J Inf Process Syst, Vol.17, No.3, pp.441~452, June 2021
https://doi.org/10.3745/JIPS.01.0075

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

A Federated Multi-Task Learning Model Based on Adaptive Distributed Data Latent Correlation Analysis

Shengbin Wu* and Yibai Wang*

Abstract

Federated learning provides an efficient integrated model for distributed data, allowing the local training of different data. Meanwhile, the goal of multi-task learning is to simultaneously establish models for multiple related tasks, and to obtain the underlying main structure. However, traditional federated multi-task learning models not only have strict requirements for the data distribution, but also demand large amounts of calculation and have slow convergence, which hindered their promotion in many fields. In our work, we apply the rank constraint on weight vectors of the multi-task learning model to adaptively adjust the task's similarity learning, according to the distribution of federal node data. The proposed model has a general framework for solving optimal solutions, which can be used to deal with various data types. Experiments show that our model has achieved the best results in different dataset. Notably, our model can still obtain stable results in datasets with large distribution differences. In addition, compared with traditional federated multi-task learning models, our algorithm is able to converge on a local optimal solution within limited training iterations.

Keywords

Data Distribution, Federated Multi-Task Learning, Rank Constraint, Underlying Structure

1. Introduction

The concept of federated learning is to independently construct models in local distributed data, and then concentrate these models into a federated algorithm with high efficiency and strong recognition abilities through the encryption technology [1,2]. It is proposed by the Google Scholar to analyze Android user data, and the algorithm allows each user to locally train the model, which can effectively protect the user's privacy [3,4]. With the growing storage and computation of Internet data, it is a huge waste of computing resources to train all the data together and the security of users cannot be guaranteed. Unlike traditional machine learning models that train all data simultaneously, the decentralized operation of federated learning provides a safe and efficient framework for the integration and the processing of distributed data.

The multi-task learning model can learn several related tasks at the same time; thus, the obtained discrimination ability of the final model is better than any single task-based model [5,6]. The goal of multi-task learning is to mine the underlying structure between different tasks, which can improve the identify ability of each item model. According to the data observation between tasks, the multi-task

Manuscript received October 21, 2020; first revision April 9, 2021; accepted May 4, 2021.
Corresponding Author: Yibai Wang (hunanhappywang@163.com)
* School of Information Engineering, Changsha Medical University, Changsha, Hunan, China (shengbinwu123456@163.com, hunanhappywang@163.com)

learning model can be roughly divided into two categories: the first category assumes that the distribution between the data is known and can be served as prior knowledge [7,8], while the second category assumes that the model can directly obtain the data relationship during the training process [9,10]. Naturally, when treating each node in the federated learning as a task, we can regard this as a special form of multi-task learning framework. Specifically, the multi-task learning method simultaneously learn related models, and the data source of different tasks are from the same distribution. While the collection devices of federated learning are usually independent and have technical differences, the machine of each node is also inconsistent [11].

The distributed multi-task learning model is a relatively novel research field, and it focuses on the distributed data in each task [12]. In research works, it is assumed that the task correlation of distributed data and the structure is a joint sparse setting. Several typical machine learning models are applied to extract the latent-shared information between distributed tasks, such as lasso [13], multi-kernel learning [14], low-rank representation [15] and etc. In the selection of loss function, l_1 norm, l_2 norm, and their mixed forms are adopted to affect all tasks. The distributed multi-task learning model is similar to the federated multi-task learning, but system challenges in the federal learning prevent the distributed model from exerting its effects [16]. The model [17] can effectively solve the system challenges and a general solution framework is proposed, guiding the algorithm to the convergence on a better optimal solution.

However, the above model cannot flexibly measure the similarity between tasks. For example, when the behavioral habits in a group of Android users are highly similar, or there is a large difference among different regions, the penalty function needs to be changed adaptively. Meanwhile, for small sample datasets in the multi-task learning (such as medicine imaging), obtaining a patient's information is time-consuming and costly; therefore, the analysis of each sample is extremely valuable. In our work, we propose a federated multi-task learning model based on the adaptive distributed data correlation analysis (FMLADA). We utilize the value of p in $l_{2,p}$ norm to control the sparse of tasks [18,19], and our model still offers a better performance in small sample datasets.

