An Alternating Minimization Method to Train Neural Network Models for Brain Wave Classification

Grant Sheen *

Abstract

An alternating minimization (AM) method, which updates variables one-by-one while fixing the rest, is developed to train a neural network with low rank weights for brainwave classification. The training involves minimizing a non-smooth and non-convex cross entropy loss function. The neural network model does a projection inside a hidden layer for low dimensional feature extraction. The sub-problem for each variable is shown to be either convex or piece-wise convex with a finite number of minima. The sub-problems are solved using the bisection method with bisection intervals analytically derived. The overall iterative AM method is descending and convergent, free of step size (learning parameter) in the standard gradient descent method. Experiments consist of noninvasive multiple electrode recordings and the classification of brain wave data. The AM method significantly outperforms the standard neural network model trained by the gradient descent method in classifying four thoughts for both normal and Alzheimer subjects.

Keywords: neural networks, low rank weights, non-convex and non-smooth optimization, alternating minimization, descent and convergence, brain wave classification.

AMS subject classifications: 92B20, 65K10, 90C26.

*Sage Hill High School, 20402 Newport Coast Drive, Newport Coast, CA 92657, USA.
Email: gsheen11@gmail.com.
Mentor: Prof. Knut Solna, Department of Mathematics, UC Irvine, Irvine, CA 92697, USA.
Email: ksolna@math.uci.edu.

1 Introduction

Brain wave classification is a fascinating and challenging topic with a broad range of applications in brain-computer communication, health sciences, and biomedical engineering. In this paper, I develop and analyze a neural network (NN) model to study classification of brain waves from daily thoughts (resting, reading, eating, walking etc.) of both normal and Alzheimer subjects. Such capability can serve as mind aids for Alzheimer patients who have lost vision or speech [12]. Electroencephalography (EEG) is an electrophysiological method to record electrical activity of the brain. Noninvasive EEG involves placing electrodes along the scalp. While a subject thinks, the recorded brain signals show as wavy lines with peaks and valleys (left plot of Fig. 3). Analyzing EEG brain waves can help us understand and detect brain disorders (dementia/seizure/stroke), conduct brain imaging, analyze thoughts, control robots, and play video games using the mind (see [13, 19, 6, 10, 11, 17, 14, 21] among others). However, EEG data sets even for normal people are rather limited in the public domain, in sharp contrast to the abundant image and speech data driving the recent advances of artificial intelligence. For this reason, part of my work also concerns data collection.

A good description of classical brain signal processing and classification methods is [13]. The basic procedure consists of preprocessing, feature extraction and classification. Preprocessing removes noise and artifacts (e.g. muscle movement) in frequencies outside of the brain wave range $[1, 42]$ Hertz (Hz). Feature extraction involves Fourier or wavelet transforms so a meaningful interpretation is possible. For example in the Fourier (frequency) domain, a feature vector gives the energy distribution (so called power spectrum density) in the $[1, 42]$ Hz for a 1 second duration of brain wave. The power spectrum densities typically have different shapes for brain waves from different thoughts (right plot of Fig. 3). The representative classifiers in the last step are: statistical, large margin (support vector machines), and neural networks. The classifiers are related in various ways. The most general form of classifiers is neural networks (NN). The NNs are nonlinear classifiers with multiple hidden layer structures [22] capable of extracting hierarchical features like the mammalian brain, *provided there are sufficient data to train them.* Due to the small size data sets in EEG brain waves, a multi-layered NN (or deep neural net) can cause overfitting [21].

On the hardware side, traditional EEG recording requires a subject to wear a cap embedded with electrodes that are lubricated by conductive gel and wired to a computer for digitization. However, wireless headsets have been gaining popularity in the last decade. As a result, recording brainwaves transitioned from a pure lab environment to virtually any location for research, as wireless headsets became available for EEG recordings. The top of the line wireless headsets are Epoc and Epoc+ manufactured by Emotiv [5]. They have as many as 14 saline hydrated sensors, see [14] for an application of Epoc in a 3 class task (left, right, center) of a gaming system.

The recording of normal subjects in our study uses Epoc+ (14 channel wireless) headset in a home setting, while that of an Alzheimer subject is done by a 66 channel EEG cap in a lab. To reduce noise and variability of the features, dimensional reduction is necessary. This has been done in the past by principal component analysis (PCA), and linear discriminant

analysis (LDA), [3, 19]. My main contributions here are: 1) *accomplished dimensional reduction within an NN by training a low rank factored form of network weights*, 2) *developed a descending and convergent alternating minimization method for network training based on the piecewise convex properties of the objective function* instead of the standard stochastic gradient descent (SGD) method, 3) *achieved 4 class thought classification results, significantly out-performing those from the standard (non-factored form) NN trained by SGD.*

I would like to point out that the alternating minization (AM) method and similar approaches based on variable splitting has a long history, see the recent book [7] for an overview. Some examples are Boltzmann machine learning [2], matrix completion [9], and the alternating direction method of multipliers (ADMM) which has been applied to NN training lately [20]. The AM for Boltzmann machine learning [2] concerns matching a parametrized probability distribution to an observed distribution, however does not involve non-smooth activation function as in my NN model. ADMM involves a Lagrange multiplier to handle variable splitting while AM does not. My AM method of NN training appears to be first in that it exploits the piece-wise convex structure of the objective function and the scaling property of the non-smooth activation function of NN for minimization in each dimension via the bisection method.

The rest of the paper is organized as follows. In section 2, I introduce my proposed neural network model in low dimensional (factored) form and the quadratically regularized cross entropy (QRCE) function for training in the binary classification case. I show the alternating minimization (AM) method based on a division of variables and the convex or piecewise convex structure in each variable. The multi-class classification model and a convergence theory for AM follow with a summary of algorithm. For minimizing QRCE in each variable, a bisection method is effective, for which I derive the bisection intervals analytically. In section 3, I describe the recording, preprocessing, and numerical experiment. A comparison of AM and SGD in terms of the descent vs. oscillatory behaviors of objective values is illustrated, along with a visualization of the two and three dimensional feature vectors colored according to their class labels. The AM trained NN in factored form outperforms the SGD trained standard NN model by a 10 % margin in 4-class classification tasks for both normal and Alzheimer subjects. I conclude with remarks in section 4.

2 Neural Network and Nonconvex Optimization

Let the input data be feature vectors in $\mathbb{R}^D$, $D \gg C$, C the total number of classes. I study training of a neural network (NN) to reduce feature dimension and perform classification. Dimension reduction in NN has been shown to out-perform PCA on image data [8]. I first consider binary classification, demonstrate the piecewise convexity property of the training objective function, and present the alternating minimization (AM) method. The extensions to multi-classes will follow. For an introduction to neural networks, see section 14.7 of [15] and chapter 4 of [22].

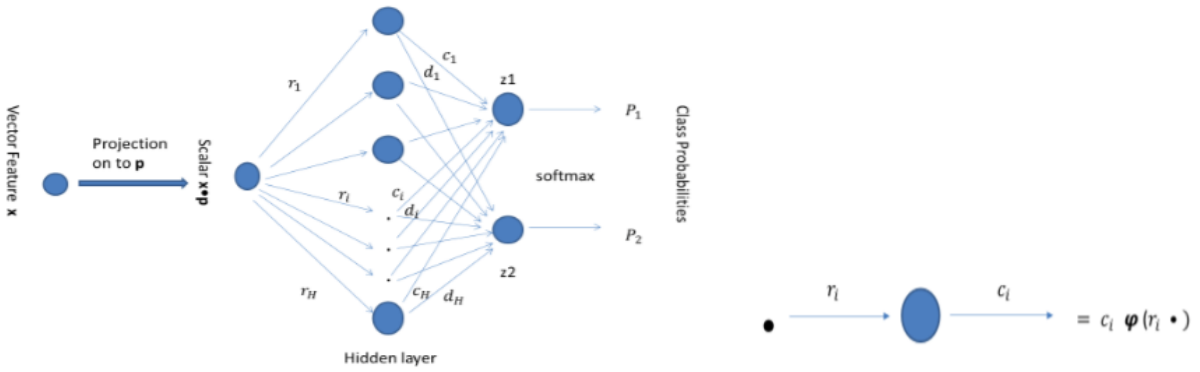


Figure 1: Left: schematic of the forward NN model (2.1)-(2.3) omitting (c_0, d_0). Right: input to output map of a neuron operation in the hidden layer and its mathematical expression.

