ISSN 1611-1699 / eISSN 2029-4433

VILNIUS TECH
Vilnius Gediminas Technical University

JOURNAL of BUSINESS ECONOMICS & MANAGEMENT

2025
Volume 26
Issue 3
Pages 555–575
https://doi.org/10.3846/jbem.2025.23768

# SMALL SAMPLE DATA PRICING RESEARCH BASED ON REPTILE ALGORITHM

Junxin SHEN[1], Yi YANG[1], Fanghao XIAO[2]✉

[1]*Faculty of Management and Economics, Kunming University of Science and Technology, Kunming, China*
[2]*Marxist College, Xiamen Institute of Technology, Xiamen, China*

**Article History:**
- received 10 June 2024
- accepted 18 March 2025

**Abstract.** Reasonably pricing data resources contributes to the development of the digital economy. However, in the early stage of the development of the data market, the data transaction volume is small, and the resulting small sample problem makes it difficult to accurately model and forecast the price of data. To address this issue, this paper constructs a meta-learning-based data price prediction model: MLP-Reptile. The model introduces a meta-learning tuning module to optimize the weight parameters of the base model, facilitating effective knowledge transfer learned from multiple tasks to enhance prediction accuracy in new tasks. Experimental results demonstrate that the proposed MLP-Reptile model excels in small sample data pricing tasks, outperforming other models. Additionally, the paper analyzes the different primary factors influencing data prices in various industries. The methods proposed in this research are universally applicable for addressing the small sample problem in data pricing, providing a reference for solving similar issues.

**Keywords:** data pricing, small sample, Bayesian optimization, meta-learning, Reptile algorithm, digital economy.

**JEL Classification:** D44, C10, D47, O33.

✉Corresponding author. E-mail: *xiaofanghao@xit.edu.cn*

## 1. Introduction

In the era of the digital economy (Rong, 2022), data, as a new factor of production (Jia et al., 2023; Hao et al., 2023a), is playing an increasingly important role in the production and distribution of the digital economy (You et al., 2022; Ye et al., 2022). In recent years, data markets have been continuously emerging, aiming to fully empower data circulation and effectively solve data silo issues, thereby promoting the openness and sharing of global data resources (Miao et al., 2021). Pricing data will further enhance the development and utilization of data (Hao et al., 2023b). Establishing effective data pricing models is of great significance to the economy and society.

Currently, there is a common problem of small samples in the data pricing process. In the context of diverse data itself and the continuous emergence of new data types, accurately pricing each type of data has become a pressing challenge. Data of specific types are generated in the early stages with relatively small volumes, and there are also some difficult-to-obtain data such as rare disease data (Li et al., 2023; Sun et al., 2024; Zhao et al., 2024), natural disaster data (Du et al., 2022; Ge et al., 2023; Weng & Paal, 2024), and data from specific scenarios (Li et al., 2024; Paeedeh et al., 2024). Although the available data for these types of information is limited, they are still valuable, and accurate pricing can promote the effective utilization of these data. Pricing models rely on a large number of data samples for modeling and analysis. However, in the field of data pricing, reliable deep learning models are still facing challenges due to the scarcity of existing transaction data. Therefore, it is urgent to find feasible methods that can make relatively accurate predictions even in the case of small samples.

In recent years, research on rapid learning with limited samples has continued to emerge. Among them, using meta-learning methods to train models has achieved significant learning effects. By learning shared features and knowledge across multiple tasks, meta-learning methods exhibit strong generalization capabilities on novel tasks and demonstrate significant potential in the development of deep learning-based regression models for small datasets (Lee & Yang, 2022).

This paper aims to analyze the primary factors influencing the pricing of data resources in small sample cases, with the establishment of a prediction model for accurately forecasting data resource prices in such scenarios as its objective. Firstly, employing Bayesian optimization to determine the optimal hyperparameters of the base Multilayer Perceptron (MLP) neural network, subsequently refining the model using the Reptile algorithm, and augmenting model performance through the incorporation of batch normalization layers and the addition of L2 regularization terms to the meta-objective of the Reptile algorithm.

The contributions of this study are summarized as follows:

1. The study comprehensively considers both intrinsic factors and market factors influencing data prices, analyzing the main characteristics affecting data resource prices in specific industries. This provides valuable insights for decision-makers in various industries regarding data resource management.
2. A novel small sample learning method based on the meta-learning Reptile algorithm is proposed for predicting data resource prices in situations with limited data samples. This marks the first application of meta-learning methods to address the small sample problem in the data resource pricing process.
3. The proposed model not only better utilizes limited sample data but also achieves more accurate pricing results for resources.

The structure of this paper is as follows: First, the introduction presents the significance of data pricing and the challenges posed by small sample sizes. The literature review section discusses the current research status and existing methods for data pricing, particularly focusing on small sample learning and meta-learning. The methodology section describes the construction of the MLP-Reptile model, including the Bayesian optimization process and the design of the Reptile algorithm. The experimental results and analysis section evaluates the model's performance using real-world data. Finally, the conclusion summarizes the findings, discusses the limitations of the study, and suggests directions for future research.