Overall, the main contributions of this article are:

1) A flexible and efficient distributed data constraint method is introduced into the federated multi-task learning model, which can adaptively control the correlation degree of different data sources.
2) We propose a general optimization framework for the federated multi-task learning model based on the adaptive distributed data correlation analysis algorithm. With the non-convex of $l_{2,p}$ norm, our model can guarantee the convergence of the main function and obtain a local optimal solution.

2. Related Work

2.1 Multi-Task Learning

Currently, most machine learning models are based on single task learning. For complex problems, the resolution process can be decomposed into simple and independent sub-problems to be separately solved. However, each sub-problem is related to each other through some shared factors or through representation. Multi-task learning will put the related tasks (sharing the latent information) together to learn, which can achieve better generalization performance [20]. Multi-task learning is a kind of derivation transfer learning method [5]. The main task uses the domain-related information possessed by

the training signal of the related task as a machine learning method that usually derives the deviation to improve the generalization. Multiple related tasks are learned in parallel at the same time, the gradient is backpropagated, and the underlying shared representation is used to help each other learn to improve the generalization. Shared representation is the focus of multi-task learning, with a purpose to improve the generalization, which can be divided into parameter-based sharing and constraint-based sharing [21].

2.2 Distributed Machine Learning

The emergence of large-scale training data provides a material basis for training large models. These models can easily have millions or even billions of parameters. On one hand, these large-scale machine learning models have super expressive capabilities and can help solving many difficult learning problems. On the other hand, they also have drawbacks, i.e., they are easy to overfit in the training set. It has achieved good results on the known test data, but the performance is unsatisfactory on the unknown test data, which force the scale of the training data. The results inevitably lead to the double challenges of big data and large models, putting forward new requirements for computing power and storage capacity. The computational complexity is high, and the single-machine training may consume an unacceptable amount of time. Therefore, more parallel processors or computer clusters have to be used to complete the training tasks; the large storage capacity makes the single-machine unable to meet the demand and have to use the distribution style storage.

There are roughly three reasons for the need to use distributed machine learning: the first is that the amount of calculation is too large; the second is the huge amount training data; and the third is the large scale of the model. When the amount of calculation is too large, the multi-thread or multi-machine parallel computing based on shared memory (or virtual memory) can be adopted [22]. In the case of too much training data, the data needs to be divided and distributed to multiple working nodes for training, so that the local data of each working node is within tolerance. Each working node will train a sub-model based on local data, and will communicate with other working nodes according to certain rules (the content of communication is mainly sub-model parameters or parameter updates) to ensure the final effective integration from each working node. The result of training is a global machine learning model [23]. For the case where the model scale is too large, the model needs to be divided and assigned to different working nodes for training. Different from data parallelism, the dependency between each sub-model within the framework of model parallelism is quite strong, because the output of one sub-model may be the input of another sub-model. If the communication of intermediate calculation results is not carried out, the whole cannot be completed. In general, model parallelism has higher requirements for communication. The above three distributed machine learning situations are usually mixed together in practice. When it is necessary to distinguish the proportions, data parallelism is still the most common situation, because the large amount of training data leads to the slow training speed, and this is still the main contradiction in the field of distributed machine learning.

2.3 Federated Learning

Federated learning is an emerging basic artificial intelligence technology, which was first proposed by Google in 2017 [1]. It was originally used to solve the problem of the local update of Android mobile phone end users. The design goal is to ensure the exchange of big data. On the premise of ensuring information security, protecting the privacy of terminal data and personal data, and ensuring legal

compliance, the high-efficiency machine learning is carried out between multiple parties or multiple computing nodes. Among them, the machine learning algorithms that can be used in federated learning are not limited to neural networks, but also include important algorithms such as random forests.

Overall, federated learning can be divided into three types, including horizontal federated learning, vertical federated learning and federated transfer learning (FML) [3]. Horizontal federated learning is aimed at the situation where the features of two datasets overlap more and the samples overlap less. In this learning, the dataset is divided according to the horizontal (i.e., user dimension), and part of the data with the same characteristics but not from the same sample are taken out to conduct training models. On the other hand, vertical federated learning is used in the case where the two datasets overlap more and the features overlap less. The dataset is divided according to the vertical direction (i.e., feature dimension), and the same samples not the same features are taken out for training. Federated transfer learning does not directly segment the data when the user and user characteristics of the two datasets are less overlapped, but uses the transfer learning to overcome the lack of data or labels.