2.1 Binary Classification Model

Let p be a "principal component" (a "neural discriminant vector") where each input data is projected, $p \in \mathbb{R}^D$. The output of a neural network with a single hidden layer for an input vector $x \in \mathbb{R}^D$ is:

$$z_1 = c_0 + \sum_{i=1}^{H} c_i \, \phi(r_i \, (x \cdot p)), \tag{2.1}$$

$$z_2 = d_0 + \sum_{i=1}^{H} d_i \, \phi(r_i \, (x \cdot p)), \tag{2.2}$$

where (c_i, d_i, r_i)'s $(i = 1, 2, \cdots, H)$ are the weights between neurons; (c_0, d_0) the bias, and ϕ is the activation function or the rectified linear unit (ReLu): $\phi = \phi(u) = \max(u, 0)$. The scaling property of ReLu: $\phi(\mu \, u) = \mu \, \phi(u)$, if $\mu > 0$, will be utilized later, see also [16] on its role in network learning. For normalized data centered at zero with unit variance, I have ignored the bias parameters inside ϕ for simplicity and efficiency. The parameter H is the number of hidden neurons. The model (2.1)-(2.2) has a projection $(x \cdot p = \vec{x} \cdot \vec{p})$ inside hidden layer. A schematic is shown in Fig. 1. The operation inside ϕ is same as input x left multiplied by a rank-1 matrix $[r_1, \cdots, r_H]' p$, hence a factored form instead of the $H \times D$ dimensional full weight matrix in a general NN with one hidden layer. The output (z_1, z_2) (excitation) is mapped into class probabilities by the softmax function:

$$P_i = \frac{\exp\{z_i\}}{\exp\{z_1\} + \exp\{z_2\}}, \quad i = 1, 2, \tag{2.3}$$

so that $P_i \geq 0$, $P_1 + P_2 = 1$. The weight and bias parameters are chosen to minimize the averaged cross-entropy (CE):

$$J_{\rm CE} := \frac{1}{M} \sum_{m=1}^{M} \left(- \sum_{i=1}^{C} P_{\rm emp}(i|o_m) \, \log P_{\rm nn}(i|o_m) \right), \tag{2.4}$$

where $P_{\rm emp}(i|o_m)$ is the empirical probability of class i observed on the training data set denoted by $\{o_m, m = 1, \cdots, M\}$, and $P_{\rm nn}$ is NN's class probability output (2.3). A common choice is a hard class label, $P_{\rm emp}(i|o) = 1$ if $i = c$, zero otherwise; where c is the class label for an observation o in the training set. Then

$$- \sum_{i=1}^{C} P_{\rm emp}(i|o_m) \, \log P_{\rm nn}(i|o_m) = - \log P_{c_m}(o_m), \tag{2.5}$$

the negative log-likelihood, c_m being the class label of o_m.

Let $x_{i,m}$ $(m = 1, \cdots, M)$ be the vector input data from class i $(i = 1, 2)$, where each class has the same number M of data points. The k-th component of vector $x_{i,m}$ is denoted by $x_{i,m,k}$. It follows from (2.4)-(2.5) that:

$$\begin{aligned}
0 \;\leq\; & 2\,M\,J_{\rm CE} = - \sum_{m=1}^{M} \log \, P_1(x_{1,m}) - \sum_{m=1}^{M} \log \, P_2(x_{2,m}) \\
= \; & - \sum_{m=1}^{M} \log \frac{\exp\{c_0 + \sum_{i=1}^{H} c_i \phi(r_i(x_{1,m} \cdot p)\}}{\exp\{c_0 + \sum_{i=1}^{H} c_i \phi(r_i(x_{1,m} \cdot p))\} + \exp\{d_0 + \sum_{i=1}^{H} d_i \phi(r_i(x_{1,m} \cdot p))\}} \\
& - \sum_{m=1}^{M} \log \frac{\exp\{d_0 + \sum_{i=1}^{H} d_i \phi(r_i(x_{2,m} \cdot p))\}}{\exp\{c_0 + \sum_{i=1}^{H} c_i \phi(r_i(x_{2,m} \cdot p))\} + \exp\{d_0 + \sum_{i=1}^{H} d_i \phi(r_i(x_{2,m} \cdot p))\}} \\
= \; & - \sum_{m=1}^{M} \left(c_0 + d_0 + \sum_{i=1}^{H} c_i \phi(r_i(x_{1,m} \cdot p)) + \sum_{i=1}^{H} d_i \phi(r_i(x_{2,m} \cdot p)) \right) \\
& + \sum_{m=1}^{M} \log \left(\exp\{c_0 + \sum_{i=1}^{H} c_i \phi(r_i(x_{1,m} \cdot p))\} + \exp\{d_0 + \sum_{i=1}^{H} d_i \phi(r_i(x_{1,m} \cdot p))\} \right) \\
& + \sum_{m=1}^{M} \log \left(\exp\{c_0 + \sum_{i=1}^{H} c_i \phi(r_i(x_{2,m} \cdot p))\} + \exp\{d_0 + \sum_{i=1}^{H} d_i \phi(r_i(x_{2,m} \cdot p))\} \right).
\end{aligned} \tag{2.6}$$

Given the data $(x_{1,m}, x_{2,m})$, the NN training minimizes a high dimensional non-convex and non-smooth objective function (2.6) over (c, d, r, p). Non-smoothness is due to ϕ being piecewise linear. The cross entropy function (2.6) grows at most linearly in these variables and is often regularized by quadratic functions to reduce overfitting [22]. The quadratically

regularized cross entropy function for the binary classification problem is:

$$
\begin{aligned}
2\,M\,J_{QRCE} &= \frac{\lambda}{2}\|(c,d,r,p)\|_2^2 - \sum_{m=1}^{M}\left(c_0+d_0+\sum_{i=1}^{H}c_i\phi(r_i(x_{1,m}\cdot p))+\sum_{i=1}^{H}d_i\phi(r_i(x_{2,m}\cdot p))\right)\\
&+ \sum_{m=1}^{M}\log\left(\exp\{c_0+\sum_{i=1}^{H}c_i\phi(r_i(x_{1,m}\cdot p))\}+\exp\{d_0+\sum_{i=1}^{H}d_i\phi(r_i(x_{1,m}\cdot p))\}\right)\\
&+ \sum_{m=1}^{M}\log\left(\exp\{c_0+\sum_{i=1}^{H}c_i\phi(r_i(x_{2,m}\cdot p))\}+\exp\{d_0+\sum_{i=1}^{H}d_i\phi(r_i(x_{2,m}\cdot p))\}\right).
\end{aligned}
\tag{2.7}
$$

Factoring out $\exp\{c_0+\sum_{i=1}^{H}c_i\phi(r_ix_{1,m}+s_i)\}$ ($\exp\{d_0+\sum_{i=1}^{H}d_i\phi(r_ix_{2,m}+s_i)\}$) from the third (fourth) term, I arrive at a compact form of (2.7):

$$
\begin{aligned}
2\,M\,J_{\mathrm{QRCE}} &= \frac{\lambda}{2}\|(c,d,r,p)\|_2^2 + \sum_{m=1}^{M}\log\left(1+\exp\{(d_0-c_0)+\sum_{i=1}^{H}(d_i-c_i)\phi(r_i(x_{1,m}\cdot p))\}\right)\\
&+ \sum_{m=1}^{M}\log\left(1+\exp\{(c_0-d_0)+\sum_{i=1}^{H}(c_i-d_i)\,\phi(r_i(x_{2,m}\cdot p))\}\right).
\end{aligned}
\tag{2.8}
$$

2.2 Alternating Minimization (AM) and Piecewise Convexity

I shall minimize (2.7) block by block: after an initialization, first minimize over (c,d)'s with (r,p)'s fixed, then minimize over (r,p)'s with (c,d)'s fixed, and iterate till convergence. When (r,p)'s are fixed, the 2nd term of (2.6) is linear in (c,d)'s, the 3rd and 4th terms are convex in (c,d)'s. This can be seen from a composition property of convex functions:

Proposition 2.1 (chapter 3, [1]). *Let $H_j(y)$, $j=1,2,\cdots,m$, be convex in $y\in\mathbb{R}^n$, then the function $\log(\sum_{j=1}^{m}\exp\{H_j(y)\})$ is convex in y.*

Proposition 2.1 follows from the function $\log(\sum_{j=1}^{m}e^{z_j})$ being convex and non-decreasing in each argument z_j, and H_j being convex. The 3rd and 4th terms of (2.7) are convex because $H_1=c_0+\sum_{i=1}^{H}c_i\phi(r_i(x_{j,m}\cdot p))$ is linear in c_i's and $H_2=d_0+\sum_{i=1}^{H}d_i\phi(r_i(x_{j,m}\cdot p))$ is linear in d_i's, $i=0,1,\cdots,H$, $j=1,2$. Hence *the objective is convex and smooth in each pair of (c_0,d_0) or (c_j,d_j) with (r,p)'s fixed.*