## 2. Research status

### 2.1. Small sample issue in data resource pricing

The marketization of data is still in its developmental stage, with relatively low trading volumes. From the perspective of the data itself, the continuous evolution of the social and economic environment has led to the emergence of new data types. For example, the rise of e-commerce has brought about online shopping data, while the widespread use of social media has led to the generation of social network data. New data categories are triggered by new demands, and the corresponding data is often collected in limited quantities in the initial stages. Additionally, some data types, due to their special nature, such as rare event data, rare disease data, data from specific populations, etc., are limited in quantity and are not generated frequently. Although these data may have small volumes, their accurate pricing is crucial for promoting data trading and circulation, thereby maximizing the value of data.

From an objective perspective, the scarcity of data transactions is determined by various factors. Some data trading platforms have single functionality, leading to insufficient on-platform trading volume, while transaction costs prompt data product supply and demand parties to seek off-platform transactions (Zhao et al., 2024). In the field of data trading, legal regulations and institutional frameworks are not perfect, and many internet data cannot be supplied due to issues such as unclear property rights, failing to meet market demand (Liu, 2021; Liao & Li, 2023). Data privacy and security issues (Li et al., 2020) are also important obstacles in data trading (Lv et al., 2021; Hao et al., 2023a). Due to the lack of clear data property rules and legal protection, data owners have low trust in data transactions and adopt a cautious attitude towards data trading, resulting in a limited amount of tradable data on data trading platforms. Additionally, there is a lack of consistent standards and interoperability among current data trading platforms, leading to insufficient market supply in the data market (Liu, 2021). These factors are summarized in Table 1, which outlines the key contributors to the low transaction volumes in data trading platforms.

**Table 1.** Factors contributing to low transaction volumes

| Factor classification | Impact factors |
|---|---|
| The intrinsic characteristics of the data | Limited data volume in the early stages of demand |
| | Difficulty in obtaining some data |
| Objective environment | Single functionality of data trading platforms (Zhao, 2022) |
| | Imperfect legal regulations and institutional frameworks (Liu, 2021; Demetzou et al., 2023; Guo et al., 2024) |
| | Need for strengthening data security and privacy protection (Yu & Zhao, 2019; Lv et al., 2021) |
| | Lack of consistent standards and interoperability among data trading platforms (Liu, 2021) |

## 2.2. Data pricing methods

Existing data pricing methods include traditional asset valuation methods, such as income approach, market approach, cost approach, property comprehensive evaluation method (a combination of subjective evaluation and objective quantification), economic methods (game theory, real options theory etc.), and other pricing methods. Among numerous pricing methods, intelligent algorithm evaluation has significant advantages over traditional algorithms in terms of nonlinear fitting capability, prediction accuracy, quantifiability, and computational efficiency. Machine learning methods have been proven to be well applied in the field of data pricing. Some scholars have proposed that the data pricing problem can be analogized to the multi-armed bandit problem in reinforcement learning, and applying the strategies proposed can bring good returns (Xu et al., 2016). The combination of machine learning methods and market models has been used to formulate optimal pricing schemes and has confirmed that the proposed model can achieve profit maximization (Niyato et al., 2016). With intelligent pricing methods as the research object, static and dynamic pricing methods and their practices have been studied (Tian & Wu, 2023). Data science and machine learning methods have been used to evaluate data rights, data pricing, and data privacy technologies (Xu et al., 2023). However, the above studies have not attempted to further improve the accuracy and applicability of predictions in the small sample problem.

### 2.3. Application of meta-learning methods in small sample problems

High-precision data models can improve decision efficiency and thus create greater economic value. Therefore, the accuracy of data models is closely related to the value of the models. Deep learning typically requires a large amount of labeled data to support it. However, many real-world applications face challenges such as difficulty in obtaining labeled data and high processing costs for model training (Yanik et al., 2022, 2024). Traditional model optimization methods are difficult to apply effectively when training samples are severely insufficient, leading to difficulties in model optimization. In the case of small samples, models are prone to overfitting, and updates of a small number of parameters may not enable the network to learn feature representations with strong generalization capabilities. Therefore, ensuring that machine learning models can quickly learn from small sample data and improve generalization capabilities has become a practical problem that must be addressed.

To address this problem, research on small sample learning continues to emerge. Small sample learning aims to circumvent the serious performance degradation of traditional machine learning methods when sample data is insufficient, using fewer sample data to construct machine learning models that can solve practical problems. Among them, the use of meta-learning methods for model training has achieved significant learning effects, and meta-learning can be used to improve performance in data scarce scenarios (Minot & Reddy, 2024; Vo et al., 2024). Finn et al. (2017) proposed a model-agnostic meta-learning method called Model-agnostic Meta Learning (MAML). This method is both independent of the model's structure and does not introduce new parameters, and it is compatible with any model trained using gradient descent. In 2018, Nichol et al. (2018) proposed simplifying computations in the Model-Agnostic Meta-Learning (MAML) framework by replacing the second-order parameter derivatives with first-order derivatives. They modified the initialization parameter update rules in the MAML algorithm and introduced the Reptile algorithm. This algorithm directly utilizes data from individual small-sample tasks for parameter updates, thereby enhancing network speed. The Reptile method has achieved good results in some small sample studies (Tian et al., 2021). This study adopts the Reptile algorithm to address the small sample data pricing problem.