2.4 Federated Multi-Task Learning

Intuitively, the information sharing framework between different tasks of multi-task learning provides a natural choice to combine with the node network in the federated setting. When federated learning is applied to multi-task learning, a correlation model is designed within this framework to explore the potential correlation structure between different tasks. The models in different tasks are the optimal solutions for processing the distributed data. The federated learning algorithm integrates the classification results of different tasks and then feeds back the latest convergence direction to each model to improve its robustness and generalization ability.

For distributed multi-task data, due to the inconsistency of data distribution, the algorithm is easily to fall into the local optimal solution, which impedes the effectiveness of federated learning model in integration and optimization. In our work, we will propose a general distributed multi-task algorithm solution framework, which can effectively prevent the model from falling into a poor local optimal solution.

3. Proposed Method

3.1 $l_{2,p}$-norm based Minimization

Supposing $X \in R^{d\times n}$ represents the training sample, and $x_i \in R^{d\times 1}$ is a sample, where d is the dimensionality of features and n denotes numbers of the sample. When the algorithm is applied to solve two classification problems with the square loss strategy, its main function can be written as:

$$\min\left\|y - X^T G\right\|_2^2 + \lambda / 2R(g) \tag{1}$$

where G is the projection vector, λ is the regularization parameter, and $y = \left[y_1,...,y_n\right]^T \in R^{n\times 1}$ contains the label set of all samples. When the training sample set is distributed or is relatively sparse, l_1 norm and l_p are usually used for the regular term. By changing the value of p, we can adaptively constrain the sparsity between samples.

For the multi-task model, it is necessary to constrain the correlation between different projects at the same time. Then the vector regular terms l_1 and l_p can be rewritten as $l_{2,1}$ and $l_{2,p}$. $l_{2,1}$ means that the constrain l_2 norm is for the in-line and the l_1 norm is for the inter-line. $l_{2,p}$ is a more flexible matrix constrain, which limits the degree of association between different tasks by adjusting the value of p. Its function form can be written as:

$$\|G\|_{2,p}^{p} = \sum_{r=1}^{d} \left(\sum_{j=1}^{m} g^{(j)2} \right)^{p/2} = \sum_{r=1}^{d} \|g_r\|_2^p \tag{2}$$

In the following text, we will design a unified framework for solving the approximate optimal solution of the non-convex $l_{2,p}$ norm, and prove both the convergence of the result and the maximum approximation of the original solution (Fig. 1).

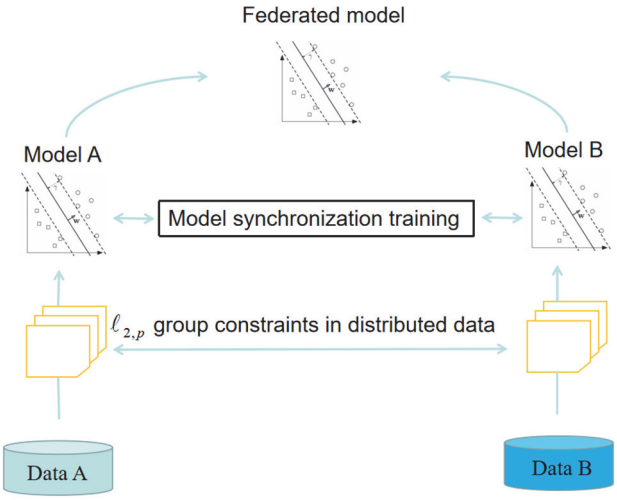

Fig. 1. The framework of our proposed method.

3.2 Main Function

For group data, the $l_{2,1}$ norm is a very effective constraint, but it may lack flexibility and effectiveness when processing distributed data [24]. For distributed multi-task learning models, their datasets consist of clean information samples and noisy samples. In this case, using $l_{2,p}$ norm offers a good choice for federated multi-task learning. The sparseness of the data can be adaptively controlled by changing the value of p. When the data distribution between tasks differs greatly, the value of p will approach 0; while the data between tasks are relatively similar, the value of p will approach 1.

