Article

EEG-Based Emotion Recognition by Retargeted Semi-Supervised Regression with Robust Weights

Ziyuan Chen 1, Shuzhe Duan 1,2 and Yong Peng 2,3, *

1 Zhiyue Honors College, Hangzhou Dianzi University, Hangzhou 310018, China
2 School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, China
3 Zhejiang Key Laboratory of Brain-Machine Collaborative Intelligence, Hangzhou 310018, China
* Correspondence: yongpeng@hdu.edu.cn

Abstract: The electroencephalogram (EEG) can objectively reflect the emotional state of human beings, and has attracted much attention in the academic circles in recent years. However, due to its weak, non-stationary, and low signal-to-noise properties, it is inclined to cause noise in the collected EEG data. In addition, EEG features extracted from different frequency bands and channels usually exhibit different levels of emotional expression abilities in emotion recognition tasks. In this paper, we fully consider the characteristics of EEG and propose a new model RSRRW (retargeted semi-supervised regression with robust weights). The advantages of the new model can be listed as follows. (1) The probability weight is added to each sample so that it could help effectively search noisy samples in the dataset, and lower the effect of them at the same time. (2) The distance between samples from different categories is much wider than before by extending the $\epsilon$-dragging method to a semi-supervised paradigm. (3) Automatically discover the EEG emotional activation mode by adaptively measuring the contribution of sample features through feature weights. In the three cross-session emotion recognition tasks, the average accuracy of the RSRRW model is 81.51%, which can be seen in the experimental results on the SEED-IV dataset. In addition, with the support of the Friedman test and Nemenyi test, the classification of RSRRW model is much more accurate than that of other models.

Keywords: electroencephalogram (EEG); robustness; semi-supervised classification; emotion recognition; emotional activation pattern mining

1. Introduction

As a complex psychological state, emotion plays a key role in human cognition, including rational decision-making, perception, interpersonal communications and human intelligence [1]. Therefore, emotion recognition has attracted the attention of researchers from various disciplines. Usually, researchers investigate emotion recognition from the data sources of language, body movements, speech, and facial expressions. However, these patterns have certain drawbacks. (1) When subjects deliberately disguise their emotions, the performance of the method may be significantly affected by the deceptive data collected from the subjects based on the above data model. (2) The previous mode is impractical for people with physical disabilities (deafness, aphasia and etc.) Therefore, we need a more objective mode of emotional recognition. Firstly, considering that emotion is hard to measure as the result of spontaneity, thus it can be observed by the accompanying physiological reactions in the central nervous system and periphery [2,3]. Secondly, EEG is a signal from the central nervous system, which has several advantages, such as large amounts of information, simple operation and low-cost [4]. So, together with the rapid development of non-stationary signal processing and analysis techniques, EEG-based emotion recognition has become a research hotspot [5].
In recent years, emotion recognition based on EEG signals has aroused the discussion of researchers. Li et al. [6] summarized preceding research results, including traditional machine learning works [7], deep learning based works [8], transfer learning based works [9,10] and ensemble learning based works [11]. Considering that it is weak and unstable, noise can interfere with the EEG data during the data acquisition process, therefore, the quality of different EEG samples may vary accordingly. In addition, due to the multi-channel and multi-rhythm properties of EEG data, features from different frequency bands and channels should have different correlations with emotional effects. However, existing studies either ignore both the issues of sample and the importance of feature or only consider one of them.

Therefore, we establish the Retargeted Semi-supervised Regression with Robust Weights model (RSRRW). Firstly, to improve the robustness of the model, we introduce the probability weight factor, which has clear physical meaning. When the weight is 0, it means that the corresponding sample is a noise point, otherwise, it is a normal point. If a certain sample deviates from the entirety too much, the RSRRW model will choose to skip absolutely over the error as well as its contribution to noise points, thus ensuring that the model does not skew toward these outliers. In order to maintain the good discriminative ability of the model, the \( \epsilon \)-dragging method is introduced to widen the gap between different classes. Additionally, feature weight factor is taken into consideration for the purpose of exploring the extent to which different dimensions of feature weight contribute to the emotion recognition task.

Consequently, the main contributions of this work to available data can be summed up.

- A new factor, probability weight, is added to each sample in the model. With the help of probability weight, the model can differentiate noise in the samples and remove the negative effects of noise. At the same time, the value of this variable also has a clear physical meaning.
- Innovatively apply the \( \epsilon \)-dragging technique to the semi-supervised paradigm. It is aimed at estimating the direction matrix by gradually optimizing the label of the unlabeled samples during the learning process, which effectively increases the margin between classes.
- Compared with similar models, RSRRW has higher recognition accuracy. With the help of the feature weight factor, RSRRW can discover EEG activation patterns relating to the task, and resultantly determine comparatively significant frequency bands and predominant leads of the EEG under the current task.

**Notations.** In this paper, the frequency bands of EEG will be denoted by *Delta*, *Theta*, *Alpha*, *Beta*, and *Gamma*. Greek letters, such as \( \eta, \lambda, \gamma \), represent the model parameters (e.g., \( \alpha \) represents the confidence level). Matrices and vectors are respectively identified by boldface uppercase and lowercase letters, respectively. The \( \ell_{2,1} \)-norm of matrix \( W \in \mathbb{R}^{d \times c} \) is defined as \( \| W \|_{2,1} = \sum_{d=1}^{d} \sqrt{\sum_{j=1}^{c} w_{ij}^2} = \sum_{d=1}^{d} \| w^d \|_2 \). \( w^d \) is the \( d \)-th row of \( W \) and \( w_j \) is the \( j \)-th column of \( W \). The \( \odot \) is a Hadamard product operator of matrices.

The main body of this article is organized as follows. Section 2 briefly reviews the background about EEG-based emotion recognition and some related techniques. In Section 3, we introduce the formulation and optimization of the RSRRW model in detail. In Section 4, we conduct experimental studies to illustrate the performance of RSRRW. Section 5 concludes the whole paper.

**2. Related Work**

**2.1. EEG-Based Emotion Recognition**

A typical EEG-based emotion recognition process includes three stages, data preprocessing, feature extraction and model training.

Since the EEG signals are easily disturbed during acquisition, in order to provide quality data, preprocessing is necessary in the experiment. EEG data preprocessing usually includes sampling and artifact removal. In general, the sampling frequency of EEG is...
usually 128∼1024 Hz [12]. The higher the sample rate, the more detail the EEG can capture in the data, but it also produces more noise. Noises, in other words, the causes of interference, mainly contain electrooculography, electrocardiogram, electromyography, vascular waves and so on. Common processing methods include principal components analysis, independent component analysis, and various filtering algorithms.

Feature extraction occupies an important position in the EEG emotion recognition task, and the way of feature extraction will directly have a great impact on the performance of emotion classification. EEG feature extraction methods can be classified into four categories, namely feature extraction based on time domain, feature extraction based on frequency domain, feature extraction based on time-frequency domain and feature extraction based on spatial domain. To be specific, the feature extraction methods based on time domain include histogram analysis method [13], Hjorth parameter [14], event-related potential [15] and so on. Time domain-based analysis methods often obtain information by analyzing the geometric features of EEG signals, and the rate of information loss is very low. The frequency domain-based method mainly performs feature extraction by transforming the EEG signal from the time domain to the frequency domain, and dividing the acquired spectrum into multiple sub-bands. Its general methods include power spectral density [16], higher-order crossover [17], differential entropy (DE) [18] and higher-order spectrum [19] and so on. EEG feature extraction based on time-frequency includes short-time fourier transform [20], wavelet transform [21], wavelet packet transform [22], Hilbert–Huang transform [23] and more. Feature extraction based on spatial domain is mainly to explore the correlation features between EEG signals at different locations. The methods include common spatial patterns [24], sub-band common spatial pattern [25], rational asymmetry [26], etc. Some review studies summarized the feature extraction methods in EEG-based emotion recognition [23,27].

As for the model training process, a lot of efforts were made in the past decades. Roughly, existing models can be categorized into linear and nonlinear models. In recent studies, deep learning has attracted much attention due to its powerful nonlinear learning ability. Thammasan et al. utilized the deep belief network (DBN) to classify the EEG emotions based on the handcrafted features extracted from EEG (fractal dimension, power spectral density, etc.) [28]. Li et al. used the differential entropy features of EEG to build 2D-images, and used CNN network to complete the task of emotion recognition, and achieved good results [29]. On account of the previous study, some deep learning models further unify feature extraction and recognition to form an end-to-end learning model [30,31]. Although the above deep learning models have achieved relatively good results in EEG emotion recognition tasks, most of their models stay at the stage of black-box training modes, the outcomes are poorly interpretable, and the underlying mechanism is relatively abstract [32]. Furthermore, to improve the interpretability of the model, Peng et al. proposed a unified framework named GFIL for discovering feature importance in EEG emotion recognition tasks [33]. In [34], GFIL was extended to a semi-supervised graph learning framework. Recent advances in EEG-based emotion recognition can be found in [6].