## 3. MLP-Reptile small sample data pricing model

This paper first uses Bayesian optimization as part of the proposed model to optimize the hyperparameters of the MLP model, enabling more effective learning across different pricing tasks. Subsequently, the Reptile algorithm is introduced. The model learns from multiple different pricing tasks to acquire prior knowledge, enabling more accurate interpretation of the variation of the dependent variable and capturing the intrinsic patterns and complex relationships of the data, thereby better utilizing limited samples for generalization (Zhang et al., 2022), and demonstrating superior performance in new tasks. The application of meta-learning methods such as Reptile has demonstrated significant performance improvements in small-sample tasks across various fields (Nichol et al., 2018), including spatiotemporal prediction with limited data (Tian et al., 2021), few-shot short-term wind power forecasting (Chen et al., 2024). These findings align with our model's design goals to address the data pricing challenges posed by limited training samples.

## 3.1. Basic neural network model based on Bayesian optimization

The value of hyperparameter directly affects the performance and prediction effect of MLP model. Bayesian algorithms can help find the optimal combination of hyperparameters for the model. Specifically, Bayesian optimization evaluates the most promising hyperparameters in each iteration, taking into account the training results of previous tasks, thereby enabling the model to adapt more quickly to new tasks. We choose Gaussian Process (GP) as the surrogate model and employs Expected Improvement (EI) as the acquisition function to enhance the algorithm's performance in terms of efficiency and accuracy. Taking the search for the maximum value as an example, the optimal combination of hyperparameters can be represented as:

$$x_{opt} = \arg\max_{x \in X} f(x) . \tag{1}$$

In the Equation:

$x$ represents the combination of hyperparameters for the MLP neural network.

$X$ denotes the set of hyperparameters.

$f(x)$ represents the mapping from the combination of hyperparameters to the model's generalization performance.

$x_{opt}$ represents the optimal combination of hyperparameters.

## 3.2. Design of the Reptile algorithm

The focus of parameter optimization-based strategies lies in learning a good optimizer parameter $\theta_0$ to better optimize existing parameters $\theta$. Using well-initialized parameters $\theta_0$ can effectively reduce computational costs, allowing the model's existing parameters to converge more quickly towards the target direction.

The MAML algorithm is used to implement update rules, and meta-learning optimizes deep neural network parameters based on the following objectives (Equations (1) and (2)) (Finn et al., 2017):

$$\arg\min_{\theta} \sum_{T_i} L_{T_i}(f_{\theta_i'}) , \tag{2}$$

where $L$ is the loss function of the deep neural network, $f$ representing a neural network with parameters $\theta$. $\theta'$ represents the parameters updated through gradient descent with an internal learning rate $\alpha$. The update rule is as follows: $\theta_i' \leftarrow \theta - \alpha \nabla_\theta L_{T_i}(f_{\theta_i})$.

Equation (1) involves the gradient update rule, which includes the double gradient step:

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_{T_i} L_{T_i}(f_{\theta - \alpha \nabla_\theta L_{t_i}(f_{\theta_i})}) , \tag{3}$$

where $\beta$ is the outer learning rate, The double gradient step enables the training steps of $T_i$ to be applicable to other tasks.

However, the computation cost and complexity associated with the second-order derivatives in the double gradient step are considerable. To address these issues, Nichol et al. (2018) proposed the Reptile algorithm, which iteratively updates the parameter space of the base model. The Reptile algorithm enables rapid adjustment and adaptation across different tasks. Specifically, it randomly samples a data pricing task and utilizes its training set to train the base pricing model. Then, using gradient descent, it iteratively adjusts the model parameters on this task. In this way, the model can learn features and knowledge relevant to the current task.

First, using the mean squared error loss function $L_{MSE}$, the parameters $\theta$ of the deep neural network, and the gradient step size $s$ for stochastic gradient descent, the following update rule $U$ is adopted:

$$U : W \leftarrow \theta - \eta \nabla L_{MSE}(\theta) , \tag{4}$$

where $W$ represents the intermediate weights of the deep neural network after the gradient step, $\eta$ is the learning rate, and $\nabla$ is the gradient operator. Then, multiple steps of stochastic gradient descent are performed, updating $U'$ for $s$ times with stochastic gradients, where the gradient step size is determined by the batch training size $B_M$ and the number of training steps $E_M$, balancing the accuracy and efficiency of the network by adjusting the step size.

After completing a single task, the model parameters are updated to reflect the outer learning rate $\beta$, as well as the difference between the previous weights and the SGD result corresponding to $W - \theta$, as follows:

$$\theta \leftarrow \theta + \beta(W - \theta) = \theta + \beta(U^S(\theta) - \theta) , \tag{5}$$

where $U^S$ represents the updated parameters for $s$ iteration according to Equation (3). A crucial point is that the training process described above needs to be repeated across multiple pricing tasks. By iteratively performing training and parameter adjustments on different tasks, the model can learn shared information and patterns among tasks and apply them to new tasks.