Let $X^{(m)}$ be the m-th task data matrix, $g^{(m)} \in R^{d \times n}$ denotes the projection vector of m-th data, where d is the dimensionality of features, n is the number of samples and m=1,…,M. In our work, we use a matrix G to store the projection matrix of all tasks, where $G = \left(g^{(1)},..., g^{(M)}\right)$. The main function can be formulated as follows:

$$\min_g \frac{1}{2} \sum_{m=1}^{M} \left\| Y - X^{(m)T} g^{(m)} \right\|_2^2 + \lambda/2 \|G\|_{2,p}^{p} \tag{3}$$

Because of the non-convex of Eq. (3), we adopt the alternating direction method of multipliers (ADMM) framework to independently solve the parameters in the equation [25]. However, there are often inconsistent data distributions between different tasks, which may affect the performance of federation model to integrate and optimize all blocks. To alleviate the problem, we apply a semantic alignment strategy to actively align the data of different tasks. The Eq. (3) will be:

$$\min_{g,Q^{(m)}} \frac{1}{2}\sum_{m=1}^{M}\left\|Q^{(m)}Y - X^{(m)T}g^{(m)}\right\|_2^2 + \lambda/2\left\|G\right\|_{2,p}^{p} \quad s.t.\, Q^{(m)T}Q^{(m)} = I \tag{4}$$

where $Q^{(m)} \in R^{n\times n}$, and I is the diagonal matrix whose diagonal elements are all 1, while other elements are 0.

3.3 Optimization

Then, the augmented Lagrangian [26] function of Eq. (4) can be written as:

$$L\left(Q^{(m)},G\right) = \frac{1}{2}\sum_{m=1}^{M}\left\|Q^{(m)}Y - X^{(m)T}g^{(m)}\right\|_2^2 + \lambda/2\left\|G\right\|_{2,p}^{p} \tag{5}$$

In Eq. (5), the parameters $Q^{(m)}$ and G are independent each other, and we can fix the other parameters when solving one parameter. The solution of $Q^{(m)}$ and G are presented as follows:

$$\min_{Q^{(m)}} L\left(Q^{(m)},G\right) = \frac{1}{2}\sum_{m=1}^{M}\left\|Q^{(m)}Y - X^{(m)T}g^{(m)}\right\|_2^2 \quad s.t.Q^{(m)T}Q^{(m)} = I \tag{6}$$

It can be equivalently written as:

$$Q^{(m)} = \min_{Q^{(m)}} tr\left(Y^T Q^{(m)T}Q^{(m)}Y - 2Q^{(m)T}YX^{(m)T}g^{(m)} + X^{(m)T}g^{(m)T}g^{(m)}X^{(m)}\right) \quad s.t.Q^{(m)T}Q^{(m)} = I \tag{7}$$

Then it can be simplified to:

$$Q^{(m)} = \max_{Q^{(m)}} Q^{(m)T}YX^{(m)T}g^{(m)} \quad s.t.Q^{(m)T}Q^{(m)} = I \tag{8}$$

Let $U^*S^*V^{*T} = SVD\left(YX^{(m)T}g^{(m)}\right)$, based on the work in xx, the solution of $Q^{(m)}$ can be obtained as:

$$Q^{(m)} = U^*V^{*T} \tag{9}$$

The solution of G:

Because of the non-convex of $l_{2,p}$ norm, we use iterative reweighted least squares (IRLS) to solve this problem. For the coefficient matrix G, we define the diagonal weighting matrix W as:

$$w_{ii} = \frac{p}{2}\left\|G_r\right\|_2^{p-2} \tag{10}$$

where w_{ii} is the i-th diagonal element, and G_r denotes the r-row of G. Then Eq. (5) can be approximately represented as:

$$\min_{g^{(m)}} = \frac{1}{2}\left\|Q^{(m)}Y - X^{(m)T}g^{(m)}\right\|^2 + \lambda/2\,tr\left(G^T WG\right) \tag{11}$$

The close-form solution of $g^{(m)}$ is:

$$g^{(m)} = \left(\lambda G + X^{(m)T} X^{(m)}\right)^{-1} \tag{12}$$

3.4 Convergence Analysis and Computational Complexity

In terms of the calculation amount, each iteration of our equation involves the calculation of two parameters $Q^{(m)}$ and $g^{(m)}$ which have the close-form solutions. The complexity of computing $g^{(m)}$ is $O(nm + n^2)$, where *m*, *n* are the numbers of features and samples. Let the iteration number of algorithm FMLADA be T_1, the time complexity of computing $g^{(m)}$ is $OT_1(nm + n^2)$.