Consider minimizing $2\,M\,J_{QRCE}$ over (c_i,d_i) for some i, with (c_j,d_j) $(j\neq i)$ and (r_l,p_l), $\forall l$, fixed. The quadratically regularized two dimensional sub-problem takes the form:

$$
\begin{aligned}
J_1 &= -c_i\sum_{m=1}^{M}\phi_{1,i,m} - d_i\sum_{m=1}^{M}\phi_{2,i,m} + \sum_{m=1}^{M}\log(A_{i,m}\,e^{c_i\varphi_{1,i,m}}+B_{i,m}\,e^{d_i\phi_{1,i,m}})\\
&+\sum_{m=1}^{M}\log(E_{i,m}\,e^{c_i\varphi_{2,i,m}}+F_{i,m}\,e^{d_i\phi_{2,i,m}})+\lambda\,(c_i^2+d_i^2)/2,
\end{aligned}
\tag{2.9}
$$

where $\phi_{1,0,m} = \phi_{2,0,m} = 1$, $\phi_{k,i,m} := \phi(r_i(x_m^k \cdot p))$ $(k = 1, 2)$, if $i \geq 1$; $\lambda \in (0, 1)$; and the positive quantities $(A_{i,m}, B_{i,m}, E_{i,m}, F_{i,m})$ are defined as follows:

$$\begin{aligned} A_{i,m} &:= \exp\{c_0 + \sum_{j=1, j\neq i}^{H} c_j \phi(r_j(x_{1,m} \cdot p))\},\ B_{i,m} := \exp\{d_0 + \sum_{j=1, j\neq i}^{H} d_j \phi(r_j(x_{1,m} \cdot p))\} \\ E_{i,m} &:= \exp\{c_0 + \sum_{j=1, j\neq i}^{H} c_j \phi(r_j(x_{2,m} \cdot p))\},\ F_{i,m} := \exp\{d_0 + \sum_{j=1, j\neq i}^{H} d_j \phi(r_j(x_{2,m} \cdot p))\} \end{aligned}$$

Setting the gradient of J_1 to zero gives the critical point equations:

$$0 = \frac{\partial J_1}{\partial c_i} = \lambda\, c_i - \sum_{m=1}^{M} \phi_{1,i,m} + \sum_{m=1}^{M} \frac{A_{i,m}\, \phi_{1,i,m} e^{c_i \phi_{1,i,m}}}{A_{i,m}\, e^{c_i \phi_{1,i,m}} + B_{i,m}\, e^{d_i \phi_{1,i,m}}} + \sum_{m=1}^{M} \frac{E_{i,m}\, \phi_{2,i,m} e^{c_i \phi_{2,i,m}}}{E_{i,m}\, e^{c_i \phi_{2,i,m}} + F_{i,m}\, e^{d_i \phi_{2,i,m}}} \tag{2.10}$$

and

$$0 = \frac{\partial J_1}{\partial d_i} = \lambda\, d_i - \sum_{m=1}^{M} \phi_{2,i,m} + \sum_{m=1}^{M} \frac{B_{i,m}\, \phi_{1,i,m}\, e^{d_i \phi_{1,i,m}}}{A_{i,m}\, e^{c_i \phi_{1,i,m}} + B_{i,m}\, e^{d_i \phi_{1,i,m}}} + \sum_{m=1}^{M} \frac{F_{i,m}\, \phi_{2,i,m}\, e^{d_i \phi_{2,i,m}}}{E_{i,m}\, e^{c_i \phi_{2,i,m}} + F_{i,m}\, e^{d_i \phi_{2,i,m}}}. \tag{2.11}$$

For any $\lambda > 0$, adding (2.10) and (2.11) yields:

$$\lambda(c_i + d_i) = 0, \quad \text{or} \quad d_i = -c_i. \tag{2.12}$$

Equation (2.12) provides a reduction to a single variable in searching for a minimal point. It suffices to consider (2.10) which becomes:

$$\sum_{m=1}^{M} \frac{B_{i,m}\, \phi_{1,i,m}\, e^{-c_i \phi_{1,i,m}}}{A_{i,m}\, e^{c_i \phi_{1,i,m}} + B_{i,m}\, e^{-c_i \phi_{1,i,m}}} = \lambda\, c_i + \sum_{m=1}^{M} \frac{E_{i,m}\, \phi_{2,i,m}\, e^{c_i \phi_{2,i,m}}}{E_{i,m}\, e^{c_i \phi_{2,i,m}} + F_{i,m}\, e^{-c_i \phi_{2,i,m}}}$$

or:

$$\Gamma = \Gamma(c_i) := \lambda\, c_i + \sum_{m=1}^{M} \frac{E_{i,m}\, \phi_{2,i,m}}{E_{i,m} + F_{i,m}\, e^{-2c_i \phi_{2,i,m}}} - \sum_{m=1}^{M} \frac{B_{i,m}\, \phi_{1,i,m}}{B_{i,m} + A_{i,m}\, e^{2c_i \phi_{1,i,m}}} = 0. \tag{2.13}$$

Since the left hand side of (2.13) is strictly increasing in c_i from $-\infty$ to $+\infty$, there is a unique value of $c^*_{i,\lambda}$ satisfying (2.13), which can be obtained by a bisection algorithm (see e.g. subsection 14.5.1 of [15]).

Let us summarize the above as:

Proposition 2.2. *The reduced objective J_1 from $2\,M$ J_{QRCE} is strictly convex and has unique minimal point when restricted to the two variables (c_i, d_i) for any $i = 0, 1, \cdots, H$ with other variables fixed. The minimal point is of the form $c_i(1, -1)$ with c_i equal to the unique root $c^*_{i,\lambda}$ of the scalar increasing function Γ of (2.13).*

The next step is to fix all $(c_l, d_l) = (c_l, -c_l)$'s and minimize $2\,M\,J_{QRCE}$ over r_i for some i with p and r_j fixed ($j \neq i$). Since the r_i variable is inside the piecewise linear function φ, the objective is piecewise convex. This is also the case when updating p componentwise with (c, d, r) fixed. See Fig. 2. I shall treat these sub-problems in detail in the setting of multi-class classification model of the next subsection. By the end of p update, one has completed a full cycle of AM iteration, which repeats till convergence of the objective values.

2.3 Multi-class Classification Model

I extend my neural network model with a projection inside a hidden layer from binary to multi-class problems. Let p_j be the "principal directions" ("neural discriminant vectors"), $j = 1 \cdots, J$. If C is the class number, J is typically $C-1$, similar to LDA [3]. Let the input vector be $\vec{x}_{in} = (x_{in,1}, x_{in,2}, \cdots, x_{in,D}) \in \mathbb{R}^D$. The network output (excitation) is:

$$z_c = b_c + \sum_{i=1}^{H} w_{ci}\, \varphi(\sum_{j=1}^{J} r_{ij}\, (\vec{x}_{in} \cdot \vec{p}_j)), \tag{2.14}$$

where $\vec{x}_{in} \cdot \vec{p}_j$ is the inner product of $\vec{x}_{in}$ and $\vec{p}_j \in \mathbb{R}^D$, and c is a class label from 1 to C. The output class probabilities are:

$$P_c(\vec{x}_{in}) = \frac{\exp\{z_c(\vec{x}_{in})\}}{\sum_{n=1}^{C} \exp\{z_n(\vec{x}_{in})\}}. \tag{2.15}$$

The vector arrows on x_{in} and p_j will be ignored from here on. The m-th input vector from class c is $x_{c,m}$. The excitation of the standard one hidden layer network is of the form:

$$z_c = b_c + \sum_{i=1}^{H} w_{ci}\, \varphi(\sum_{k=1}^{D} u_{ik} x_{in,k}). \tag{2.16}$$

Notice that the weight matrix inside φ of (2.14) is a factored from of (u_{ik}) in (2.16). The total number of parameters in (2.14) is $JD + JH + CH + C$ while that of (2.16) is $HD + CH + C$. Typically $H = \rho J$, $J = C - 1$, $\rho = 2(3)$ to reduce variance, then the number of parameters of (2.16) exceeds that of (2.15) by $(\rho - 1)JD - JH = J[(\rho - 1)D - H] = (C - 1)[(\rho - 1)D - \rho(C - 1)]$. For $D = 420$, $C \in [3, 10]$, $\rho = 2(3)$, this number is in the range $[832, 3618]$ ($[1668, 7317]$). Let $U = (u_{ij})$, $R = (r_{ij})$, $P =$ stacking up p_j's, then (2.14) is a low rank ($=J$) approximation of (2.16) in the sense that $U \approx RP$, $\text{rank}(U) = H > J$.