2.2. Related Techniques
2.2.1. Emotion Models

According to the existing research, affective models can be roughly divided into two categories, one is the discrete affective state model, and the other is the dimension-space affective model. In the discrete model, the emotional space covers a limited number of basic emotions. The six basic emotions are generally recognized as anger, disgust, surprise, sadness, happiness and fear, based on which other emotions are mostly considered as their combinations [35,36]. In typical dimensional space models, emotion is considered to be distributed in two-dimension or three-dimension space, and the attributes of different dimensions provide evidence for locating positions of emotions. Among these spacial models, it is common for researchers to employ valence-arousal model (VA) [37] and
valence-arousal-dominance model (VAD) [38]. In VA, the valence axis is used to measure
the positive-negative degree of emotions while the arousal axis is used for measuring the
intensity of emotions, as exemplified in Figure 1a. Compared with VA, VAD in Figure 1b
adds the dominance axis to measure the degree to which emotions can be controlled.

![Figure 1](image)

**Figure 1.** Emotion models VA (a) and VAD (b). (a) The VA model consists of the dimensions
named valence and arousal, (b) The VAD model consists of the dimensions named valence, arousal
and dominance.

2.2.2. Rescaled Least Squares Regression

Given a data matrix \( \mathbf{X} = [x_1, x_2, \cdots, x_n] \in \mathbb{R}^{d \times n} \) where \( d \) is the feature dimension, \( n \) is
the number of samples. We use \( \mathbf{Y} = [y_1, y_2, \cdots, y_n] \in \mathbb{R}^{c \times n} \) to denote the binary indicator
matrix. Specifically, if sample \( x_i \) belongs to the \( j \)-th class and \( y_i \) is the \( i \)-th column of \( \mathbf{Y} \), then
the \( j \)-th element of \( y_i \) is 1 and all the others of \( y_i \) are 0. Its mathematical form is

\[
y_{ij} = \begin{cases} 1, & x_i \text{ belongs to the } i \text{-th emotion;} \\ 0, & \text{otherwise.} \end{cases}
\]

Least squares regression is a typical statistical analysis technique, which has high va-

lidity for data analysis. It has been favored in previous researches, and its semi-supervised
general form is as follows

\[
\min_{\mathbf{W}, \mathbf{b}, \mathbf{Y}_u} \| \mathbf{W}^T \mathbf{X} + \mathbf{b}^T - \mathbf{Y} \|_2^2 + \gamma \| \mathbf{W} \|_F^2,
\]

\[\text{s.t. } \mathbf{Y}_u \geq 0, \mathbf{Y}_u^T \mathbf{1} = 1,\]

where \( \mathbf{Y}_u \) are unlabeled samples.

In the EEG emotion recognition task, considering that EEG data has the natural
characteristic of multi-channel, the importance of data from different channels can be of
difference from each other in the classification task, however, objective function (2) fail
to meet the requirements of differentiated feature contribution. Thus Chen et al. [39]
conducted the following model,

\[
\min_{\mathbf{W}, \mathbf{b}, \mathbf{Y}_u} \| \mathbf{W}^T \mathbf{X} + \mathbf{b}^T - \mathbf{Y} \|_2^2 + \lambda \| \mathbf{W} \|_{2,1}^2,
\]

\[\text{s.t. } \mathbf{Y}_u \geq 0, \mathbf{Y}_u^T \mathbf{1} = 1, \Theta = \text{diag}(\sqrt{\mathbf{\theta}}), \theta \geq 0, \theta^T \mathbf{1} = 1,\]

where \( \Theta \) is a diagonal matrix, \( \Theta_{jj} = \sqrt{\theta_j} \), \( \theta_j \) describes the significance of the \( j \)-th feature.
Performing a simple transformation of the objective function (3), we obtain

\[
\min_{\mathbf{W}, \mathbf{b}, \mathbf{Y}_u} \| \tilde{\mathbf{W}}^T \mathbf{X} + \mathbf{b}^T - \mathbf{Y} \|_2^2 + \lambda \| \tilde{\mathbf{W}} \|_{2,1}^2,
\]

\[\text{s.t. } \mathbf{Y}_u \geq 0, \mathbf{Y}_u^T \mathbf{1} = 1, \tilde{\mathbf{W}} = \Theta \mathbf{W},\]

where \( \tilde{\mathbf{W}} = \Theta \mathbf{W} \).
2.2.3. Discriminative Least Squares Regression

The traditional LSR method is usually employed for data fitting. When it is used for data classification, the regression target \( y_i \) of sample \( x_i \) will be represented as a discrete value (such as +1 for the first category, −1 for the second category) or represented by one-hot encoding. In classification, data points from different classes are expected to be farther apart; however, traditional LSR cannot achieve the goal.

Therefore, Xiang et al. proposed the Discriminative Least Squares Regression (DLSR) method [40]. By introducing the \( \varepsilon \)-dragging technology, the class label information is embedded in the LSR framework and then the regression targets from different categories of samples move to the opposite direction, which can expand the distance between sample points from different classes. Specifically, for a positive slack variable \( \varepsilon_{ij} \), if \( y_{ij} = 1 \), the output will become \( 1 + \varepsilon_{ij} \); if \( y_{ij} = 0 \), the output will become \( -\varepsilon_{ij} \). This method can be naturally extended to multi-class classification task. The mathematical form of DLSR is,

\[
\min_{w,b,M} \|X^T W + 1b^T - Y - B \odot M\|_F^2 + \lambda \|W\|_F^2,
\]

\[\text{s.t. } M \geq 0.\] (5)

The matrix \( B \) is defined as,

\[
B_{ij} = \begin{cases} 
+1, & y_{ij} = 1, \\
-1, & y_{ij} = 0,
\end{cases}
\] (6)

which is used to determine the target dragging direction. Variable \( M \) is the traction matrix and its elements are iteratively updated in model learning.

In the optimization process, \( \varepsilon \)-dragging constructs the loss function into two types of piecewise function, for the purpose of increasing the sample distance between different categories. Below is an example to explain it. Considering that we have a four-class classification problem, and the sample label vectors are in one-hot encoding. It is advisable to set its label of a sample from the first class as \( [1, 0, 0, 0]^T \), and set the predicted value of the model under each category as \( W^T x + b = [\hat{y}_1, \hat{y}_2, \hat{y}_3, \hat{y}_4]^T \). For its category (the first category, #1), if its predicted value \( \hat{y}_1 > 1 \), \( \varepsilon \) will be updated as \( \varepsilon_1 - 1 \), so that the error of the sample under the category is 0. For the other categories (the second, third, fourth category, #2 #3 #4), if its predicted value is \( \hat{y}_j < 0 \) (\( j \neq 1 \)), \( \varepsilon \) will be updated as \( -\hat{y}_j \). Accordingly, it makes the error of this sample under other categories be 0. The detailed mathematical principle will be explained in the third step of optimization process in Section 3.2.

Table 1 graphically illustrates the purpose of the \( \varepsilon \)-dragging method. For a four-class dataset with eight sample points, the regression targets of the models (2) and (5) can be found respectively in the third and fourth columns in the Table 1.

<table>
<thead>
<tr>
<th>Sample</th>
<th>Class</th>
<th>LSR Targets</th>
<th>DLSR Targets</th>
<th>Constraint</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_1 )</td>
<td>1</td>
<td>([1, 0, 0, 0]^T)</td>
<td>([1 + \varepsilon_{11}, -\varepsilon_{12}, -\varepsilon_{13}, -\varepsilon_{14}]^T)</td>
<td>( \varepsilon_{11}, \varepsilon_{12}, \varepsilon_{13}, \varepsilon_{14} \geq 0 )</td>
</tr>
<tr>
<td>( x_2 )</td>
<td>1</td>
<td>([1, 0, 0, 0]^T)</td>
<td>([1 + \varepsilon_{21}, -\varepsilon_{22}, -\varepsilon_{23}, -\varepsilon_{24}]^T)</td>
<td>( \varepsilon_{21}, \varepsilon_{22}, \varepsilon_{23}, \varepsilon_{24} \geq 0 )</td>
</tr>
<tr>
<td>( x_3 )</td>
<td>2</td>
<td>([0, 1, 0, 0]^T)</td>
<td>([-\varepsilon_{31}, 1 + \varepsilon_{32}, -\varepsilon_{33}, -\varepsilon_{34}]^T)</td>
<td>( \varepsilon_{31}, \varepsilon_{32}, \varepsilon_{33}, \varepsilon_{34} \geq 0 )</td>
</tr>
<tr>
<td>( x_4 )</td>
<td>2</td>
<td>([0, 1, 0, 0]^T)</td>
<td>([-\varepsilon_{41}, 1 + \varepsilon_{42}, -\varepsilon_{43}, -\varepsilon_{44}]^T)</td>
<td>( \varepsilon_{41}, \varepsilon_{42}, \varepsilon_{43}, \varepsilon_{44} \geq 0 )</td>
</tr>
<tr>
<td>( x_5 )</td>
<td>3</td>
<td>([0, 0, 1, 0]^T)</td>
<td>([-\varepsilon_{51}, -\varepsilon_{52}, 1 + \varepsilon_{53}, -\varepsilon_{54}]^T)</td>
<td>( \varepsilon_{51}, \varepsilon_{52}, \varepsilon_{53}, \varepsilon_{54} \geq 0 )</td>
</tr>
<tr>
<td>( x_6 )</td>
<td>3</td>
<td>([0, 0, 1, 0]^T)</td>
<td>([-\varepsilon_{61}, -\varepsilon_{62}, 1 + \varepsilon_{63}, -\varepsilon_{64}]^T)</td>
<td>( \varepsilon_{61}, \varepsilon_{62}, \varepsilon_{63}, \varepsilon_{64} \geq 0 )</td>
</tr>
<tr>
<td>( x_7 )</td>
<td>4</td>
<td>([0, 0, 0, 1]^T)</td>
<td>([-\varepsilon_{71}, -\varepsilon_{72}, -\varepsilon_{73}, 1 + \varepsilon_{74}]^T)</td>
<td>( \varepsilon_{71}, \varepsilon_{72}, \varepsilon_{73}, \varepsilon_{74} \geq 0 )</td>
</tr>
<tr>
<td>( x_8 )</td>
<td>4</td>
<td>([0, 0, 0, 1]^T)</td>
<td>([-\varepsilon_{81}, -\varepsilon_{82}, -\varepsilon_{83}, 1 + \varepsilon_{84}]^T)</td>
<td>( \varepsilon_{81}, \varepsilon_{82}, \varepsilon_{83}, \varepsilon_{84} \geq 0 )</td>
</tr>
</tbody>
</table>