Algorithm 1. Federated multi-task learning model based on adaptive distributed data correlation analysis (FMLADA)

Server executes:

initialize ω_0

$m \leftarrow \max\left(M_1, M_2, ..., M_n\right)$

for each round 1,2,..., do

for each modal $m \in M_t$ in parallel do:

$\omega_{t+1}^k \leftarrow ParaUpdate\left(k, \omega_t\right)$

$\omega_{t+1} \leftarrow \sum_{m=1}^{Mt} \frac{n_m}{n} \omega_{t+1}^k$

ParaUpdate(m, ω_t)

for each local epoch *i* from 1 to *N* do

for each modal m do

$\omega \leftarrow \omega - \eta \nabla \ell\left(\omega; b\right)$

return ω to server

4. Experiment

4.1 Datasets

In order to verify the effectiveness of federated learning for the distributed multi-task classification model, we used several medical imaging data, including Alzheimer's Disease Neuroimaging Initiative (ADNI, https://loni.usc.edu/) and the depression database [27]. For patients, it is of great practical significance to use their medical imaging data to accurately determine pathological information. In pathological analysis, it is not only important to make the simple judgment of whether or not they are sick, but also to precisely analyze the disease stage of the patient and to report it to the medical team for corresponding treatment strategies. The different stages can be regarded as tasks. Our model is to determine the condition of the patient in the given medical imaging data, and then compare it with the given label to measure the effect of the model. Due to the differences in data between different patients, for example, a person with multiple comprehensive diseases and a patient with only a few clinical phenomena, their data are inconsistent. The former often contains the judgment of other noise influence

models, while the latter is often regarded as a clean sample and is easy to classify. Using the potential information between different tasks can improve the classification ability of the model. In our experiment, we will analyze the effect of introducing the federated learning on the model's robustness and generalization ability.

The ADNI dataset includes 913 patients with five pathological stages, including 160 Alzheimer disease (AD), 82 significant memory concern (SMC), 272 early mild cognitive impairment (EMCI), 187 late mild cognitive impairment (LMCI), and 211 normal control (NCs). Table 1 shows the details of 913-ADNI dataset. The depression data includes three stages: mild, moderate, and severe. There are 48 subjects participated in the experiment, of which 24 are patients and 24 are healthy subjects. Their sex, age and education level are similar. The experimental images are obtained from a standard database [27] and include five features, including bias positive attention, bias negative attention, the rate of positive pupil diameter, the rate of negative pupil diameter and the positive saccadic slope. Through the eye tracking systems, we can obtain 1,728 subjects, including 864 healthy subjects, and 864 depressed subjects.

Table 1. Demographic characteristics of the studied sample in the 913-ADNI database

	HC	SMC	EMCI	LMCI	AD
Number of subjects	210	82	272	187	160
Sex					
Male	109	33	153	108	95
Female	101	49	119	79	65
Age (yr)	76.13±6.54	72.45±5.67	71.51±7.11	73.86±8.44	75.18±7.88
Education (yr)	16.44±2.62	16.78±2.67	16.07±2.62	16.38±2.81	15.86±2.75

4.2 Experimental Setting and Comparison Methods

In terms of experimental settings, we use a 10-fold cross-validation strategy, which equally divides the dataset into 10 points. Each time 9 sub-datasets are used as the training set, and the remaining 1 sub-dataset is used as the test set, so that the average value of 10 cycles is used as the final results. Regardint to the statistical standards of the model, we use the traditional accuracy (ACC), sensitivity (SEN), specificity (SPE), and the area under the receiver operating characteristic curve (AUC). For multi-task classification tasks, we first separately classify each task. Subsequently the federated learning is adopted to integrate all classification tasks, and then we compare the results with individual classification results. In addition, for the align process of the distributed data, we will show the changes in the data distribution during the experiment. Finally, the convergence of the proposed general framework algorithm is analyzed.

In this paper, we will compare the proposed model with traditional multi-task classification models, distributed multi-task classification models, and federated distributed multi-task classification models. Zhang and Shen [21] proposed a SVM-based ADNI multi-task classification model in 2012, which is a great improvement compared with the independent classification model. The distributed multi-task learning (DMTL) proposed by Wang et al. [12] uses a simple linear classification model for each task to integrate multiple tasks with group lasso constraints. Meanwhile, scholars have introduced two new subspace pursuit methods in the DMTL: the distributed greedy subspace pursuit (DGSP) [28] and the dual pursuit for subspace learning (DPSL) [29]. The two methods can effectively simplify the calculation process and speed up the convergence of the model, prematurely avoiding the model falling into a poor

local optimal solution. Smith et al. [30] introduced the novel federated learning into the distributed multi-task learning (FDMTL). Their work mainly solved the problem of the unbalanced data distribution and dealt with system challenges in federated learning. Simth et al. [30] proposed the federated multi-task learning (FMTL) to handle the statistical challenges and to set a novel system-aware optimization method.