## 3.3. Improved MLP-Reptile model

### *3.3.1. Optimization of model structure*

In small sample tasks, due to the limited number of training samples, models are more prone to overfitting. The model may memorize the details of the training samples but fail to generalize to the entire data distribution. Due to the unique nature of small sample tasks, the model needs to make multiple attempts and experiments from limited samples to quickly learn and determine suitable initialization strategies. Meta-learning algorithms may take a long time to converge to the optimal solution. The MLP model is capable of flexibly modeling complex nonlinear relationships, adapting to the characteristics of different data structures, and capturing high-order dependencies, making it suitable for limited data points and high-dimensional feature spaces in small sample studies.

L2 regularization and Batch Normalization have been widely demonstrated in the literature to reduce overfitting and accelerate model convergence (Hoerl & Kennard, 1970; Ng, 2004; Santurkar et al., 2018; Bjorck et al., 2018; Ioffe & Szegedy, 2015). In the architecture configuration of the MLP model, this paper adopts the method of integrating batch normalization layers. Batch normalization is added after each fully connected layer to alleviate issues related to weight initialization sensitivity, find appropriate weight initialization faster, promote a more stable optimization process, and reduce the trend of overfitting. By normalizing the inputs of each batch, batch normalization reduces the problems of vanishing and exploding gradients, making the model easier to learn and converge in fewer iterations, thereby improving training speed. The detailed structure of the model is presented in Table 2, which outlines the specific layer configuration of the MLP model used in this study.

**Table 2.** Layer configuration of the MLP model

| Layer Name | Type | Input size | Output size | Activation function |
|---|---|---|---|---|
| fc1 | Linear | input_dim | 10 | ReLU |
| relu1 | Activation | – | – | ReLU |
| fc2 | Linear | 10 | 32 | ReLU |
| relu2 | Activation | – | – | ReLU |
| fc3 | Linear | 32 | output_dim | – |

### *3.3.2. Regularization constraint*

The Reptile meta-learning algorithm achieves rapid learning on new tasks through training on a small number of samples. However, due to the limited training data, there is a risk of overfitting during the meta-training process, resulting in lower prediction accuracy of the model on new tasks. L2 regularization is a commonly used technique to prevent overfitting, which limits the model complexity by adding a penalty term to the model parameters.

To reduce overfitting of the network during meta-training and improve the model's predictive ability on new tasks during meta-testing, this study adds an L2 regularization term to the meta-objective of the Reptile algorithm, resulting in a new meta-objective. Here, $\lambda$ is a hyperparameter ranging from 0 to 1. According to the meta-objective, the update formula for $\theta$ is as shown in Equation (6):

$$\theta \leftarrow \theta + \beta(W - \theta) = \theta + \beta(U^S(\theta) - \theta) + \frac{\lambda}{2} \| \theta \|^2 . \tag{6}$$

### *3.3.3. MLP-Reptile*

The MLP-Reptile model updates its parameters across multiple pricing tasks using the improved Reptile algorithm, gradually approaching the parameters of each task through weighted averaging. In the pre-training phase, Bayesian optimization is used for hyperparameter tuning. Subsequently, the MLP possesses model parameter states, exhibiting good overall performance across all tasks in the training set. Using this model for regression prediction on the dataset enables the model to learn general patterns and features, facilitating easier adaptation to new tasks in subsequent meta-learning phases.

During the fine-tuning phase, the model weights transferred from the pre-training phase are fine-tuned to the target small sample tasks. L2 regularization and a small sample loss function are integrated during this process to prevent overfitting during adaptation to small sample tasks. The desired output at this stage is an optimized MLP regressor capable of accurately predicting pricing tasks. Specifically, the steps of the Reptile algorithm are as shown in Table 3.

**Table 3.** Steps of the Reptile algorithm

| Step | Task |
|---|---|
| 1 | Randomly select a task from the training task set as the current task. |
| 2 | Perform multiple steps of stochastic gradient descent (SGD) on the current task s times to obtain temporary parameters. |
| 3 | Update the model parameters based on the difference between the previous model parameters and the results of multiple-step SGD. |
| 4 | Repeat the above steps iteratively to update the model parameters. |

Figure 1 depicts the MLP-Reptile framework proposed in this paper. The Reptile algorithm aims to learn initial parameters that can adapt to any new task, achieving better performance on new pricing tasks by considering the similarities among multiple pricing tasks. Firstly, random task sampling is performed on the dataset to obtain n task sets, enabling the model to quickly adapt to new tasks by learning shared information and structures among different pricing tasks. Then, the MLP model is chosen as the base model, and the Reptile algorithm updates the parameters of the base model on each task. After training on n tasks is completed, the new parameters are updated, resulting in a new data pricing model.