3. Method

In this section, we present the model formulation of RSRRW firstly and then its detailed optimization procedure.
In EEG-based semi-supervised emotion recognition problem, we use the matrix $X = [x_1, x_2, \cdots, x_n] \in \mathbb{R}^{d \times n}$ to represent the EEG data. $X$ consists of two subsets, one is labeled samples $X_l \in \mathbb{R}^{d \times n_l}$, corresponding to the labels $Y_l \in \mathbb{R}^{c \times n_l}$. The other one is unlabeled samples $X_u \in \mathbb{R}^{d \times n_u}$, correspondingly to the labels $Y_u \in \mathbb{R}^{c \times n_u}$. $d$ is the feature dimension of EEG samples, $c$ is the number of emotion categories, $n_l$ is the number of labeled samples, $n_u$ is the number of unlabeled samples and $n = n_l + n_u$.

The target of this algorithm is to accurately predict $Y_u$ using the given data $X = [X_l, X_u]$ and $Y_l$.

### 3.1. Model Formulation

The RSRRW model framework is shown in Figure 2, in which the model part mainly includes four components, namely sample weight learning, feature weight learning, $\epsilon$-dragging process, coefficient matrix learning. The functions of each part are summarized as follows. (1) Based on sample weight learning, add a probability weight to each sample point in the data. When the sample is a normal point, its probability weight is 1, otherwise, its probability weight is 0. (2) Based on feature weight learning, differentiate the contributions of features. (3) In the $\epsilon$-dragging process, the regression targets of different categories are forced to move in opposite directions, expanding the distance between categories. (4) Coefficient matrix learning, using the least squares method to learn $W$ and $b$.

![Figure 2. The general framework of RSRRW model.](image-url)

The $\epsilon$-dragging strategy was proposed in [40], and its mathematical form is shown in

$$
\min_{W, b, M} \|V^T X + b 1^T - Y - B \otimes M\|_F^2 + \lambda \|V\|_F^2,
\text{s.t. } M \geq 0,
$$

(7)

where $V$ is original coefficient matrix.

Considering that EEG data is a multi-channel time series data, features from different frequency bands and leads have different contributions to specific tasks, and the objective function (7) cannot distinguish the importance of different features, which is defective. Therefore, we introduce the sample feature weight representation matrix $\Theta$ to describe the importance of sample features, and construct the following objective function,

$$
\min_{W, b, M} \|V^T \Theta X + b 1^T - Y - B \otimes M\|_F^2 + \lambda \|V\|_F^2,
\text{s.t. } M \geq 0, \Theta = \text{diag}(\sqrt{\theta}), \theta \geq 0, \theta^T 1 = 1.
$$

(8)
Considering the weak and unstable characteristics of EEG data, the quality of different samples varies greatly. In order to improve the robustness of the model to samples, we introduce the sample probability weight \( s \), so that the model can automatically filter outliers during the training process. The optimized objective function is

\[
\min \sum_{i=1}^{n} s_i \left\| V^T \Theta x_i + b - y_i - d_i \otimes m_i \right\|^2 + \lambda \| V \|^2_F, \tag{9}
\]

s.t. \( s^T 1 = k, 0 \leq s_i \leq 1, \Theta = \text{diag}(\sqrt{\theta}), \theta \geq 0, \theta^T 1 = 1, m_i \geq 0. \)

In model (9), \( s \) is the sample weight, \( k \) is the number of normal points in the sample, \( d_i \) is the \( i \)-th column of \( B \). However, model (9) can only be used in a supervised setting, here we extend it to a semi-supervised setting,

\[
\min \sum_{i=1}^{n} s_i \left\| V^T \Theta x_i + b - y_i - (2y_i - 1) \otimes m_i \right\|^2 + \lambda \| V \|^2_F, \tag{10}
\]

s.t. \( s^T 1 = k, 0 \leq s_i \leq 1, \Theta = \text{diag}(\sqrt{\theta}), \theta \geq 0, \theta^T 1 = 1, \)
\( Y = [Y_l; Y_u], Y_u \geq 0, Y_u 1 = 1, m_i \geq 0. \)

By setting \( W = \Theta V \), we have \( V = \Theta^{-1} W \) and then the above objective function can be rewritten as

\[
\min \sum_{i=1}^{n} s_i \left\| W^T x_i + b - y_i - (2y_i - 1) \otimes m_i \right\|^2 + \lambda \| \Theta^{-1} W \|^2_F, \tag{11}
\]

s.t. \( s^T 1 = k, 0 \leq s_i \leq 1, \Theta = \text{diag}(\sqrt{\theta}), \theta \geq 0, \theta^T 1 = 1, \)
\( Y = [Y_l; Y_u], Y_u \geq 0, Y_u 1 = 1, m_i \geq 0. \)

When \( s, W, b, M \) are fixed, because \( \Theta = \text{diag}(\sqrt{\theta}) \) and \( \theta^T 1 = 1 \), the second term of objective function (11) can be rewritten as

\[
\min_{\theta \geq 0, \theta^T 1 = 1} \sum_{j=1}^{l} \frac{\| w_j \|_2^2}{\theta_j}. \tag{12}
\]

According to the Lagrange multiplier method, we write the Lagrangian function of (12) about \( \theta_j \) and set its derivative to \( \theta_j \) to 0, then we can get

\[
\theta_j = \frac{\| w_j \|_2^2}{\sum_{j=1}^{l} \| w_j \|_2^2}. \tag{13}
\]

With the above solution to \( \theta \), objective function (12) is equivalent to

\[
\min_{\theta \geq 0, \theta^T 1 = 1} \| w \|_{2,1}^2. \tag{14}
\]

Now model (11) can be rewritten as

\[
\min \sum_{i=1}^{n} s_i \left\| W^T x_i + b - y_i - (2y_i - 1) \otimes m_i \right\|^2 + \lambda \| W \|_{2,1}^2, \tag{15}
\]

s.t. \( s^T 1 = k, 0 \leq s_i \leq 1, Y = [Y_l; Y_u], Y_u \geq 0, Y_u 1 = 1, m_i \geq 0. \)

3.2. Model Optimization

There are five variables in model (15) that need to be optimized, namely \( W, b, Y_u, s, M \). For this problem, we designed a joint iterative optimization algorithm, which divides
the above problem into four sub-problems to solve them separately and performs multiple iterations until convergence.

- Update \( W, b \) by fixing \( Y_u, s \) and \( M \).

Performing simple variable substitution on objective function (15), we can get

\[
\min_{W,b} \sum_{i=1}^{n} s_i \mu(e_i) + \lambda \|W\|_{2,1}^2 \tag{16}
\]

where \( e_i = \|W^T x_i + b - z_i\|_2^2 \), \( z_i \) is the \( i \)-th column of \( Z \), \( Z = Y + (2Y - 11^T) \odot M \) and \( \mu \) is the function of \( e \), \( \mu(e) = e^2 \).

We introduce an effective algorithm [41] to solve objective function (16). The general problem is as follows

\[
\min_{x \in C} \sum_i h_i(g_i(x)) + f(x). \tag{17}
\]

The form of \( x \) and \( g(x) \) could be scalar, vector or matrix. Algorithm 1 describes the detailed procedures. By comparison of (16) and (17), we can find \( \mu(e), e_i(W, b), \lambda \|W\|_{2,1}^2 \) in problem (17) as \( h(x), g_i(x), \) and \( f(x) \) in (17), respectively.

