," IEEE Transactions on Neural Systems & Rehabilitation Engineering, vol. 11, no. 2, pp. 204-207, 2003.

[5] T. Jebara, Machine Learning Distriminative and Generative, Kluwer Academic Publishers Group, Dordrecht, Netherlands, 2004.

[6] T. Jaakkola, D. Haussler, "Exploiting Generative Models in Discriminative Classifiers," in Advances in Neural Information Processing Systems, Denver,CO, 1998, pp. 487493.

[7] T. Jaakkola, M. Diekhans, D. Haussler, "A Discriminative Framework for Detecting Remote Protein Homologies," Journal of Computational Biology, vol. 7, no. 1-2, pp. 95114, 2000.

[8] S. Amari, Differential-Geometrical Methods in Statistics, Springer-Verlag, Berlin, 1985.

[9] E. Demidenko, Mixed Models Theory and Application, Wiley, NY, 2004.

[10] A.P. Dempster, N.M. Laird, D.B. Rubin, "Maximum Likelihood with Incomplete Data via the E-M Algorithm," Journal of the Royal Statistical Society, vol. B39, pp. 138, 1977.

[11] Y. Huang, D. Erdogmus, S. Mathan, M. Pavel, "Large-scale Image Database Triage via EEG Evoked Responses," in Proc. of IEEE Int. Conf. Acoustics, Speech, and Signal Processing, Las Vegas, NV, 2008, pp. 429432.

[12] Y. Huang, D. Erdogmus, S. Mathan, M. Pavel, Mixed Effects Models for EEG Evoked Response Detection," in Proc. of IEEE Workshop on Machine Learning for Signal Processing, Cancun, Mexico, 2008.

[13] K. Tsuda, S. Akaho, M. Kawanabe, K.R. Muller, "Asymptotic Properties of the Fisher Kernel," Journal of Neural Computation, vol. 16, no. 1, pp. 115-137, 2004.