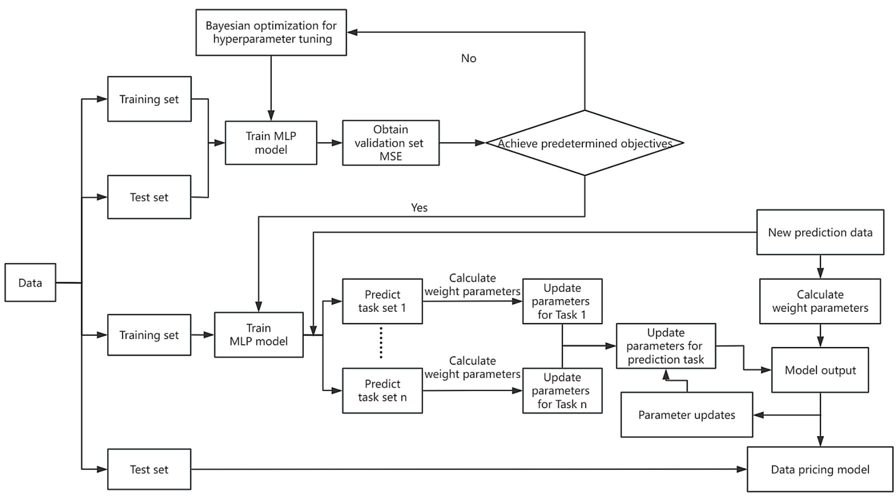


**Figure 1.** Meta-learning optimization framework for prediction models

## 4. Experimental results and analysis

### 4.1. Data source and preprocessing

#### *4.1.1. Data source*

The data for this study is sourced from Guoxin Youyi Data (n.d.). Guoxin Youyi Data Company, initiated by the National Information Center, is a technology platform-oriented enterprise focusing on next-generation information technologies such as big data, artificial intelligence, blockchain, and the Internet of Things. Youyi Data (n.d.) provides data trading resources across multiple industries through its data marketplace platform.

The required data for this research is obtained from this platform using web crawling technology, considering both intrinsic factors and market dynamics influencing prices. Taking into account the quantifiability and availability of data, nine dimensions of intrinsic data factors are selected based on the current standard issued by the National Standards Committee (2018), "Data Trading Service Platform Transaction Data Description Standard" (GB/T 36343-2018).

The evaluation algorithm not only reflects intrinsic data factors but also considers market feedback factors. Additionally, it considers dimensions formed when data are traded as commodities and compared with similar types of goods, including scarcity score, applicability score, timeliness score, and sales volume. The platform manages each score based on both rule attributes and application scope dimensions to enhance matching accuracy.

By employing network analysis, a dynamic weight allocation scoring model is formed, ensuring reliability in the scoring process. These key factors influencing data prices are summarized in Table 4.

**Table 4.** Factors influencing data prices

| Factor classification | Factors | data type | Corresponding dimensions from GB/T 36343—2018 |
|---|---|---|---|
| Intrinsic factors | Consistency | integer | Data quality |
| | Structurization | integer | |
| | Rating | integer | |
| | Completeness | integer | |
| | Data size | numerical | Data scale |
| | Quantity | integer | |
| | Information redundancy | integer | Data uniqueness |
| | Data category | character | Industry classification |
| | Data labels | character | |
| Market factors | Scarcity | integer | – |
| | Applicability | integer | |
| | Timeliness | integer | |
| | Sales volume | integer | |

#### *4.1.2. Preprocessing*

##### *4.1.2.1. Outlier handling*

Machine learning models are typically constructed based on the statistical characteristics and distribution of data, and outliers may deviate from the normal distribution and trend of data, reducing the model's fit to the data. According to the principle of box plot, the upper bound and lower bound of the column data are first calculated. The upper bound is the third quartile (75th percentile) plus 1.5 times the interquartile range (IQR), and the lower bound is the first quartile (25th percentile) minus 1.5 times the IQR. Then, data beyond the upper and lower bounds are removed to eliminate outliers.

#### *4.1.2.2. One-hot encoding*

The dataset features include categorical fields such as "data category" and "data tags." "Data category" includes 10 categories such as public opinion monitoring, industrial economy, scientific research and technology, precision marketing, and traffic geography, while "data tags" include 56 data labels such as stocks, e-commerce, COVID-19, social, scientific research data, and comprehensive enterprises. One-Hot encoding transforms these categorical variables into numerical forms directly usable by machine learning algorithms. Each category is transformed into a new binary feature column, with each data entry assigned a 1 (belonging to that category/tag) or 0 (not belonging to that category/tag), facilitating correct interpretation and use of these features by the model.

#### *4.1.2.3. Target variable processing*

The original dataset's price ranges from 0 to 1,500,000 yuan, with 75% of prices at 889.5 yuan or below, showing a right-skewed distribution. However, many models assume that data are normally distributed, and they perform best when data follow a normal distribution. Box-Cox adjusts the form of transformation by introducing a parameter (lambda) to make the data distribution closer to a normal distribution, reducing or eliminating heteroscedasticity and improving the reliability of the model.