**Algorithm 1 Solution of (17).**

**Input:** Initialize \( x \in C \)

**Output:** The optimal \( x \)

1. while not converged do
2. Calculated \( d_i = h'_i(g_i(x)) \);
3. Solve the following minimization problem \( \min_{x \in C} \sum_i \text{Tr}(d_i^T g_i(x)) + f(x) \);
4. end while

Firstly, we need to calculate \( d_i \). The formulation of \( d_i \) is

\[
d_i = \mu'(e_i) = \frac{1}{2 \|W^T x_i + b - z_i\|_2^2}. \tag{18}
\]

Secondly, we need to solve

\[
\min_{W,b} \sum_{i=1}^{n} s_i \|W^T x_i + b - z_i\|_2^2 + \lambda \|W\|_{2,1}^2. \tag{19}
\]

To facilitate subsequent calculations, we rewrite (19) into matrix form

\[
J = \text{Tr}\left( (W^T X + b1^T - Z) \Lambda (W^T X + b1^T - Z)^T \right) + \lambda \|W\|_{2,1}^2, \tag{20}
\]

where \( \Lambda = SD, S = \text{diag}(s_1, s_2, \ldots, s_n), D = \text{diag}(d_1, d_2, \ldots, d_n) \). Considering that (19) is an unconstrained optimization problem, we use gradient descent to solve it. Take the partial derivative of \( J \) with respect to \( b \), we have

\[
\frac{\partial J}{\partial b} = 2(W^T X + b1^T - Z) \Lambda 1. \tag{21}
\]

Let (21) take 0, we can get the optimization formula of \( b \) as

\[
b = \frac{ZA1 - W^T X A1}{1^T A1}. \tag{22}
\]
Replacing \( b \) in (20) with the result obtained in (22) and simplifying it, we can get
\[
J = \operatorname{Tr} \left( (W^T X - Z) K (W^T X - Z)^T \right) + \lambda \sum_{i=1}^{d} \sqrt{\|w^i\|_2^2} + \delta. \tag{23}
\]
where \( K = (\mathbf{I} - \Lambda_{11}^T \Lambda_{11}) \Lambda (\mathbf{I} - \Lambda_{11}^T \Lambda_{11})^T \) and \( K = K^T \). Similarly, (23) is also an unconstrained optimization problem, so we take the partial derivative of (23) in terms of \( W \), then we can get
\[
\frac{\partial J}{\partial W} = 2 (XKX^T W - XKZ^T + \lambda Q W), \tag{24}
\]
where \( Q \in \mathbb{R}^{d \times d} \) is a diagonal matrix whose \( i \)-th diagonal elements is
\[
q_{ii} = \frac{\sum_{j=1}^{d} \sqrt{\|w^j\|_2^2 + \delta}}{\sqrt{\|w^i\|_2^2 + \delta}}, \tag{25}
\]
and \( \delta \) is a fixed minimal constant value. By making the partial derivative value 0, the expression of \( W \) is obtained
\[
W = (XKX^T + \lambda Q)^{-1}XKZ^T. \tag{26}
\]

- Update \( s \) by fixing \( Y_u, W, b \) and \( M \).

The corresponding objective function in terms of variable \( s \) is
\[
\min_s \sum_{i=1}^{n} s_i e_i \text{ s.t. } s_i \geq 0, s^T 1 = k. \tag{27}
\]
where \( e_i \) measures the approximation error on sample \( x_i \) by the \( \ell_2 \)-norm. It is interesting that parameter \( k \) and the variable \( s \) are closely related in model (27). To be specific, the value of \( k \) represents the number of elements of one in \( s \), corresponding to the number of normal samples involved in model training. By calculating and ranking the loss \( e_i \) for each sample, the optimal \( s \) can be obtained, in which the weights of the first \( k \) samples with the smallest errors are all set as one and the remaining values are zeros. This regularity is depicted by the following Theorem 1.

**Theorem 1.** The optimal \( s \) in problem (27) is a binary vector, in which the corresponding weights of the first \( k \) samples with the smallest errors are one and the others are zero.

**Proof 1.** Suppose there is another weight vector \( s' \) that satisfies
\[
\sum_{j=1}^{k} s'_j + \sum_{j=k+1}^{n} s'_j = k, \tag{28}
\]
\[
\text{s.t. } \sum_{j=1}^{n} s'_j \mu(e^{(j)}) \leq \sum_{j=1}^{n} s'_j \mu(e^{(j)}). \tag{29}
\]
Firstly, we sort each sample in ascending order of error, i.e.,
\[
\mu(e^{(1)}) \leq \mu(e^{(2)}) \leq \cdots \leq \mu(e^{(j)}) \leq \cdots \leq \mu(e^{(n)}). \tag{30}
\]
After a simple split on (29), we get
\[
\sum_{j=1}^{k} s'_j \mu(e^{(j)}) + \sum_{j=k+1}^{n} s'_j \mu(e^{(j)}) \leq \sum_{j=1}^{k} \mu(e^{(j)}). \tag{31}
\]
By moving the first term to the left of the inequality sign to the right, we have

\[ \sum_{j=k+1}^{n} s'_j \mu(e^{(j)}) \leq \sum_{j=1}^{k} (1 - s'_j) \mu(e^{(j)}). \]  

(32)

According to (30), we can get

\[ \sum_{j=k+1}^{n} s'_j \mu(e^{(j)}) \geq (1 - s'_j) \mu(e^{(k+1)}). \]  

(33)

After combining (32) and formula (33), we can get

\[ \sum_{j=1}^{k} (1 - s'_j) \mu(e^{(k+1)}) \leq \sum_{j=1}^{k} (1 - s'_j) \mu(e^{(j)}). \]  

(34)

When inequality signs in (30) are not taken equal at the same time, Equation (34) obviously does not hold, so the assumption is logically impossible. Therefore, the optimal \( s \) in problem (27) is a binary vector, in which the corresponding weights of the first \( k \) samples with the smallest errors are one and the others are zero.

- Update \( M \) by fixing \( Y_u, W, b \) and \( s \).

From problem (15), the objective function in terms of variable \( M \) is

\[
\min_{M \geq 0} \| W^T X + b^T - Y - (2Y - 11^T) \odot M \|_2^2 = \| P - B \odot M \|_2^2,
\]

(35)

where \( P \triangleq W^T X + b^T - Y \) and \( B \triangleq 2Y - 11^T \). It is easily verified that matrix \( B \) is similar to the one defined in [40] in supervised learning. Considering that in the optimization process of (35), the optimization process of each element is independent of each other, therefore, for an element in the matrix, we can convert (35) into the following form

\[
\min_{M_{ij} \geq 0} (P_{ij} - B_{ij} M_{ij})^2.
\]

(36)

Obviously, the solution to the above optimization problem is

\[ M_{ij} = \max\left(\frac{P_{ij}}{B_{ij}}, 0\right). \]

(37)

Then, the solution to (35) is

\[ M = \max(P ./ B, 0). \]

(38)

In the formula (38), we find that the \( \epsilon \)-dragging method essentially encourages the predicted value \( \hat{g} \) to migrate towards \( 1 + \epsilon_{ij} \) or \( -\epsilon_{ij} \) by constructing the loss function as a two-class piecewise function. Below we will use an example to detail the updating method during the optimization process. One point to note is that in the algorithm, we have implemented the \( \epsilon \)-dragging method through matrix \( M \) and matrix \( B \), and the elements in \( M \) can be regarded as \( \epsilon_{ij} \). To maintain the consistence of symbols, the element is represented as \( M_{ij} \).

For the sample \( x_i \), we set the emotional label as \( y_i = [1, 0, 0, 0]^T \). In the model (35), its predicted value is \( \hat{y}_i = W^T x_i + b = [\hat{y}_1, \hat{y}_2, \hat{y}_3, \hat{y}_4]^T \). In order to distinguish it from \( b \)
in the LSR, we set the $i$-th column of the direction matrix $B$ to be $B = [\tilde{b}_1, \tilde{b}_2, \cdots, \tilde{b}_n]$. The direction vector of $x_i$ is $\tilde{b}_i = [1, -1, -1, -1]^T$. The error generated by the sample is,

$$\text{error}_i^2 = \|\tilde{y}_i - y_i - \tilde{b}_i \odot m_i\|^2_2$$

(39)

Based on (39), the $\text{error}_i^2$ consists of four items, which can be divided into two categories. One is the error generated by the predicted value of the category to which the sample belongs, corresponding to the first item in (39); the other is the error generated by the predicted value of the category to which the sample belongs, corresponding to the second, third, and fourth items in (39). We can easily conclude the following findings.