Fig. 2. The classification results of our method and comparison methods in AD data: (a) accuracy, (b) sensitivity, (c) specificity, and (d) AUC.

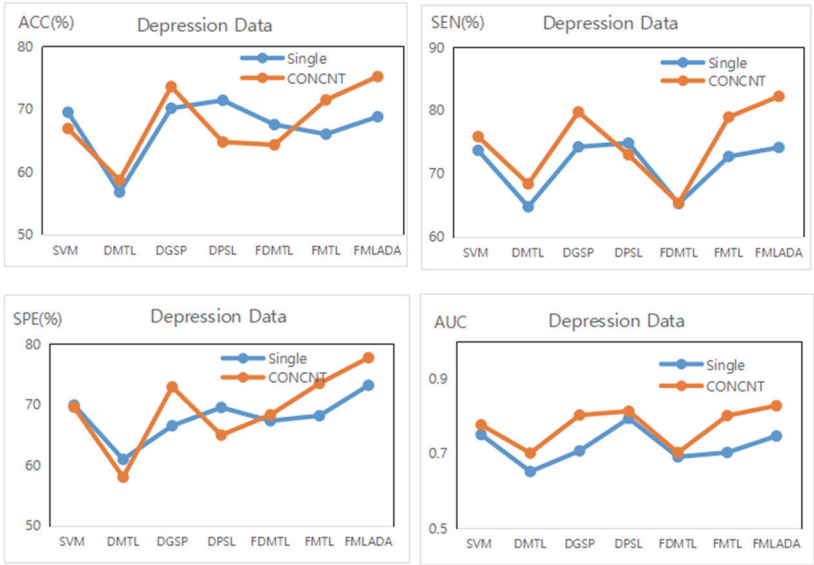

Fig. 3. The classification results of our method and comparison methods in depression data: (a) accuracy, (b) sensitivity, (c) specificity, and (d) AUC.

Figs. 2 and 3 show the classification results of popular comparison methods and our proposed model. Evidently, in a relatively small dataset, our proposed model has achieved the best performance. Table 2

reports the detailed information of UCI datasets. Table 3 present the classification results from state-of-the-art methods and our model in six UCI datasets. Benefit from the adaptive group sparsity adjustment constraint, we can accurately determine the degree of correlation between different data sources. Meanwhile, from Figs. 2 and 3, compared with the single-task model, the multi-task federated learning model exhibits a better classification effect and displays a better generalization ability. In the experimental operation stage, our model takes less time to obtain the convergence value. For approximate functions, the proposed general algorithm solution framework avoids non-convex calculations, and encourages the model to better obtain local optimal solutions. The comparison models based on deep model structures such as neural networks, cannot demonstrate the effectiveness in the situation where experimental data are more precious and scarce, and it is difficult for these models to obtain good convergence results in a short time.

Table 2. Notations of UCI databases

Datasets	Dimension	Number[a]
Heart	22	157/110
Parkinsons	22	147/48
Haberman	3	225/81
Diabetic	19	540/611
Breast	9	458/241

[a] The number of positive/negative samples.

Table 3. Results of six baselines and our model in 5 UCI datasets (unit: %)