## 4.2. Evaluation metrics

In this paper, mean square error (MSE), root mean square error (RMSE), mean absolute error (MAE) and coefficient of determination ($R^2$) are selected to evaluate the performance of the model on the test set. Smaller values of MSE, RMSE, and MAE indicate more accurate model predictions. The closer the value of $R^2$ is to 1, the closer the model's prediction is to the actual result. The formulas for the above evaluation metrics are as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 ; \tag{7}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} ; \tag{8}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| ; \tag{9}$$

$$R^2 = 1 - \frac{\sum_i (\hat{y}_i - y_i)^2}{\sum_i (\bar{y}_i - y_i)^2} . \tag{10}$$

## 4.3. MLP-Reptile model evaluation

### *4.3.1. Local interpretation based on LIME*

LIME (Local Interpretable Model-agnostic Explanations) is a method used to explain the prediction results of machine learning models. LIME works by using an interpretable model to generate localized explanations centered around a single prediction. This capability gives LIME a high degree of versatility and adaptability, allowing it to provide explanations not only for popular models such as deep learning neural networks, random forests, and gradient boosting but also for any other credible machine learning model (Ali et al., 2023). Due to this characteristic, it is referred to as "model-agnostic."

The primary advantage of LIME's model-agnostic nature is its ability to integrate seamlessly with a wide range of models, even when dealing with complex predictions in high-dimensional feature spaces. LIME produces locally faithful explanations that offer insights into how individual features influence model predictions (Ribeiro et al., 2016). This characteristic is particularly critical in small-sample scenarios, where traditional global interpretability methods may fail due to limited data variability.

In our study, LIME's focus on predictions allows for a better understanding of data pricing in specific industries, ensuring that the model's decisions are transparent and actionable. LIME focuses on the explanation of individual samples, which can be especially important in small sample datasets, as the dataset itself may be small and there may be significant differences between samples. LIME generates new data instances near the sample and interprets these instances to provide a local explanation of the model's predictions.

In the context of small sample datasets, LIME's local explanations help to reveal how the features of specific data points affect the prediction results, enabling us to better understand the model's decision-making process at the individual level. LIME is used to explain the features that affect data prices in three representative industries. As shown in Figure 2, for the price of data resources in the three industries of sentiment analysis (Figure 2a), precision marketing (Figure 2b) and financial credit assessment (Figure 2c), the size, application value, quantity, consistency and sales volume of data resources are important factors affecting the model's pricing of data resources. These factors not only reflect the attributes of the data resources themselves but also encompass comprehensive influences from industry demand, market conditions, and data quality.

Take the Sentiment Analysis industry data as an example to analyze the interpretation results of LIME, as shown in the following Table 5.

Specifically, the data size has the most significant positive impact on the price, indicating that a larger amount of data can provide more comprehensive information, thereby enhancing its market value. Data with low application value and quantity have a negative impact on price, indicating that these data with poor characteristics have lower demand and price in the market. In addition, the degree of structure, consistency and timeliness have a positive impact on the price of data, indicating that the improvement of these characteristics can increase the utilization value and market recognition of data. These interpretation results help us understand how much the model depends on different features in the pricing decision process, and provide guidance for data providers to improve data quality and market competitiveness.

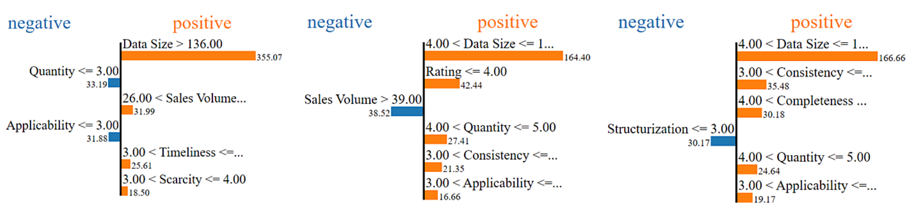


**Figure 2.** Local Explanations of LIME: a – sentiment analysis; b – precision marketing; c – financial credit assessment

**Table 5.** LIME explanation for sentiment analysis data

| Feature | Impact value | Description |
|---|---|---|
| Data size | positive impact: 356.43 | Larger data volume provides more comprehensive information, increasing its value. |
| Applicability | negative impact: 46.39 | When the application value score is less than or equal to 3, the data's demand and price in the market are lower. |
| Quantity | negative impact: 25.44 | Data with lower quantity scores have lower value. |
| Structurization | positive impact: 29.24 | Data with a structuring degree score between 3 and 5 have higher utilization value. |
| Consistency | positive impact: 17.31 | Data with consistency scores between 4 and 5 have higher accuracy and utilization value. |
| Timeliness | positive impact: 16.68 | Data with timeliness scores between 3 and 4 have greater market value and are more desirable when updated promptly. |