There are two steps in the calculation of the error of the category of the sample (that is, the first item in (39)). Firstly, $M_{1,i}$ can be represented by the following piecewise function,

$$M_{1,i} = \begin{cases} \hat{y}_1 - 1, & \hat{y}_1 \geq 1 \\ 0, & \hat{y}_1 < 1 \end{cases}$$

(40)

Then the corresponding error is calculated as,

$$\text{error}_{1,i}^2 = \begin{cases} 0, & (\hat{y}_1 - 1)^2 \\ (\hat{y}_1 - 1)^2, & \hat{y}_1 \geq 1 \\ (\hat{y}_1 - 1)^2, & \hat{y}_1 < 1 \end{cases}$$

(41)

The formula (41) indicates that when the predicted value $\hat{y}_1$ of the category to which the sample belongs is greater than 1, the error generated by $\hat{y}_1$ is 0. However, in the traditional LSR, an error will still be generated at this time, and its value is $(\hat{y}_1 - 1)^2$. The $\epsilon$-dragging method uses the no-negative slack variable $M_{1,i}$, offsetting the corresponding error. Thereby it encourages the predicted value $\hat{y}_1$ of the category to move in the direction $1 + \epsilon$.

Similarly, for the samples of the other categories (i.e., items 2, 3, and 4 in (39)), the analysis is presented below. The variable $M_{c_{j},i} (c_j \neq 1)$ can be represented as

$$M_{c_{j},i} = \begin{cases} 0, & \hat{y}_{c_{j}} \geq 0, \\ -\hat{y}_{c_{j}}, & \hat{y}_{c_{j}} < 0 \end{cases}$$

(42)

Secondly, the corresponding error is calculated as,

$$\text{error}_{c_{j},i}^2 = \begin{cases} 0, & \hat{y}_{c_{j}}^2 \\ \hat{y}_{c_{j}}^2, & \hat{y}_{c_{j}} \geq 0, \\ \hat{y}_{c_{j}}^2, & \hat{y}_{c_{j}} < 0 \end{cases}$$

(43)

The formula (43) indicates that when the predicted value $\hat{y}_{c_{j}}$ of other categories is less than 0, the error generated by $\hat{y}_{c_{j}}$ is 0. However, in LSR, an error will still be generated at this time, and its value is $(\hat{y}_{c_{j}} - 1)^2$. The $\epsilon$-dragging method uses the no-negative slack variable $M_{c_{j},i}$ to offset the corresponding error. Thereby it encourages the predicted value of other category $\hat{y}_{c_{j}}$ to move in the direction $-\epsilon$.

- Update $Y_u$ by fixing $s, W, b$ and $M$.

Rows in $Y_u$ are independent of each other, and then we can optimize $Y_u$ in a column-wise manner. To be specific, for $i = l + 1, l + 2, \cdots, l + u$, the corresponding objective function in terms of $y_i$ is

$$\min_{y_i \geq 0, y_i^T 1 = 1} \left\| W^T x_i + b + m_i - y_i \odot (1 + 2m_i) \right\|^2_2 \triangleq \|a_i - y_i\|^2_2$$

(44)
where \( a \triangleq (W^T x_i + b + m_i) / (1 + 2m_i) \). To simplify the derivation, we optimize the equivalent form of problem (44) as

\[
\min_{y_i \geq 0, y_i^T 1 = 1} \frac{1}{2} \|a_i - y_i\|_2^2.
\] (45)

Problem (45) can be solved by the Lagrange multiplier method combined with KKT condition. The corresponding Lagrangian function is

\[
L = \frac{1}{2} \|y_i - a_i\|_2^2 - \eta (y_i^T 1 - 1) - \beta^T y_i,
\] (46)

where \( \eta \) and \( \beta \in \mathbb{R}^c \) are the Lagrangian multipliers respectively in scalar and vector forms. Assume that \( \tilde{y}_i \) is the optimal solution and the associated optimal Lagrangian multipliers are \( \tilde{\eta} \) and \( \tilde{\beta} \). According to the KKT condition, we have

\[
\begin{align*}
\tilde{y}_j &\geq 0, \\
\tilde{\beta}_j &\geq 0, \\
\tilde{y}_j \tilde{\beta}_j &= 0, \\
\tilde{y}_j - a_j - \tilde{\eta} - \tilde{\beta}_j &= 0,
\end{align*}
\] (47)

for each \( j \in \{1, 2, \cdots, c\} \). The last expression in (47) can be written in vector form as

\[
y - a - \tilde{\eta} 1 - \tilde{\beta} = 0.
\] (48)

Because \( y^T 1 = 1 \), the parameter \( a \) in (48) can be written as

\[
\tilde{\eta} = \frac{1 - y^T a - 1^T \tilde{\beta}}{c}.
\] (49)

Substituting (49) into (48), we have

\[
\tilde{y} = a - \frac{11^T}{c} a + \frac{1}{c} 1 - \frac{1^T \tilde{\beta}}{c} 1 + \tilde{\beta}.
\] (50)

Denote \( \overline{\beta} = \frac{1^T \tilde{\beta}}{c} \) and \( g = a - \frac{11^T}{c} + \frac{1}{c} 1 \), (50) can be rewritten as

\[
\tilde{y} = g_j + \tilde{\beta}_j - \overline{\beta} 1.
\] (51)

Therefore we have

\[
\tilde{y}_j = g_j + \tilde{\beta}_j - \overline{\beta}.
\] (52)

Through (47) and (52), we can get \( g_j + \tilde{\beta}_j - \overline{\beta} = (g_j - \overline{\beta})_+ \), where

\[
(f(\cdot))_+ = \begin{cases} 0 & f(\cdot) < 0, \\ f(\cdot) & \text{otherwise.} \end{cases}
\] (53)

Therefore, (52) can be written as

\[
\tilde{y}_j = (g_j - \overline{\beta})_+.
\] (54)
At this time, if the optimal $\tilde{\beta}$ can be determined, according to (54), the optimal $\tilde{y}$ can also be determined. Similar to (54), (52) can be rewritten as $\tilde{\beta}_j = \tilde{y}_j + \tilde{\beta} - g_i$ such that $\tilde{\beta}_j = (\tilde{\beta} - g_i)_+$. Therefore, $\tilde{\beta}$ can be calculated as

$$\tilde{\beta} = \frac{1}{c} \sum_{j=1}^{c} (\tilde{\beta} - g_j)_+. \quad (55)$$

According to the constraint $y^T 1 = 1$ and (54), we can define the following function

$$f(\tilde{\beta}) = \sum_{j=1}^{c} (g_j - \tilde{\beta}) - 1, \quad (56)$$

and the optimal $\tilde{\beta}$ should satisfy $f(\tilde{\beta}) = 0$. When (56) equals 0, the optimal $\tilde{\beta}$ can be calculated by the Newton method

$$\tilde{\beta}_{k+1} = \tilde{\beta}_k - \frac{f(\tilde{\beta}_k)}{f'(\tilde{\beta}_k)}. \quad (57)$$

Finally, the optimization procedure to sub-problem (44) is listed in Algorithm 2 and the procedure to solve problem (15) is summarized in Algorithm 3.

**Algorithm 2** The algorithm to solve sub-problem (45).

**Input:** vector $a_i \in \mathbb{R}^c$

**Output:** variable $x \in \mathbb{R}^c$

1: Compute $g = a_i - \frac{11^T c + 1}{c}$;
2: Use Newton’s method to obtain the root $\tilde{\beta}$ of (56);
3: Obtain the optimal solution by (54);

**Algorithm 3** The optimization algorithm of RSRRW.

**Input:** Labeled EEG data $X_l \in \mathbb{R}^{d \times n_l}$ and its corresponding label matrix $Y_l \in \mathbb{R}^{c \times n_l}$, unlabeled EEG data $X_u \in \mathbb{R}^{d \times n_u}$, parameters $\lambda$;

**Output:** The estimated label $Y_u \in \mathbb{R}^{c \times n_u}$.

1: Initialize $W$ randomly. Initialize each element of $Y_u$ as $\frac{1}{c}$. Initialize each element of $M$ as 0. Initialize each element of $s$ as 1;
2: while not converged do
3: Update $b$ via (22);
4: Update $W$ via (26);
5: Update $s$ by solve (27);
6: Update $M$ via (38);
7: Update $Y_u$ by solving (45) for each $i_{n_l+n_u}$;
8: end while

### 3.3. Discussion on RSRRW

In this section, we will discuss the variable $s$ and $\epsilon$-dragging methods in RSRRW. In addition, the complexity of the RSRRW model will also be briefly analyzed.

- We know that $s_i |_{i=1}^n$ reflects the importance level of the $i$-th sample. Through proof in Section 3.2, we can know that $s$ is a binary vector. When sample $i$ is a noise point, the loss caused by it is 0, that is, sample $x_i$ will not get involved in the training process. On the contrary, when the sample $x_i$ is a normal point, its weight is 1, meaning that it contributes to the training. Therefore, with the help of the sample weight $s$, RSRRW can automatically find outliers in the data and remove them, thereby preventing the model from skewing towards outliers. Comparison experiments suggest that RSRRW
performs better than DSLR. At the same time, through the visualization of $s$ in the
time dimension, we can intuitively observe the time distribution of outliers during the
experiment, which can also provide some references for future experimental design.
We will elaborate on this in Section 4.3 below.