Datasets		SVM	DMTL	DGSP	DPSL	FDMTL	FMTL	FMLADA
Heart	ACC	85.42±0.029	85.97±0.017	85.40±0.034	84.43±0.061	85.23±0.019	87.29±0.011	88.37±0.012
	AUC	83.13±0.031	86.22±0.014	84.11±0.053	81.89±0.009	81.26±0.054	76.47±0.019	87.43±0.055
Parkinsons	ACC	81.82±0.033	82.35±0.031	83.96±0.005	86.63±0.004	87.70±0.061	64.17±0.021	89.84±0.023
	AUC	81.09±0.049	80.43±0.012	79.84±0.009	82.36±0.017	83.02±0.032	50.23±0.028	85.54±0.046
Haberman	ACC	75.08±0.021	76.04±0.023	76.07±0.025	76.33±0.025	76.20±0.017	76.33±0.009	77.58±0.004
	AUC	67.10±0.054	73.18±0.035	74.32±0.063	74.22±0.013	71.11±0.031	50.85±0.019	74.47±0.036
Diabetic	ACC	91.41±0.033	91.20±0.059	91.48±0.064	91.25±0.029	91.41±0.019	90.33±0.015	91.53±0.055
	AUC	86.79±0.016	89.48±0.023	83.39±0.055	89.75±0.036	88.62±0.026	65.68±0.048	90.41±0.017
Breast	ACC	73.07±0.062	73.55±0.019	74.20±0.028	73.51±0.066	73.66±0.057	72.79±0.015	74.42±0.022
	AUC	63.54±0.011	67.32±0.015	65.91±0.023	68.33±0.001	68.48±0.068	61.47±0.039	68.52±0.003

5. Conclusion

In this paper, we propose a general FMTL model framework. The $l_{2,p}$ norm is used to adaptively constrain distributed data so that the sparsity of the multi-task model can be controlled during the federated learning process. However, for the non-convexity of the constraint norm, we adopt an approximation method to transform it into a convex function, and prove the rationality and superiority of this method. While ensuring the function convergence, our proposed method is able to minimize the error between the original function and the approximate function. The experimental results also verify the effectiveness. Compared with other popular models, our model achieves the best results, and the algorithm can also obtain an improved local solution in a limited number of iterations.

In future work, we will also explore massive multi-task datasets. Due to the expansion of datasets, the current model might be very slow and inefficient in mining the correlation between data. We will adopt a simple and effective feature extraction model to abstract all data, and the federated learning will guide the iterative optimization in each step of the model.

Acknowledgement

This work is supported by the Scientific Research Fund of Hunan Education Department (No. 19C0190 and 20C0218).

References

[1] J. Konecny, H. Brendan McMahan, F. X. Yu, A. T. Suresh, D. Bacon, and P. Richtarik, "Federated learning: strategies for improving communication efficiency," 2017 [Online]. Available: https://arxiv.org/abs/1610.05492.

[2] H. Brendan McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, Fort Lauderdale, FL, 2017, pp. 1273-1282.

[3] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, article no. 12, 2019. https://doi.org/10.1145/3298981

[4] H. Brendan McMahan, E. Moore, D. Ramage, and B. A. Y. Arcas, "Federated learning of deep networks using model averaging," 2016 [Online]. Available: https://arxiv.org/abs/1602.05629v1

[5] Y. Xue, X. Liao, L. Carin, and B. Krishnapuram, "Multi-task learning for classification with Dirichlet process priors," *Journal of Machine Learning Research*, vol. 8, pp. 35-63, 2007.

[6] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345-1359, 2010.

[7] X. T. Yuan, X. Liu, and S. Yan, "Visual classification with multitask joint sparse representation," *IEEE Transactions on Image Processing*, vol. 21, no. 10, pp. 4349-4360, 2012.

[8] L. Argote and E. Miron-Spektor, "Organizational learning: from experience to knowledge," *Organization Science*, vol. 22, no. 5, pp. 1123-1137, 2011.

[9] C. Vens, J. Struyf, L. Schietgat, S. Dzeroski, and H. Blockeel, "Decision trees for hierarchical multi-label classification," *Machine Learning*, vol. 73, no. 2, pp. 185-214, 2008.

[10] T. Evgeniou and M. Pontil, "Regularized multi-task learning," in *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Seattle, WA, 2004, pp. 109-117.

[11] S. Rosen, Z. Qian, and Z. M. Mao, "Appprofiler: a flexible method of exposing privacy-related behavior in android applications to end users," in *Proceedings of the 3rd ACM Conference on Data and Application Security and Privacy*, San Antonio, TX, 2013, pp. 221-232.

[12] J. Wang, M. Kolar, and N. Srerbo, "Distributed multi-task learning," in *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, Cadiz, Spain, 2016, pp. 751-760.

[13] R. Tibshirani, "Regression shrinkage and selection via the lasso: a retrospective," *Journal of the Royal Statistical Societ B*, vol. 73, no. 3, pp. 273-282, 2011.

[14] R. G. Brereton and G. R. Lloyd, "Support vector machines for classification and regression," *Analyst*, vol. 135, no. 2, pp. 230-267, 2010.