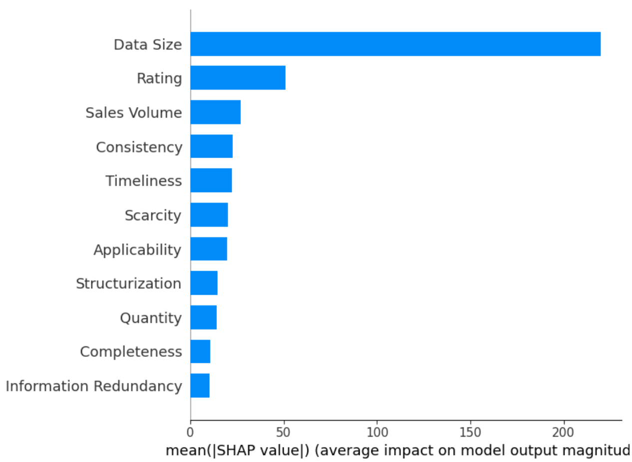


**Figure 3.** Feature importance analysis chart based on SHAP values

SHAP (SHapley Additive exPlanations) is a widely used method for interpreting machine learning models (Lundberg & Lee, 2017). It is based on the Shapley value from game theory, assigning contribution values to each feature to explain the model's output. This approach quantifies the impact of individual features on model predictions, thereby revealing the key features the model relies on when making decisions. In feature importance analysis, SHAP values provide a global perspective in terms of averages, illustrating the overall influence of each feature on prediction outcomes. We used the SHAP method to calculate the SHAP values for all features and analyzed their global impact on the model's predictions. Figure 3 illustrates the feature importance based on SHAP values, where dataset size has the highest average impact on prediction outcomes, followed by Rating, indicating that these features play a significant role in the model.

#### 4.3.2. Optimization effect

The proposed MLP-Reptile model was run on a standard laptop equipped with an Intel Core i7-10700 CPU and 16GB of memory. The entire training process took approximately 621.78 seconds without GPU acceleration. Despite the limited hardware conditions, the model demonstrated high computational efficiency, further highlighting its applicability in small-sample scenarios with constrained computational resources. Furthermore, to address the computational demands of larger datasets, future work could explore distributed computing or model simplification and optimization techniques.

##### 4.3.2.1. Optimization effect on overall industry data

To verify the effectiveness of the MLP-Reptile model, this study compares it with six methods, including classical machine learning models: Linear Regression (LR), K-Nearest Neighbors (KNN), Support Vector Regression (SVR), ensemble model: Adaptive Boosting (AdaBoost), and models based on different base models and meta-learning algorithms: MLP-MAML (MLP model optimized by MAML algorithm), RNN-Reptile (RNN model optimized by Reptile algorithm), CNN-Reptile (CNN model optimized by Reptile algorithm), and MLP-Reptile (the proposed method).

Traditional model optimization methods are difficult to apply effectively when training samples are severely lacking. However, meta-learning frameworks show great potential in developing deep learning-based regression models using small datasets (Lee & Yang, 2022). As shown in Table 6, the results show that the MLP-Reptile model outperforms other models in handling small sample problems. By training on multiple relevant tasks, the MLP-Reptile model enables the model to better adapt to data characteristics when facing new tasks, improving the method's stability and generalization ability. This provides insight that pre-trained weights and optimization of the objective function help accelerate the adaptation process during small sample training. This observation further highlights the advantages of the proposed model in small sample problems.

**Table 6.** Comparison of evaluation metrics between Reptile algorithm and traditional machine learning models (Training Set Size: 3800)

| | LR | KNN | SVR | AdaBoost | MLP-MAML | RNN-Reptile | CNN-Reptile | MLP-Reptile |
|---|---|---|---|---|---|---|---|---|
| MSE | 4.1906 | 6.5117 | 5.4992 | 4.2208 | 4.1551 | 4.0400 | 4.2959 | 3.7497 |
| RMSE | 2.0471 | 2.5518 | 2.3450 | 2.0545 | 2.0384 | 2.0100 | 2.0726 | 1.9364 |
| MAE | 1.5213 | 1.8884 | 1.5339 | 1.7304 | 1.4978 | 1.5030 | 1.5225 | 1.4313 |
| R2 | 0.6474 | 0.4521 | 0.5373 | 0.6449 | 0.6472 | 0.6570 | 0.6353 | 0.6816 |

##### 4.3.2.2. Optimization effect on industry-specific data

The demand for data resources in various industries is constantly changing, and the challenge of small-sample prediction becomes more prominent and severe when new types of data resource demands emerge in emerging industries. On the Guoxin Youyi (n.d.) trading platform, the "Precision Marketing", "Financial Credit Assessment", "Sentiment Analysis", and "Research and Technology" industries have only 85, 602, 1450, and 1477 data points, respectively. However, by training on small-sample data in specific industries, it is possible to more accurately capture the unique characteristics and trends of that industry, thereby improving the accuracy and effectiveness of the model in practical applications and providing more reliable decision support for different industries. Therefore, experiments were conducted on data from various industries to assess the model's performance. As shown in Table 7 below, the proposed model performs well on small sample data of these industries.

**Table 7.** $R^2$ values for industry-specific data

| | data volume | LR | KNN | SVR | AdaBoost | MLP-Reptile |
|---|---|---|---|---|---|---|
| Precision marketing | 85 | 80.47% | 18.94% | 38.10% | 85.12% | 86.70% |
| Financial credit assessment | 602 | 19.76% | –11.24% | 5.11% | 60.89% | 61.72% |
| Sentiment analysis | 1450 | 19.74% | –13.16% | 3.00% | 33.24% | 37.77% |
| Research and technology | 1477 | 97.33% | 89.59% | 94.65% | 96.88% | 97.11% |

#### *4.3.3. Ablation experiment*

In this section, we conduct ablation experiment to validate the effectiveness of the two improvements proposed in this paper for MLP-Reptile (adding batch normalization layers to the MLP model and improving the meta-objective function). The results are shown in Table 8.