- In [40], the $\epsilon$-dragging method was applied to supervised learning tasks and $B$
was obtained by calculating the label of the training set samples, so as to obtain the dragging
direction of each sample and achieve the purpose of expanding the distance between
classes. Considering that in the EEG emotion recognition task, there are usually few
labeled data samples, and the distribution of data from different sessions is quite
different. Therefore, it may be hard to obtain ideal results by wholly relying on the
supervised learning paradigm to complete the emotion recognition task. Therefore,
we extend the $\epsilon$-dragging method to the semi-supervised learning process and esti-
mate the dragging direction of each sample through the unlabeled sample category
probability obtained in the optimization process, realizing semi-supervised $\epsilon$-dragging
method. Ideally, if $Y_u$ can be estimated accurately, then we can get the exact dragging
direction of each sample. In the following experiments, we will intuitively show the
performance of $\epsilon$-dragging method.

- We analyze the asymptotic time complexity of our model in this section. Firstly,
in the optimization of $b$, the asymptotic time complexity is $O(n^2c + ndc + n^2c)$.
Secondly, in the optimization of $W$, the asymptotic time complexity of $(XKX^T + \lambda Q)^{-1}$ is
$O(n^2d + d^2c + d^3)$, the asymptotic time complexity of $XKZ^T$ is $O(n^2d)$. To calculate $K$,
considering that it depends on the operation of the diagonal matrix and the unit matrix,
so its time complexity is obviously lower than $O(n^2d)$. Then, as for the improvement
of $s$, its asymptotic time complexity is $O(n \log n)$. Finally, in the optimizing process of $M$,
its asymptotic time complexity is $O(dcn)$. In the optimization process of $Y_u$, since $W^TX$ has been completed in the process of updating $M$, its asymptotic time complexity
is $O(uc)$, where $u$ is the number of unlabeled samples. In general, $n \approx u > d \gg c$.
Therefore, the overall asymptotic time complexity of the RSRRW algorithm model is
$O(n^2dt)$, where $t$ is the number of iterations.

4. Experiments

4.1. Data Sets

The SEED-IV dataset contains EEG signals collected from subjects while they’re watching
movie clips. It selects 72 film clips with highly emotional contents as signals for
emotional induction. A total of 15 healthy right-handed subjects (including 7 males and
8 females) participated in the experiment, and each one needed to take exams in three
different sessions on different days. The four emotional states evoked by the video are
happy, fearful, neutral, and sad. In each session, one affective state corresponds to six trials.
The experimental process of each session is shown in Figure 3. There will be a 5 s start
prompt before each video clip is played, and the subjects will get a 45 s self-assessment
after the playback ends.

![Figure 3. Experimental protocol for SEED-IV [42].](image)

The EEG acquisition equipment used during the experiment included the ESI Neuroscan
system and a 62-electrode cap that conforms to the international 10–20 placement
standard. In the experiments mentioned in this paper, we used differential entropy (DE)
features extracted from five frequency bands, namely Delta (1–3 Hz), Theta (4–7 Hz), Alpha
(8–13 Hz), Beta (14–30 Hz) and Gamma (31–50 Hz). The sample $x_i$ is formed by concatenating the values of the 62 leads in each of the 5 frequency bands, resulting in a dimension of 310. More details can be found in [42].

4.2. Experimental Settings

During the experiments, we performed the cross-session emotion recognition task in chronological order. In other words, each subject is supposed to complete three tasks, including “session 1–session 2”, “session 1–session 3”, “session 2–session 3”. In the “session 1–session 2” task, the samples in session 1 are labeled data, and the samples in session 2 are unlabeled data. During the model training process, the samples from session 2 will be gradually labeled.

We compare the RSRRW algorithm with five closely related models, including the traditional semi-supervised linear square regression model (sLSR), semi-supervised support vector machine model (sSVM), the optimized rescaled linear square regression model (RLSR) [39], discriminative linear square regression model (DLSR) [40], and robust semi-supervised least squares regression model (RSLSR) [43]. RLSR is an optimization of sLSR, which adds a feature weight factor to describe feature importance on the basis of sLSR. DLSR is another optimization of sLSR. On the basis of LSR, it adds the $\epsilon$-dragging method to increase the distance between categories. In sSVM, we use a linear kernel. RSLSR adds sample weights on the basis of sLSR, which can filter out outliers in samples. In the above five methods, there are hyperparameters that need to be adjusted ($\lambda$ in sLSR, sSVM, RLSR, DLSR, RSLSR, RSRRW; $k$ in RSRRW, RLSR). In this experiment, the adjustment range of $\lambda$ is $\{2^{-10}, 2^{-9}, \cdots, 2^{10}\}$. For the RSSLS and RSRRW, the adjustment range of $k$ is $\{[0.8a], [0.81a], \cdots, a\}$, where $a$ is the number of samples. The iteration stopping condition is that the number of iterations reaches 100 or the rate of change of the objective function $obj = \frac{obj(t+1) - obj(t)}{obj(t)}$ is less than or equal to $1 \times 10^{-5}$. The typical number of iterations of RSRRW is around 70.

4.3. Results and Analysis

Tables 2–4 shows the recognition accuracy of the above models in the cross-session EEG emotional states recognition tasks. Among them, Table 2 shows the experimental results with session 1 serving as labeled data and session 2 as unlabeled data. Table 3 shows the experimental results with the labeled data in session 1 and the unlabeled data in session 3. Table 4 shows the experimental results with session 2 as labeled data and session 3 as unlabeled data. The highest accuracy in each group is marked in bold.

Table 2. Cross-session emotion recognition results (%) of session 1–session 2.

<table>
<thead>
<tr>
<th>Subject</th>
<th>sLSR</th>
<th>sSVM</th>
<th>RLSR</th>
<th>DLSR</th>
<th>RSLSR</th>
<th>RSRRW</th>
</tr>
</thead>
<tbody>
<tr>
<td>subject 1</td>
<td>60.22</td>
<td>72.72</td>
<td>75.60</td>
<td>62.50</td>
<td>77.04</td>
<td>76.20</td>
</tr>
<tr>
<td>subject 2</td>
<td>79.21</td>
<td>82.45</td>
<td>83.53</td>
<td>76.44</td>
<td>83.53</td>
<td>86.78</td>
</tr>
<tr>
<td>subject 3</td>
<td>65.26</td>
<td>71.39</td>
<td>77.88</td>
<td>79.21</td>
<td>77.88</td>
<td>78.00</td>
</tr>
<tr>
<td>subject 4</td>
<td>68.39</td>
<td>55.29</td>
<td>68.51</td>
<td>72.24</td>
<td>68.51</td>
<td>80.65</td>
</tr>
<tr>
<td>subject 5</td>
<td>61.78</td>
<td>72.60</td>
<td>53.33</td>
<td>66.83</td>
<td>56.85</td>
<td>73.44</td>
</tr>
<tr>
<td>subject 6</td>
<td>56.97</td>
<td>64.66</td>
<td>53.25</td>
<td>57.93</td>
<td>58.29</td>
<td>75.36</td>
</tr>
<tr>
<td>subject 7</td>
<td>73.80</td>
<td>70.19</td>
<td>80.89</td>
<td>79.33</td>
<td>82.33</td>
<td>89.90</td>
</tr>
<tr>
<td>subject 8</td>
<td>75.36</td>
<td>67.55</td>
<td>74.76</td>
<td>68.99</td>
<td>74.76</td>
<td>88.70</td>
</tr>
<tr>
<td>subject 9</td>
<td>60.82</td>
<td>68.63</td>
<td>62.38</td>
<td>63.70</td>
<td>54.33</td>
<td>59.74</td>
</tr>
<tr>
<td>subject 10</td>
<td>45.55</td>
<td>52.76</td>
<td>47.00</td>
<td>54.33</td>
<td>59.74</td>
<td>66.83</td>
</tr>
<tr>
<td>subject 11</td>
<td>54.33</td>
<td>54.69</td>
<td>59.74</td>
<td>52.40</td>
<td>61.30</td>
<td>68.15</td>
</tr>
<tr>
<td>subject 12</td>
<td>69.23</td>
<td>56.01</td>
<td>56.49</td>
<td>63.58</td>
<td>62.14</td>
<td>75.00</td>
</tr>
<tr>
<td>subject 13</td>
<td>63.10</td>
<td>62.38</td>
<td>58.77</td>
<td>71.15</td>
<td>60.58</td>
<td>73.80</td>
</tr>
<tr>
<td>subject 14</td>
<td>84.01</td>
<td>87.62</td>
<td>85.22</td>
<td>78.85</td>
<td>85.22</td>
<td>90.26</td>
</tr>
<tr>
<td>subject 15</td>
<td>72.60</td>
<td>92.79</td>
<td>98.56</td>
<td>97.36</td>
<td>98.56</td>
<td>98.56</td>
</tr>
<tr>
<td>Avg.</td>
<td>66.04</td>
<td>68.78</td>
<td>69.13</td>
<td>69.66</td>
<td>71.82</td>
<td>79.50</td>
</tr>
</tbody>
</table>
Table 3. Cross-session emotion recognition results (%) of session 1–session 3.