The multi-class quadratically regularized cross entropy function is:

$$\begin{aligned}\text{MQRCE} &= \frac{\lambda}{2}\|(b,w,r,p_1,p_2,\cdots,p_J)\|^2 - \sum_{m=1}^{M}\sum_{c=1}^{C}\log\, P_c(x_{c,m}) \\ &= \frac{\lambda}{2}\|(b,w,r,p_1,p_2,\cdots,p_J)\|^2 - \sum_{m=1}^{M}\sum_{c=1}^{C}[b_c + \sum_{i=1}^{H} w_{ci}\,\varphi(\sum_{j=1}^{J} r_{ij}(x_{c,m}\cdot p_j))] \\ &\quad + \sum_{m=1}^{M}\sum_{c=1}^{C}\log(\sum_{n=1}^{C}\exp\{b_n + \sum_{i=1}^{H} w_{ni}\,\varphi(\sum_{j=1}^{J} r_{ij}(x_{c,m}\cdot p_j))\}). \end{aligned} \tag{2.17}$$

At fixed (r,p_j) $(j=1,\cdots,J)$, MQRCE is smooth and strictly convex in (b,w). At the minimal point, I have $\frac{\partial MQRCE}{\partial b_k}=0$, $\forall k=1,\cdots,C$, or:

$$0 = \lambda\, b_k - M + \sum_{m=1}^{M}\sum_{c=1}^{C}\frac{\exp\{b_k + \sum_{i=1}^{H} w_{ki}\,\varphi(\sum_{j=1}^{J}\, r_{ij}\,(x_{c,m}\cdot p_j))\}}{\sum_{n=1}^{C}\exp\{b_n + \sum_{i=1}^{H} w_{ni}\,\varphi(\sum_{j=1}^{J} r_{ij}\,(x_{c,m}\cdot p_j))\}} \tag{2.18}$$

which gives by summing over $k=1,\cdots,C$:

$$\sum_{k=1}^{C} b_k = 0. \tag{2.19}$$

The estimate:

$$\lambda b_k - M \le \frac{\partial MQRCE}{\partial b_k} \le \lambda b_k + M(C-1) \tag{2.20}$$

gives the bisection interval $[-1+\lambda^{-1}(1-C)M, 1+\lambda^{-1}M]$ for b_k of equation (2.18).

Similarly, I have from $\frac{\partial QRCE}{\partial w_{ki}}=0$:

$$\begin{aligned} 0 &= \lambda\, w_{ki} - \sum_{m=1}^{M}\varphi(\sum_{j=1}^{J} r_{ij}\,(x_{k,m}\cdot p_j)) \\ &+ \sum_{m=1}^{M}\sum_{c=1}^{C}\frac{\varphi(\sum_{j=1}^{J} r_{ij}\,(x_{c,m}\cdot p_j))\exp\{b_k + \sum_i w_{ki}\,\varphi(\sum_j r_{ij}\,(x_{c,m}\cdot p_j))\}}{\sum_{n=1}^{C}\exp\{b_n + \sum_i w_{ni}\varphi(\sum_j r_{ij}\,(x_{c,m}\cdot p_j))\}} \end{aligned} \tag{2.21}$$

implying when summing over $k=1,\cdots,C$:

$$\sum_{k=1}^{C} w_{ki} = 0, \quad \forall\, i=1,\cdots,H. \tag{2.22}$$

The derivative bounds:

$$\frac{\partial MQRCE}{\partial w_{ki}} \ge \lambda\, w_{ki} - \sum_{m=1}^{M}\varphi(\sum_{j=1}^{J} r_{ij}\,(x_{k,m}\cdot p_j)),$$

and

$$\frac{\partial MQRCE}{\partial w_{ki}} \leq \lambda\, w_{ki} - \sum_{m=1}^{M} \varphi(\sum_{j=1}^{J} r_{ij}\,(x_{k,m}\cdot p_j)) + \sum_{m=1}^{M}\sum_{c=1}^{C} \varphi(\sum_{j=1}^{J} r_{ij}\,(x_{c,m}\cdot p_j)),$$

give the following bisection interval on w_{ki} of equation (2.21):

$$[-1-\lambda^{-1}\sum_{m=1}^{M}\sum_{c\neq k}\varphi(\sum_{j=1}^{J} r_{ij}\,(x_{c,m}\cdot p_j)), 1+\lambda^{-1}\sum_{m=1}^{M}\varphi(\sum_{j=1}^{J} r_{ij}\,(x_{k,m}\cdot p_j))]. \tag{2.23}$$

For fixed (b, w, r), I update $p_{j,k}$, the k-th component of p_j, by writing the sum inside φ as:

$$\begin{aligned}
\sum_{l=1}^{J} r_{il}\,(x_{c,m}\cdot p_j) &= r_{ij}\,(x_c\cdot p_j) + \sum_{l\neq j} r_{il}\,(x_{c,m}\cdot p_l) \\
&= r_{ij}\,x_{c,m,k}p_{j,k} + r_{ij}(\sum_{l\neq k} x_{c,m,l}\,p_{j,l}) + \sum_{l\neq j} r_{il}\,(x_{c,m}\cdot p_l) \\
&:= r_{ij}\,x_{c,m,k}\,p_{j,k} + \tau_{c,i,m}.
\end{aligned} \tag{2.24}$$

I sort the numbers $\tau_{c,i,m}/(r_{ij}\,x_{c,m,k})$ in increasing order as $-\infty < a_1 < a_2 < \cdots < a_{S_1} < +\infty$, $S_1 \leq MCH$ (excluding any (m,c,i) where $r_{ij}\,x_{c,m,k}=0$). Over each finite interval $(-a_{s+1}, -a_s)$, $s=1,\cdots,S_1$, the objective is convex and the local minimum is found via bisection. Over the semi-infinite interval $(-a_1,+\infty)$, an interior minimum exists if and only if the objective is decreasing near $-a_1$ which can be detected by comparing the objective value at a point slightly to the right of $-a_1$ with that at $-a_1$. If the local minimum exists, I determine the bisection interval by lower bounding the partial derivative:

$$\begin{aligned}
&\frac{\partial\, MQRCE}{\partial p_{j,k}} = \lambda p_{j,k} - \sum_{m,i,c} w_{ic}\,r_{ij}\,x_{c,m,k}\;\varphi'(\sum_{j} r_{ij}\,(x_{c,m}\cdot p_j)) \\
+\;& \sum_{m,c} \frac{\sum_n(\sum_i w_{in}\,r_{ij}\,x_{c,m,k}\varphi'(\sum_{l=1}^{J} r_{il}\,(x_{c,m}\cdot p_l)))\exp\{b_n+\sum_i w_{ni}\,\varphi(\sum_{l=1}^{J} r_{il}\,(x_{c,m}\cdot p_l))\}}{\sum_n \exp\{b_n+\sum_i w_{ni}\,\varphi(\sum_{l=1}^{J} r_{il}\,(x_{c,m}\cdot p_l))\}} \\
\geq\;& \lambda p_{j,k} - \sum_{m,c,i}|x_{c,m,k}\,w_{ci}\,r_{ij}| - (\sum_{m,c}|x_{c,m,k}|)\,(\max_n \sum_i |w_{ni}\,r_{ij}|),
\end{aligned} \tag{2.25}$$

which is positive if

$$\lambda\, p_{j,k} > \sum_{m,c}|x_{c,m,k}|\;\sum_i |w_{ci}\,r_{ij}| + (\sum_{m,c}|x_{c,m,k}|)\,(\max_n \sum_i |w_{ni}\,r_{ij}|).$$

In particular, a finite bisection interval on $(-a_1,+\infty)$ is:

$$[-a_1, 1+|a_1|+\lambda^{-1}\sum_{m,c}|x_{c,m,k}|\sum_i|w_{ci}\,r_{ij}| + \lambda^{-1}(\sum_{m,c}|x_{c,m,k}|)\,(\max_n\sum_i|w_{ni}\,r_{ij}|)]. \tag{2.26}$$