[15] J. Wright, A. Ganesh, S. Rao, and Y. Ma, "Robust principal component analysis: exact recovery of corrupted low-rank matrices via convex optimization," Coordinated Science Laboratory, University of Illinois, Urbana, IL, Report No. UILU-ENG-09-2210(DC-243), 2009.

[16] X. Ding, Y. Chen, Z. Tang, and Y. Huang, "Camera identification based on domain knowledge-driven deep multi-task learning," *IEEE Access*, vol. 7, pp. 25878-25890, 2019.
[17] D. Mateos-Nunez, J. Cortes, and J. Cortes, "Distributed optimization for multi-task learning via nuclear-norm approximation," *IFAC-PapersOnLine*, vol. 48, no. 22, pp. 64-69, 2015.
[18] M. Zhao, H. Zhang, W. Cheng, and Z. Zhang, "Joint l_p- and $l_{2,p}$-norm minimization for subspace clustering with outlier pursuit," in *Proceedings of 2016 International Joint Conference on Neural Networks (IJCNN)*, Vancouver, Canada, 2016, pp. 3658-3665.
[19] M. Zhang, Y. Yang, H. Zhang, F. Shen, and D. Zhang, "$L_{2,p}$-norm and sample constraint based feature selection and classification for AD diagnosis," *Neurocomputing*, vol. 195, pp. 104-111, 2016.
[20] R. Caruana, "Multitask learning," *Machine Learning*, vol. 28, no. 1, pp. 41-75, 1997.
[21] D. Zhang and D. Shen, "Multi-modal multi-task learning for joint prediction of multiple regression and classification variables in Alzheimer's disease," *NeuroImage*, vol. 59, no. 2, pp. 895-907, 2012.
[22] Z. Hu, B. Li, and J. Luo, "Time-and cost-efficient task scheduling across geo-distributed data centers," *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, no. 3, pp. 705-718, 2018.
[23] Y. Wang, M. Nikkhah, X. Zhu, W. T. Tan, and R. Liston, "Learning geographically distributed data for multiple tasks using generative adversarial networks," in *Proceedings of 2019 IEEE International Conference on Image Processing (ICIP)*, Taipei, Taiwan, 2019, pp. 4589-4593.
[24] X. Cai, F. Nie, H. Huang, and C. Ding, "Multi-class $l_{2,1}$-norm support vector machine," in *Proceedings of 2011 IEEE 11th International Conference on Data Mining*, Vancouver, Canada, 2011, pp. 91-100.
[25] P. Heins, M. Moeller, and M. Burger, "Locally sparse reconstruction using the $l^{1,\infty}$-norm," *Inverse Problems & Imaging*, vol. 9, no. pp. 1093-1137, 2015.
[26] P. E. Gill, W. Murray, and M. A. Saunders, "SNOPT: an SQP algorithm for large-scale constrained optimization," *SIAM Review*, vol. 47, no. 1, pp. 99-131, 2005.
[27] N. Tottenham, J. W. Tanaka, A. C. Leon, T. McCarry, M. Nurse, T. A. Hare, et al., "The NimStim set of facial expressions: judgments from untrained research participants," *Psychiatry Research*, vol. 168, no. 3, pp. 242-249, 2009.
[28] K. S. Kim and S. Y. Chung, "Greedy subspace pursuit for joint sparse recovery," *Journal of Computational and Applied Mathematics*, vol. 352, pp. 308-327, 2019.
[29] S. Yi, Y. Liang, Z. He, Y. Li, and Y. M. Cheung, "Dual pursuit for subspace learning," *IEEE Transactions on Multimedia*, vol. 21, no. 6, pp. 1399-1411, 2019.
[30] V. Smith, C. K. Chiang, M. Sanjabi, and A. Talwalkar, "Federated multi-task learning," 2017 [Online]. Available: https://arxiv.org/abs/1705.10467.

Shengbin Wu **https://orcid.org/0000-0001-7412-5007**

He has got Master of Computer Science, and graduated from Central South University in 2009. He worked in Changsha Medical University as a lecturer. His research interests include big data and information fusion.

Yibai Wang **https://orcid.org/0000-0001-5817-591X**

He has got Master of Computer Science, and graduated from Central South University in 2015. He worked in Changsha Medical University as a lecturer. His research interests include biological information, machine learning.