<table>
<thead>
<tr>
<th>Subject</th>
<th>sLSR</th>
<th>sSVM</th>
<th>RLSR</th>
<th>DLSR</th>
<th>RSLSR</th>
<th>RSRRW</th>
</tr>
</thead>
<tbody>
<tr>
<td>subject 1</td>
<td>73.84</td>
<td>81.02</td>
<td>80.78</td>
<td>81.63</td>
<td>83.09</td>
<td>92.58</td>
</tr>
<tr>
<td>subject 2</td>
<td>80.29</td>
<td>86.86</td>
<td>91.00</td>
<td>86.13</td>
<td>92.21</td>
<td>92.34</td>
</tr>
<tr>
<td>subject 3</td>
<td>41.12</td>
<td>53.65</td>
<td>57.06</td>
<td>37.47</td>
<td>61.68</td>
<td>63.38</td>
</tr>
<tr>
<td>subject 4</td>
<td>81.39</td>
<td>60.58</td>
<td>80.29</td>
<td>92.94</td>
<td>80.29</td>
<td>74.21</td>
</tr>
<tr>
<td>subject 5</td>
<td>68.86</td>
<td>79.93</td>
<td>72.51</td>
<td>82.60</td>
<td>74.33</td>
<td>82.73</td>
</tr>
<tr>
<td>subject 6</td>
<td>74.70</td>
<td>74.45</td>
<td>77.13</td>
<td>82.48</td>
<td>79.93</td>
<td>83.45</td>
</tr>
<tr>
<td>subject 7</td>
<td>61.31</td>
<td>84.91</td>
<td>80.66</td>
<td>82.12</td>
<td>87.23</td>
<td>92.34</td>
</tr>
<tr>
<td>subject 8</td>
<td>86.25</td>
<td>54.87</td>
<td>83.21</td>
<td>84.55</td>
<td>84.91</td>
<td>92.46</td>
</tr>
<tr>
<td>subject 9</td>
<td>65.33</td>
<td>54.87</td>
<td>53.77</td>
<td>57.91</td>
<td>63.75</td>
<td>66.79</td>
</tr>
<tr>
<td>subject 10</td>
<td>45.01</td>
<td>60.83</td>
<td>41.85</td>
<td>39.90</td>
<td>64.48</td>
<td>66.55</td>
</tr>
<tr>
<td>subject 11</td>
<td>61.19</td>
<td>54.74</td>
<td>71.65</td>
<td>63.38</td>
<td>73.72</td>
<td>80.54</td>
</tr>
<tr>
<td>subject 12</td>
<td>55.60</td>
<td>58.39</td>
<td>67.64</td>
<td>61.92</td>
<td>70.92</td>
<td>76.28</td>
</tr>
<tr>
<td>subject 13</td>
<td>47.57</td>
<td>52.31</td>
<td>60.95</td>
<td>55.47</td>
<td>63.99</td>
<td>71.41</td>
</tr>
<tr>
<td>subject 14</td>
<td>70.92</td>
<td>71.05</td>
<td>79.44</td>
<td>67.76</td>
<td>90.88</td>
<td>92.33</td>
</tr>
<tr>
<td>subject 15</td>
<td>72.87</td>
<td>80.54</td>
<td>93.07</td>
<td>85.04</td>
<td>95.01</td>
<td>96.47</td>
</tr>
<tr>
<td>Avg.</td>
<td>65.75</td>
<td>67.27</td>
<td>72.73</td>
<td>70.75</td>
<td>77.76</td>
<td>81.25</td>
</tr>
</tbody>
</table>

Table 4. Cross-session emotion recognition results (%) of session 2–session 3.

<table>
<thead>
<tr>
<th>Subject</th>
<th>sLSR</th>
<th>sSVM</th>
<th>RLSR</th>
<th>DLSR</th>
<th>RSLSR</th>
<th>RSRRW</th>
</tr>
</thead>
<tbody>
<tr>
<td>subject 1</td>
<td>56.20</td>
<td>70.92</td>
<td>71.41</td>
<td>71.90</td>
<td>70.92</td>
<td>82.24</td>
</tr>
<tr>
<td>subject 2</td>
<td>80.29</td>
<td>77.37</td>
<td>86.73</td>
<td>85.89</td>
<td>86.01</td>
<td>91.50</td>
</tr>
<tr>
<td>subject 3</td>
<td>59.85</td>
<td>65.09</td>
<td>70.81</td>
<td>69.83</td>
<td>71.78</td>
<td>78.59</td>
</tr>
<tr>
<td>subject 4</td>
<td>83.09</td>
<td>72.14</td>
<td>77.65</td>
<td>77.74</td>
<td>76.76</td>
<td>87.71</td>
</tr>
<tr>
<td>subject 5</td>
<td>73.84</td>
<td>69.49</td>
<td>71.90</td>
<td>77.37</td>
<td>78.95</td>
<td>82.70</td>
</tr>
<tr>
<td>subject 6</td>
<td>88.32</td>
<td>65.69</td>
<td>87.43</td>
<td>79.56</td>
<td>87.10</td>
<td>91.73</td>
</tr>
<tr>
<td>subject 7</td>
<td>82.60</td>
<td>87.83</td>
<td>88.59</td>
<td>83.45</td>
<td>88.93</td>
<td>93.80</td>
</tr>
<tr>
<td>subject 8</td>
<td>77.98</td>
<td>75.30</td>
<td>77.11</td>
<td>70.32</td>
<td>81.39</td>
<td>85.85</td>
</tr>
<tr>
<td>subject 9</td>
<td>50.24</td>
<td>51.19</td>
<td>49.78</td>
<td>42.09</td>
<td>59.85</td>
<td>69.95</td>
</tr>
<tr>
<td>subject 10</td>
<td>66.91</td>
<td>60.34</td>
<td>73.27</td>
<td>65.94</td>
<td>72.87</td>
<td>76.28</td>
</tr>
<tr>
<td>subject 11</td>
<td>47.93</td>
<td>61.68</td>
<td>49.36</td>
<td>52.55</td>
<td>52.31</td>
<td>60.71</td>
</tr>
<tr>
<td>subject 12</td>
<td>80.41</td>
<td>66.79</td>
<td>72.74</td>
<td>85.28</td>
<td>79.56</td>
<td>81.27</td>
</tr>
<tr>
<td>subject 13</td>
<td>54.50</td>
<td>56.33</td>
<td>53.01</td>
<td>46.47</td>
<td>67.03</td>
<td>79.44</td>
</tr>
<tr>
<td>subject 14</td>
<td>91.48</td>
<td>94.89</td>
<td>90.69</td>
<td>91.00</td>
<td>94.04</td>
<td>95.86</td>
</tr>
<tr>
<td>subject 15</td>
<td>88.08</td>
<td>78.71</td>
<td>92.88</td>
<td>90.39</td>
<td>94.89</td>
<td>98.91</td>
</tr>
<tr>
<td>Avg.</td>
<td>72.12</td>
<td>70.25</td>
<td>74.22</td>
<td>70.75</td>
<td>77.76</td>
<td>83.77</td>
</tr>
</tbody>
</table>

Through the observation of the above data, we found the following meaningful points.

- Although the EEG data from different sessions have significant differences in distribution, our RSRRW model still achieves high recognition accuracy. Specifically, the average accuracies of RSRRW in the three cross-session tasks are 79.50%, 81.25%, and 83.77%, respectively, which are 7.68%, 3.49%, and 6.28% higher than the runner-up.

- Compared with sLSR, the RSLSR model adds the sample weight factor $s$. From the average performance of the model, in the three cross session tasks, RSLSR is 5.78%, 12.01%, and 5.37% higher than sLSR, which shows that the method of dynamically screening samples by using the sample probability weight $s$ has a good effect. That is, when the sample error is too large, we can discard these samples to ensure that the model will not shift to noise.

- Compared with sLSR, the RLSR model adds the feature weight factor $\Theta$. From the average performance of the model, RLSR is 3.09%, 6.98%, and 2.10% higher than sLSR, respectively, indicating that the model performance has been improved to a certain extent through adaptive feature weight learning. That is, for the EEG emotion recognition task, the EEG contribution of different frequency bands and different leads will be different.
• Compared with sLSR, the DLSR model adds \( \epsilon \)-dragging method. From the average performance of the model, DLSR is 3.62%, 5.00%, and 0.53% higher than sLSR, respectively. It can be found that the \( \epsilon \)-dragging method can effectively improve the model performance.