Similarly, a bisection interval for locating the minimum of the objective over $p_{j,k} < -a_{S_1}$ is:

$$[-1 - |a_S| - \lambda^{-1} \sum_{m,c} |x_{c,m,k}| \sum_i |w_{ci}\, r_{ij}| - \lambda^{-1} (\sum_{m,c} |x_{c,m,k}|)\, (\max_n \sum_i |w_{ni}\, r_{ij}|), -a_S]. \quad (2.27)$$

For fixed (b, w, p), I update r_{ij} by writing the inner sum inside φ as:

$$\sum_l r_{il}(x_{c,m} \cdot p_l) = r_{ij}\,(x_{c,m} \cdot p_j) + \sum_{l \neq j} r_{il}\,(x_{c,m} \cdot p_l) := r_{ij}\,(x_{c,m} \cdot p_j) + \nu_{c,m}. \quad (2.28)$$

I sort the numbers $\nu_{c,m}/(x_{c,m} \cdot p_j)$ in increasing order as α_s, $s = 1, \cdots, S_2$, $S_2 \leq MC$ (excluding any (c, m) where $(x_{c,m} \cdot p_j) = 0$). Over each finite interval $(-\alpha_{s+1}, -\alpha_s)$, $s = 1, \cdots, S_2$, the objective is convex and the local minimum is found via bisection. Over the semi-infinite interval $(-\alpha_1, +\infty)$, an interior minimum exists if and only if the objective is decreasing near $-\alpha_1$. If the local minimum exists, I bound the partial derivative from below:

$$\begin{aligned}
&\frac{\partial\, MQRCE}{\partial\, r_{ij}} = \lambda\, r_{ij} - \sum_{m,c} w_{ci}\,(x_{c,m} \cdot p_l)\; \varphi'(\sum_l r_{il}\,(x_{c,m} \cdot p_l)) \\
+ \;& \sum_{m,c} \frac{\sum_n w_{ni}\,(x_{c,m} \cdot p_j)\, \varphi'(\sum_{l=1}^J r_{il}\,(x_{c,m} \cdot p_l)))\, \exp\{b_n + \sum_{h=1}^H w_{nh}\, \varphi(\sum_{l=1}^J r_{lh}\,(x_{c,m} \cdot p_l))\}}{\sum_n \exp\{b_n + \sum_{h=1}^H w_{nh}\, \varphi(\sum_{l=1}^J r_{hl}\,(x_{c,m} \cdot p_l))\}} \\
\geq \;& \lambda\, r_{ij} - \sum_{m,c} \max_n |w_{ni}\,(x_{c,m} \cdot p_j)| - \sum_{m,c} |w_{ci}\,(x_{c,m} \cdot p_j)|, \quad (2.29)
\end{aligned}$$

which is positive if:

$$\lambda\, r_{ij} > \sum_{m,c} \max_n |w_{ni}\,(x_{c,m} \cdot p_j)| + \sum_{m,c} |w_{ci}\,(x_{c,m} \cdot p_j)|.$$

A bisection interval for $r_{ij} > -\alpha_1$ is:

$$[-\alpha_1, 1 + |\alpha_1| + \lambda^{-1}(\sum_{m,c} \max_n |w_{ni}\,(x_{c,m} \cdot p_j)| + |w_{ci}\,(x_{c,m} \cdot p_j)|)]. \quad (2.30)$$

Similarly, if a local minimum exists over $r_{ij} < -\alpha_S$, a bisection interval is:

$$[-1 - |\alpha_S| - \lambda^{-1}(\sum_{m,c} \max_n |w_{ni}\,(x_{c,m} \cdot p_j)| + |w_{ci}\,(x_{c,m} \cdot p_j)|), -\alpha_S]. \quad (2.31)$$

Note that MQRCE as a piecewise convex function of r_{ij}, consists of multiple pieces (generically $M\,C+2$). In Fig. 2, the multi-piece convex structure of MQRCE in a component of r (left) and p (right) is illustrated through two instances during network training.

The alternating minimization method for (2.17) based on the above analysis is summarized in Algorithm 1.

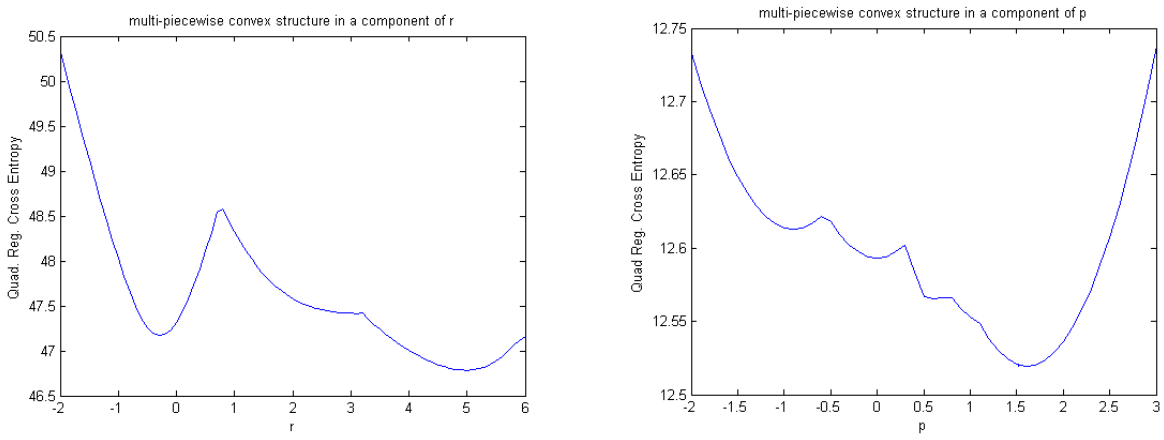


Figure 2: Left (Right): Multi-piecewise convex structure of the quadratically regularized average cross entropy function in a component of r (p) in multi-class classification.

Algorithm 1 : AM for Quadratically Regularized Multi-Class Cross Entropy (2.17).

Input: data $(x_{c,m})$, $c = 1, \cdots, C$, $m = 1, \ldots, M$, centered at zero; positive integer H.
Initialize vectors: $(b, w, r, p_j)^{(0)}$, $j = 1, \cdots J$.

for $l = 0, 1, \ldots$ **do**
  for $k = 1, \ldots, C$, **do**
    $b_k \leftarrow$ the unique root of (2.18) via bisection, while fixing b_n, $n \neq k$, and (w, r, p);
  end for
  for $k = 1, \ldots, C; i = 1, \ldots, H$, **do**
    $w_{ki} \leftarrow$ the unique root of (2.21) via bisection, while fixing w_{cj}, $c \neq k$, $j \neq i$, and (c, d, p).
  end for
  for $j = 1, \ldots, D; k = 1, \ldots, J$, **do**
    update $p_{j,k}$ by minimizing MQRCE via bisection on sub-intervals formed by the finite sequence (a_s), $s = 1, \ldots, S_1 \leq MCH$.
  end for
  for $i = 1, \ldots, H; j = 1, \ldots, J$, **do**
    update r_{ij} by minimizing MQRCE via bisection on sub-intervals formed by the finite sequence (α_s), $s = 1, \ldots, S_2 \leq MC$.
  end for
end for

Output at convergence $l = L$: $(b, w, r, p_j)^{(L)}$, $j = 1, \ldots, J$.

2.4 Convergence Analysis

I analyze convergence of the AM algorithm 1. Since the cross entropy function is nonnegative, and each subproblem reduces the objective, the AM method is descending and its objective values $\mathrm{MQRCE}((b,w,r,p)^{(l)})$ converge to a finite non-negative limit. The iterates are uniformly bounded:

$$\|(b,w,r,p)^{(l)}\|_2^2 \leq 2\lambda^{-1}\mathrm{MQRCE}((b,w,r,p)^{(l)}) \leq 2\lambda^{-1}\mathrm{MQRCE}((b,w,r,p)^{(0)}), \quad \forall l \geq 1. \tag{2.32}$$

I prove further that:

Proposition 2.3. *If in the (r,p) update of the AM algorithm 1, the nearest local minimum is used instead of a global minimum, then the two adjacent iterates $(b,w,r,p)^{(l+1)}$, $(b,w,r,p)^{(l)}$ satisfy the inequality:*

$$MQRCE((b,w,r,p)^{(l)}) - MQRCE((b,w,r,p)^{(l+1)}) \geq \eta\,|(b,w,r,p)^{(l)} - (b,w,r,p)^{(l+1)}|^2 \tag{2.33}$$

for some positive constant $\eta > 0$ independent of iteration number l. It follows that:

$$\lim_{l\to+\infty} |(b,w,r,p)^{(l)} - (b,w,r,p)^{(l+1)}| = 0. \tag{2.34}$$

Proof: In case of (b,w) updates, the objective is smooth and strongly convex, with second partial derivative in each coordinate of (b,w) above λ. Since $(b,w)^{(l+1)}$ differs from $(b,w)^{(l)}$ in one coordinate (call it ξ), and is the minimum in ξ, I have that (2.33) holds for positive constant $\eta = \lambda/4$ if $|(b,w,r,p)^{(l)} - (b,w,r,p)^{(l+1)}|$ is small enough. Here I assume that the ξ component of $(b,w)^{(l)}$ is not at the minimum, otherwise, $(b,w)^{(l+1)} = (b,w)^{(l)}$, and (2.33) is true for any η. On the other hand, the objective function is above the quadratic $\lambda\|(b,w)\|^2/2$. With $\eta = \lambda/4$, (2.33) is valid for any $(b,w,r,p)^{(l)}$ away from $(b,w,r,p)^{(l+1)}$.