In order to illustrate the performance advantages of the RSRRW model compared to other models as far as possible, we use Friedman test [44] and Nemenyi test method [45] for significance testing. Firstly, we put forward the null hypothesis \( H_0 \) that “the experimental results of each group come from the population with no significant difference in numerical value”. Obviously, the alternative hypothesis \( H_1 \) is “the experimental results of each group come from the population with obvious differences in numerical value”. Secondly, determine the number of models \( k = 6 \), and the groups of cross-subject EEG emotion state recognition tasks \( N = 45 \). Thirdly, we rank the accuracy under each cross session emotional state recognition task (the higher the recognition accuracy, the higher the ranking). After sorting, the average rank \( r_i \) of each model is calculated separately, and the result is shown in Table 5.

**Table 5. The average rank of each model.**

<table>
<thead>
<tr>
<th>Model</th>
<th>RSRRW</th>
<th>RSLR</th>
<th>DLSR</th>
<th>sLSR</th>
<th>sSVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r )</td>
<td>1.22</td>
<td>2.77</td>
<td>3.96</td>
<td>3.98</td>
<td>4.60</td>
</tr>
</tbody>
</table>

\( r \): Average rank.

Fourthly, we calculate the variable

\[
\tau_F = \frac{(N-1)\tau_{X^2}}{N(K-1) - \tau_{X^2}},
\]

where

\[
\tau_{X^2} = \frac{12N}{K(K+1)} \left( \sum_{i=1}^{K} r_i^2 - \frac{K(K+1)^2}{4} \right).
\]

In this experiment, considering that the number of models \( k = 6 \) and the number of datasets \( N = 45 \) are relatively large, it can be considered that \( \tau_{X^2} \sim \chi^2(k-1) \). Calculated by MATLAB, \( \tau_F = 39.9503 \), which is much larger than the critical value 2.2551 of the \( F \) test when \( k = 6, n = 45 \). Therefore, it can be accepted that in the case of confidence level \( \alpha = 0.05 \), reject the null hypothesis \( H_0 \) and accept the alternative hypothesis \( H_1 \) (that is, the probability of error in the judgment of rejecting the null hypothesis is 5%).

Based on this, in order to further distinguish the performance of each model, we performed post hoc test detection based on the Nemenyi test on the above experimental results. First, we calculate the critical distance (CD) of average ranking difference under this set of data through

\[
CD = q_a \sqrt{\frac{K(K+1)}{6N}},
\]

where \( q_a \) represents the critical value in the Nemenyi test. In the case of \( k = 6, \alpha = 0.05 \), the critical value is \( q_a = 2.850 \). By calculation, \( CD = 1.1241 \). According to the above experimental results, we have drawn Figure 4. In the figure, we use a vertical line to represent the critical region, and the midpoint of the line is the average rank of the model. If two lines do not overlap, then we can draw into the conclusion that these two model differ from each other significantly, otherwise, the former bears some resemblance to the latter. For example, the rank value of RSRRW is 1.22, and the rank value of RSLR is 2.77. Their corresponding vertical lines do not overlap, so there is a significant difference between the two at the confidence level of 0.05. Therefore, RSSRW significantly outperforms RSLR in the cross-session emotion recognition task.
4.4. EEG Spatial-Frequency Activation Patterns Mining

For the feature weight $\Theta$ obtained in our experiment, we can determine the contribution of each frequency band and lead in emotion recognition according to the corresponding relationship between spectral features and EEG frequency bands and channels [33]. The average band importance for all 45 experiments is shown in Figure 5a, from which we can find that the Gamma band contributes the most in distinguishing different emotional states. Based on the consensus that brain regions may correlate differently with the occurrence of affective effects, we can also determine the importance of different EEG channels by feature weights $\Theta$. As shown in Figure 5b, we show channel importance on the EEG topography and find that channels in the temporal and parietal lobes have higher importance. In Figure 6, we show the weight values of the top 10 channels. We believe that P7, CP2, TP8, CZ, FC5, TP7, F8, T7, FPZ and P8 have high importance for the cross session emotion recognition task. The layout of the 10 key leads is shown in Figure 7.

![Figure 4. Nemenyi test result.](image)

![Figure 5. The average importance of EEG channels (a) and frequency bands (b) obtained by RSRRW.](image)
4.5. Effect of the Dragging-Matrix

For samples from different categories, we want to embed the labels of the samples into the objective function to achieve the purpose of increasing the distance between categories. Therefore, we introduce the $\epsilon$-dragging method to drag the regression targets from different classes in opposite directions by matrix $M$ and $B$. The dragging-matrix $M$ can significantly expand the distance between categories, and each column can be regarded as a binary regressor. For samples grouped into class $j$, their corresponding labels become $1 + m_{ij}$, and for samples that do not belong to class $j$, their labels become $-m_{ij}$, where $m_{ij} \geq 0$.

To observe the performance of the $\epsilon$-dragging method, we selected data from subject 15: session 1–session 2, subject 5: session 2–session 3 to plot the $Y$ value of unlabeled samples. In Figure 8, we plotted certain columns of predicted labels $Y$ for the above experiments. Then, modify the order of the samples, put the samples belonging to the same category together, so as to observe the distance between samples under each emotion category. Each color represents an affective state, with dashed lines for negative samples and solid lines for positive samples. It can be found that this method can significantly increase the distance between samples.
4.6. Effect of the Sample Probability Weight

In the RSRRW algorithm, we add the weight factor $s$ to the objective function. During the learning process, all samples are sorted according to the error after training. The $k$ samples with the smallest error are marked as 1, and the others are 0. Through the above process, outliers will be automatically located and eliminated, avoiding the loss of accuracy caused by the deviation of the model from these values.

Below, we have selected EEG data from subject 1: session 3, subject 9: session 2, and subject 10: session 2 to plot the distribution of noisy samples. In Figure 9, the background color represents the real emotion category, the horizontal axis represents time, and the vertical axis represents the $s$ values of different samples. Due to the large number of actual samples, so we merely focus on the position of noisy samples. Each point in this figure does not exactly correspond to one sample in reality. The blue points indicate that the sample is normal, and the red points represent abnormal samples.

Figure 8. Examples to show the effectiveness of the $\epsilon$-dragging.

Figure 9. Cont.
In the experimental paradigm of SEED-IV, each subject only had 45 s to evaluate and rest between trials. We believe that such short time cannot make some subjects completely recover from the previous emotional state. Therefore, in the follow-up EEG emotion experiment, the rest time of the subjects after each video clip should be appropriately extended to ensure that the subjects can get enough rest before the next experiment. Alongside this, it can also improve the quality of samples to a certain extent and ensure that the samples and labels have better consistence.

5. Conclusions

In this paper, we propose a model of Retargeted Semi-supervised Regression with Robust Weights Self-learning (RSRRW). We will summarize this paper from two aspects: Advantages and Results.

Advantages: (1) Compared with the DLSR method, the robustness of the model is fully considered, and binary weights are added to each sample to determine whether the sample is a noise point, which can ensure the performance of the model when some labels are inaccurate. Compared with the traditional LSR method, RSRRW introduces feature weights $\Theta$ to distinguish the weights of different features in the EEG emotion recognition task, thus obtaining the emotional activation mode. (2) Compared with the traditional LSR method, RSRRW introduces feature weight variable $\Theta$ to distinguish the weights of different features in the EEG emotion recognition task, thus obtaining the emotional activation mode. (3) RSRRW implements the $\epsilon$-dragging method under the semi-supervised learning paradigm to expand the sample distance between different categories and improve the model performance. (4) The model has achieved good experimental results on the SEED-IV emotion recognition dataset, the average accuracies of RSRRW in the three cross-session tasks are 79.50%, 81.25%, and 83.77%, respectively, which are 7.68%, 3.49%, and 6.28% higher than the runner-up.

Results: (1) The RSRRW model greatly improves the accuracy of emotion recognition. (2) It is much more important to consider Gamma and Delta EEG frequency bands and the EEG channels locating within the temporal and (central) parietal lobes in emotion recognition tasks.
Author Contributions: Conceptualization, Y.P. and Z.C.; Data curation, Z.C. and S.D.; Investigation, Y.P.; Methodology, Z.C. and Y.P.; Software, Z.C. and Y.P.; Validation, Y.P. and S.D.; Writing—original draft preparation, Z.C. and Y.P.; Writing—review and editing, Z.C. and Y.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (61971173), Fundamental Research Funds for the Provincial Universities of Zhejiang (GK209907299001-008) and National Innovation Training Program for College Students (202210336051).

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Institutional Review Board (or Ethics Committee) of Shanghai Jiao Tong University (protocol code 2017060).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Not applicable.

Acknowledgments: The authors also would like to thank the anonymous reviewers for their comments on this work.

Conflicts of Interest: The authors declare no conflict of interest.

References
10. Li, W.; Peng, Y. Transfer EEG Emotion Recognition by Combining Semi-Supervised Regression with Bipartite Graph Label Propagation. Systems 2022, 10, 111. [CrossRef]
35. Ekman, P. Facial expression and emotion. Am. Psychol. 1993, 48, 384. [CrossRef]