In case of (r,p) updates, the objective is Lipschitz continuous, piecewise smooth and strongly convex. Assume that the coordinate of $(r,p)^{(l)}$ to be updated is not a local minimum of the objective in this coordinate with all other variables fixed at iteration step l. If the local minimum for the updated coordinate ξ falls in the interior of a sub-interval, the objective is smooth and locally strongly convex, the above argument on (b,w) update applies. If the local minimum occurs at an end point of a sub-interval, piecewise strong convexity implies that (2.33) remains valid for $\xi^{(l)}$ in the left and right intervals around the minimum with some positive constants η_- and η_+ respectively. Choosing $\eta = \min(\eta_-, \eta_+)$, I conclude (2.33). □

Next I prove:

Theorem 2.1. *If in the (r,p) update of the AM algorithm 1, the nearest local minimum is used instead of a global minimum, then the sequence $(b,w,r,p)^{(l)}$ converges subsequentially to a first order stationary point $(\bar{b},\bar{w},\bar{r},\bar{p})$ in the sense that:*

$$\nabla_{b,w}\,\mathrm{MQRCE}(\bar{b},\bar{w},\bar{r},\bar{p}) = 0, \quad 0 \in \partial_{r,p}\,\mathrm{MQRCE}(\bar{b},\bar{w},\bar{r},\bar{p}), \tag{2.35}$$

where the partial $\partial_{r,p}$ denotes sub-differential or the set of all sub-gradients in (r,p).

Proof: Let $(b, w, r, p)^{(l_j)}$ be a convergent subsequence consisting of iterates at the end of outer iterations. By construction, an update in each coordinate ξ via bisection in the inner iteration is a local minimum, so

$$0 \in \partial_\xi \text{ MQRCE}(\xi^{l_j+1}, \cdots), \tag{2.36}$$

where the dots denote all the other coordinates either at l_j step values or $l_j + 1$ step values updated before ξ coordinate. The sub-differential in (2.36) is a closed interval with left and right end values being the left and right derivatives on either side of $\xi^{(l_j+1)}$ if it is located at the end point of a sub-interval in (r, p) update, otherwise the sub-differential is zero. By (2.34), $(b, w, r, p)^{(l_j)}$, $(b, w, r, p)^{(l_j+1)}$, and all intermediate values in the inner iterations in between, converge to $(\bar{b}, \bar{w}, \bar{r}, \bar{p})$, as $j \to \infty$. The end points of sub-intervals in ξ, satisfying the bound (2.32), continuously depend on the variables inside $\cdots$, so as $j \to \infty$:

$$\partial_\xi \text{ MQRCE}(\xi^{(l_j+1)}, \cdots) \to \partial_\xi \text{ MQRCE}(\bar{b}, \bar{w}, \bar{r}, \bar{p}),$$

implying:

$$0 \in \partial \text{ MQRCE}(\bar{b}, \bar{w}, \bar{r}, \bar{p}).$$

Since MQRCE is smooth in (b, w), piecewise smooth and convex in (r, p), (2.35) follows. □

3 Numerical Experiments

In this section, I present numerical results of the neural network model (2.14)-(2.15) to thought recognition experiment. The first part of the experiment is a five to ten minute recording of EEG waves while a subject is sitting down and thinking of one of the four daily thoughts (such as resting, reading, walking, eating). The recording is either by a 14 channel wireless Epoc+ headset [5] at a home environment, or by a traditional EEG cap with 66 conductive gel lubricated electrodes in a lab environment. The recorded brain waves are transmitted to a computer and digitized at 256 Hz sampling frequency. The second part of experiment is data processing, training and testing of neural network models in thought classification. The first type of brain wave data is measured from a normal individual by myself. I recorded the wireless data at home environment following Emotive user guide [5]. The headset is slided on the head with saline hydrated felt pads installed on the sensors. The second type of data is measured from an Alzheimer subject in a lab environment by a trained professional using a traditional EEG cap at UC Irvine [18].

The recorded raw brain waves are in physical unit of microvolt, an illustration is in the left frame of Fig. 3. The raw brain signals are band-passed to the frequency range $[1, 42]$ Hz, removing low frequency (under 0.16 Hz) background, and high frequency disturbance (e.g. muscle movement). For each second of time domain samples from each channel, a power spectrum density vector of dimension 30 is calculated. The right frame of Fig. 3 plots four power spectrum density vectors of dimension 30 each, showing distinct shapes (peak and valley structures). The data matrix X is a stacking up of row vectors combining

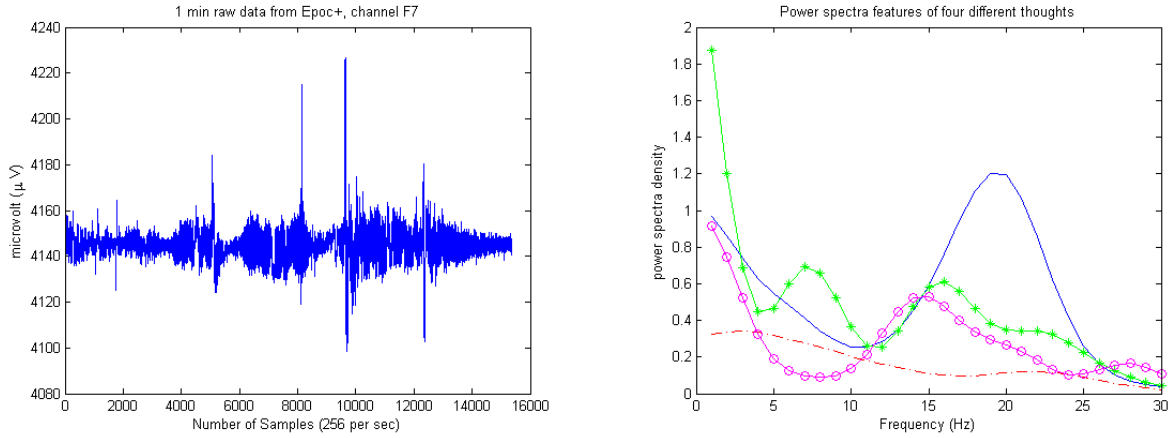
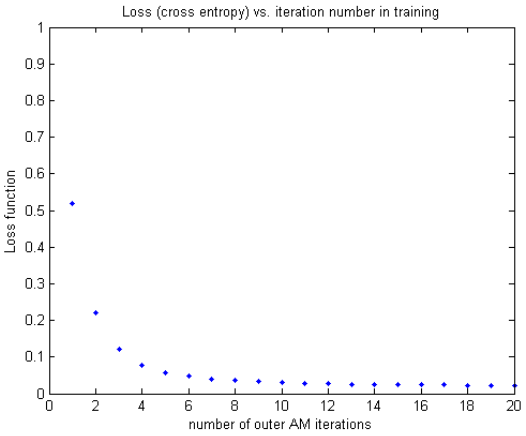


Figure 3: Left: one minute of raw data (microvolt) from electrode F7 of Emotiv headset (Epoc+). Right: sample power spectrum densities of four thoughts vs. frequency.

power spectra from all electrodes. So each row vector has dimension (D) equal to 30 times the number of electrodes ($D = 420$ for wireless Epoc+, $D = 1980$ for a traditional cap). The number of rows n is roughly the duration of recording in seconds times the number of thoughts (times 2 if the power spectra come from 50 % overlapping windows). For 5 to 10 minute recording, n is in the range of $[300, 600]$ times the number of thoughts for non-overlapping spectral windows. The pre-processing step above produces an $n \times D$ data matrix for a classification study where 50 to 80 % of the data (recorded earlier in time) will be used for training, the remaining data (recorded later in time) for testing. Due to such arrangement, the classification becomes a prediction problem, which is meaningful for the classifier to assist late stage dimentia subjects with data collected from them earlier.

The regularization parameter of (2.17) is $\lambda = 1$. The tolerance width to terminate the bisection method in the inner iterations is 0.001. Network parameters are: $H = 2J$, $J = C - 1$. The (b, w, r) are initialized independently from a unit normal distribution (cold start). The p is initialized (warm start) by sparse linear discriminant analysis [3], via slda function of the SpaSM (sparse statistical modeling) toolbox in Matlab. The AM method is descending and typically converges in 10 outer iterations (left frame of Fig. 4). The loss function in a stochastic gradient descent (SGD) method is oscillatory (right frame of Fig. 4). Since SGD computes on small random samples (mini-batches) of data instead of the entire training set, its runtime of an epoch (a full sweep through training data) is much shorter than that of an AM cycle, by as much as a factor of 10.

I set the number of outer iterations to 10, and perform 6 runs with independent random initialization except for p. The average accuracies of thought prediction and the standard deviations for an Alzheimer subject are in Table 2 according to 50 to 80% of training data. The high dimensional feature vectors (the training part of the row vectors of data matrix X) are projected to (P_1, P_2) plane in the 3 thought case and (P_1, P_2, P_3) space in the 4 thought



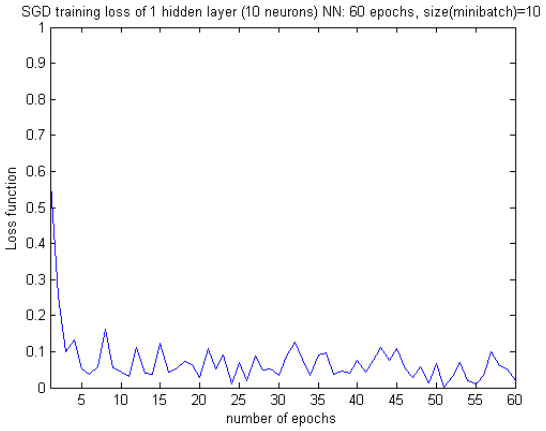


Figure 4: Left: monotone decrease of quadratically regularized cross entropy (2.17) in the number of outer iterations of AM method. Right: oscillatory behavior of the loss function vs. the number of epochs (full sweeps through training data) of the stochastic gradient descent method in a similar neural network training.

case. They are shown in Fig. 5 in case of the 80% training data. One sees that the 3 and 4 clusters are cleanly separated by colors corresponding to the class labels, indicating that the trained neural discriminant vectors produce excellent low dimensional features. The classification on the test data depends also on its variation from the training data. The more the subject is focused on the thought, the better the test accuracy measured as percentage correct in thought classification.

Table 1: Prediction accuracy by model (2.14) (average percentage correct and standard deviation over 6 random starts) for an Alzheimer subject using 66 channel EEG cap recordings of 6 minutes. Four thoughts are: reading, resting, walking and eating. Three thoughts are: reading, resting, and walking. Two thoughts are: reading and resting. Each row vector of the first column lists 3 percentages of the training data for the (4,3,2) thoughts. To present enough variation in the accuracies, the 3 percentages are not the same in case of (75,70,70)%.

%(training data)	**4 thoughts**	3 thoughts	2 thoughts
(80,80,80)%	**94.92% (2.52%)**	98.91% (0.37%)	99.48 % (1.61%)
(75,70,70)%	**90.68% (1.48%)**	96.28% (0.85%)	98.95% (1.74%)
(70,60,60)%	**85.91% (0.89%)**	96.95% (0.48%)	98.75% (0.67%)
(50,50,50)%	**80.15% (2.61%)**	95.75% (2.79%)	96.61% (0.12%)

The average accuracies of thought prediction and the standard deviations for a normal subject are in Table 2. The feature vectors projected to (P_1, P_2) plane in the 3 thought

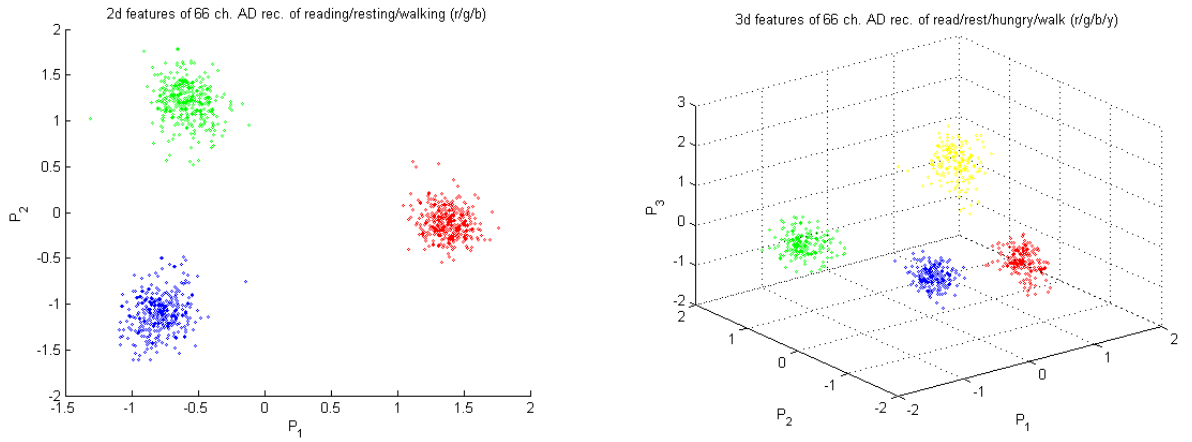


Figure 5: Left: 2 dimensional features of 66 channel EEG recording of an Alzheimer subject on the (P_1, P_2) plane after model (2.14) training on 80% of data for three thoughts (reading, resting, restroom) corresponding to colors (red, green, blue). Right: 3 dimensional features of 66 channel EEG recording of the Alzheimer subject in the (P_1, P_2, P_3) space after model (2.14) training on 80% of data for four thoughts (reading, resting, eating, walking) corresponding to colors (red, green, blue, yellow) respectively.

case and (P_1, P_2, P_3) space in the 4 thought case are shown in Fig. 6. The separation of clusters from 14 sensors is less than that from the 66 channel EEG cap in Fig. 5, due to less spatial resolution. Note also that the shapes of the clusters in Fig. 6 are quite different from each other while those in Fig. 5 are more uniform. Despite the non-uniform feature shapes, the neural network model (2.17) maintains the accuracies in the mid and upper ninety percentages. The better focus of attention from a normal subject also helps.

I compare my model (2.14) trained by AM with the standard NN model (2.16) trained by SGD in Matlab NN toolbox. The average prediction accuracies and standard deviations are calculated from the testing results of trained network at the end of $(10, 20, \cdots, 60)$ epochs. Results from 66 channel EEG cap's 6 minute recording for an Alzheimer subject (same as Table 1) are in Table 3 and those from 10 minute Epoc+ headset recording (same as Table 2) for a normal subject are in Table 4. In both Tables, the AM and SGD trainings are comparable for the 2 and 3 thoughts, indicating that 10 outer AM iterations reach a similar stationary point as the SGD over 10 to 60 epochs. However, the AM training is significantly better in the more challenging case of 4 thoughts (boldfaced). The projection to the (P_1, P_2, P_3) space captures the essential features and filters out noise in the data.

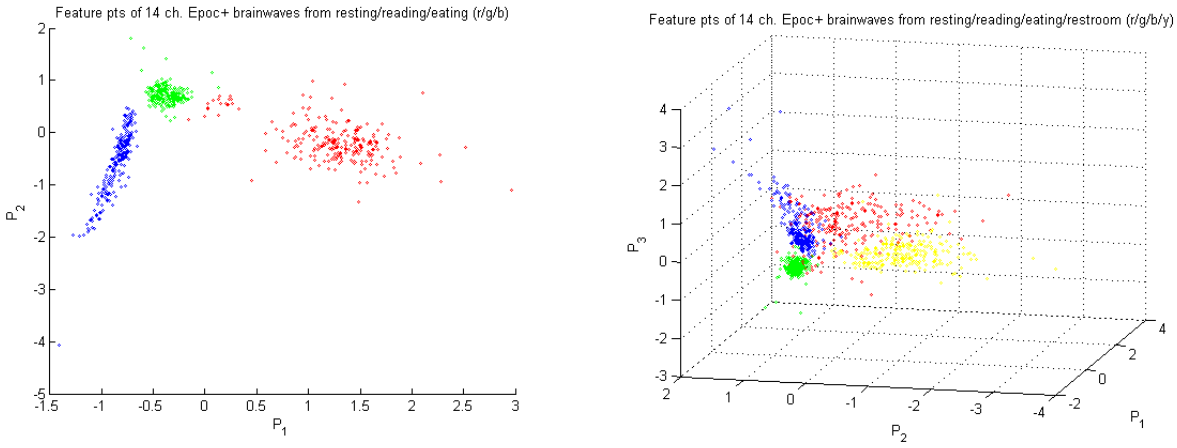


Figure 6: Left: 2 dimensional features of 14 channel wireless Epoc+ headset recording of a normal subject on the (P_1, P_2) plane after model (2.14) training on 80% of data for three thoughts (resting, reading, eating) corresponding to colors (red, green, blue). Right: 3 dimensional features of 14 channel wireless Epoc+ headset recording in the (P_1, P_2, P_3) space after model (2.14) training on 80% of data for four thoughts (resting, reading, eating, restroom) corresponding to colors (red, green, blue, yellow) respectively.

Table 2: Prediction accuracy by model (2.14) (average percentage correct and standard deviation over 6 random starts) for a normal subject using 14 channel Epoc+ recordings of 10 minutes. Four thoughts are: reading, resting, eating, and restroom. Three thoughts are: reading, resting, and eating. Two thoughts are: reading and resting.

%(training data)	**4 thoughts**	3 thoughts	2 thoughts
80%	**94.87% (1.38%)**	96.47% (1.89%)	99.66 % (0.01%)
70%	**93.93% (2.73%)**	95.87% (2.35%)	99.10% (0.01%)
60%	**96.30% (0.73%)**	94.72% (3.40%)	98.96% (0.07%)
50%	**95.76% (1.21%)**	94.49% (3.40%)	98.94% (0.06%)

Table 3: Prediction accuracy by model (2.16) trained by stochastic gradient descent method (average percentage correct and standard deviation over 60 epochs) for an Alzheimer subject using 66 channel EEG cap recordings of 6 minutes. Four thoughts are: reading, resting, walking and eating. Three thoughts are: reading, resting, and walking. Two thoughts are: reading and resting. Each row vector of the first column lists 3 percentages of the training data for the (4,3,2) thoughts.

%(training data)	4 thoughts	3 thoughts	2 thoughts
(80,80,80)%	74.47% (2.52%)	97.79% (2.62%)	99.18 % (1.20%)
(75,70,70)%	74.68% (0.27%)	97.58% (1.46%)	97.11% (2.65%)
(70,60,60)%	74.10% (1.13%)	97.05% (1.40%)	98.47% (1.81%)
(50,50,50)%	70.82% (0.47%)	92.29% (3.67%)	98.06% (0.20%)

Table 4: Prediction accuracy by model (2.16) trained by stochastic gradient descent method (average percentage correct and standard deviation over 60 epochs) for a normal subject using 14 channel Epoc+ recordings of 10 minutes. Four thoughts are: reading, resting, eating, and restroom. Three thoughts are: reading, resting, and eating. Two thoughts are: reading and resting.

%(training data)	4 thoughts	3 thoughts	2 thoughts
80%	79.70% (5.05%)	97.91% (0.34%)	98.85 % (0.42%)
70%	75.10% (1.19%)	96.48% (1.42%)	98.51% (0.47%)
60%	75.98% (3.88%)	94.35% (3.19%)	98.73% (0.11%)
50%	73.96% (1.36%)	91.00% (9.23%)	98.34% (0.65%)

4 Concluding Remarks

I studied a neural network with low rank weights for increasing the accuracy of the classification of brain waves. Based on convex and piecewise convex structures of the training objective function, I developed the alternating minimization method, and proved that it is descending and convergent. The prediction of 4-class brain waves from normal and Alzheimer subjects outperforms by 10 percentage points the standard neural network trained by the stochastic gradient descent method. With enough EEG data collected in the future, it is promising that my neural network model (2.14) may be extended to a wider and deeper network for extracting features common to all subjects within a population so that thought classification is free from individual training (similar to speech recognition [22]).

5 Acknowledgements

I'd like to express my appreciation to Prof. Knut Solna at UC Irvine and Prof. Tom Hou at CalTech for their advice and encouragement during the project. I thank Mr. Matthew Richardson of the cognitive science department of UC Irvine for providing the EEG recording of the Alzheimer's subject in this study [18]. I appreciate the constructive comments from the anonymous referees for improving the paper. Last but not least, I thank my family for their support and enthusiasm.

References

[1] S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 7th edition, 2009.

[2] W. Byrne, *Alternating Minimization and Boltzmann Machine Learning*, IEEE Transactions on Neural Networks, 3(4), pp. 612-620, 1992.

[3] L. Clemmensen, T. Hastie, D. Witten and B. Ersboll, *Sparse Discriminant Analysis*, Technometrics, 53(4), pp. 406-413, 2011.

[4] https://en.wikipedia.org/wiki/Electroencephalography

[5] Emotiv.com.

[6] F. Fraga, T. Falk, P. Kanda, R. Anghinah, *Characterizing Alzheimer's Disease Severity via Resting-Awake EEG Amplitude Modulation Analysis*, PLoS ONE, 2013, 8(8).

[7] R. Glowinski, S. Osher, W. Yin, eds, "Splitting Methods in Communication, Imaging, Science and Engineering", Scientific Computation Series, Springer, 2016.

[8] G. Hinton, R. Salakhutdinov, *Reducing the Dimensionality of Data with Neural Networks*, Science, v. 313, pp. 504–507, 2006.

[9] P. Jain, P. Netrapalli, S. Sanghavi, *Low-rank Matrix Completion using Alternating Minimization*, Proc. 45th Annual ACM Symposium on Theory of Computing, 2013.

[10] C. King, P. Wang, L. Chui, A. Do, and Z. Nenadic, *Operation of a brain-computer interface walking simulator for individuals with spinal cord injury*, J. Neuro. Eng. Rehabil., vol. 10(77), 2013.

[11] Y. Li, J. Qin, Y-L. Hsin, S. Osher, W. Liu, *s-SMOOTH: Sparsity and Smoothness Enhanced EEG Brain Tomography*, Frontiers in Neuroscience, 10(543), 2016, doi:10.3389/fnins.2016.00543.

[12] G. Liberati, J. da Rocha, L. van der Heiden, A. Roffone, N. Birbaumer, M. Belardinelli, R. Sitaram, *Towards a Brain-Computer Interface for Alzheimer's Disease Patients by Combining Classical Conditioning and Brain State Classification*, J. Alzheimer's Disease, 31(2012), S211-S220, DOI 10.3233/JAD-2012-112129.

[13] F. Lotte, M. Congedo , A. Lécuyer, F. Lamarche, and B. Arnaldi, *A review of classification algorithms for EEG-based brain computer interfaces*, Journal of Neural Engineering, 4(2):R1–R13, 2007.

[14] I. Martisius, R. Damasevicius, *A Prototype SSVEP Based Real Time BCI Gaming System*, Comput. Intell. Neurosci., Epub 2016:3861425. doi: 10.1155/2016/3861425.

[15] T. Moon, W. Stirling, "Mathematical Methods and Algorithms for Signal Processing", Prentice Hall, Upper Saddle River, NJ, 2000.

[16] V. Nair, G. Hinton, *Rectified linear units improve restricted Boltzmann machines*, pp. 807-814, International Conference on Machine Learning, 2010.

[17] E. Neto, F. Biessmann F, H. Aurlien, H. Nordby, T. Eichele T, *Regularized Linear Discriminant Analysis of EEG Features in Dementia Patients*, Front Aging Neurosci, 30(8):273. eCollection 2016.

[18] M. Richardson, *EEG data on daily thoughts of an Alzheimer's subject recorded by a 66 electrode cap*, Department of Cognitive Science, UC Irvine, summer 2017.

[19] A. Subasi, M. Gursoy, *EEG signal classification using PCA, ICA, LDA, and support vector machines*, Expert Systems with Applications, 37(2010), pp. 8659–8666.

[20] G. Taylor, R. Burmeister, Z. Xu, A. Patel, T. Goldstein, *Training neural networks without gradients: A scalable admm approach*, International Conference on Machine Learning, pp. 2722-2731, 2016.

[21] I. Walker, *Deep Convolutional Neural Networks for Brain Computer Interface using Motor Imagery*, Master Thesis in Computer Science, Imperial College, London, 2015.

[22] D. Yu, L. Deng, "Automatic Speech Recognition: A Deep Learning Approach", Signals and Comm. Technology, Springer, 2015.