**Table 8.** MSE of disintegration experiment for MLP-Reptile model

| Model | MSE |
|---|---|
| MLP + Reptile | 4.0276 |
| MLP + Reptile + BN | 3.8235 |
| MLP + Reptile + L2 | 3.8328 |
| MLP + Reptile + BN + L2 | 3.7497 |

In Table 8, "MLP + Reptile" represents the MLP model optimized by the Reptile algorithm without further modifications, using the updating strategy described in preceding Section; "MLP + Reptile + BN" denotes the model with an additional batch normalization layer, using the same original updating strategy; "MLP + Reptile + L2" indicates the model optimized by the Reptile algorithm with added L2 regularization; "MLP + Reptile + BN + L2" refers to the model with both an added batch normalization layer and L2 regularization, using the updating strategy described in preceding Section. The meta-learning iterations are set to 300, with an internal loop of 3 steps, a learning rate of 0.1 for updating model parameters, a batch size of 256 for the internal loop, and an L2 regularization weight of 0.001. During the training phase, the training set size is 3800, and during the testing phase, the test set size is 1200.

As shown in Figure 4, the experimental results indicate that the unoptimized MLP-Reptile model has a longer training time (light green line) and is prone to overfitting (dark blue line). After optimization, models with added batch normalization layers, improved Reptile algorithm's meta-objective function, and both batch normalization layers and improved Reptile algorithm's meta-objective function show improvements in model efficiency. Particularly, the model with both added batch normalization layers and the inclusion of a regularization term in the meta-objective function demonstrates the most significant enhancement in prediction efficiency (light green line) and the shortest training time (deep red line). The introduction of batch normalization layers, as proposed in Section 3.3.1, aids in quickly learning initial parameters of the model, thus leading to better performance on specific new tasks. Additionally, the method proposed in Section 3.3.2 of incorporating an L2 regularization term into the Reptile algorithm's meta-objective function also enhances the model's performance.

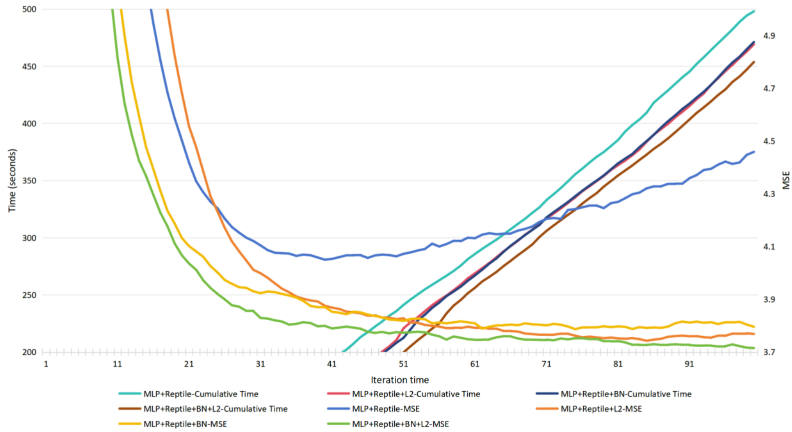


**Figure 4.** Training duration and MSE values of fusion experiment model

### *4.3.4. Effectiveness of MLP-Reptile model on different sizes of training sets*

This study employed the MLP-Reptile model to train on support sample sets of various sizes (200, 400, 800, 1800, 2800, 3800) and evaluated it on a test set of size 1200. The key experimental comparison results are shown in Table 9. By comparing the optimized MLP-Reptile model with the MLP model, it can be observed that the model trained with meta-learning exhibits a significant decrease in loss values compared to the pre-meta-learning model, particularly evident in low-sample scenarios. Notably, in the case of a sample size of 400, this improvement is particularly significant, with an increase in $R^2$ exceeding 10%. This not only confirms the effectiveness of the proposed optimization method in training with small samples but also highlights the potential to enhance model accuracy with limited data.

**Table 9.** Improvement effect of MLP-Reptile model on different size training sets

| | Before optimization | After optimization | Improvement ratio | Before optimization | After optimization | Improvement ratio |
|---|---|---|---|---|---|---|
| Training set: 200, Testing set: 1200 | | | | Training set: 400, Testing set: 1200 | | |
| MSE | 18.2989 | 14.3890 | 21.37% | 5.8876 | 5.2672 | 10.54% |
| RMSE | 4.2777 | 3.7933 | 11.32% | 2.4264 | 2.2950 | 5.42% |
| MAE | 3.3698 | 3.2388 | 3.89% | 1.9310 | 1.8024 | 6.66% |
| R2 | –0.5396 | –0.2106 | 60.97% | 0.5047 | 0.5569 | 10.34% |
| Training set: 800, Testing set: 1200 | | | | Training set: 1800, Testing set: 1200 | | |
| MSE | 5.0239 | 4.6319 | 7.80% | 4.5574 | 4.2177 | 7.45% |
| RMSE | 2.2414 | 2.1522 | 3.98% | 2.1348 | 2.0537 | 3.80% |
| MAE | 1.6873 | 1.6205 | 3.96% | 1.6318 | 1.5093 | 7.51% |
| R2 | 0.5773 | 0.6103 | 5.71% | 0.6166 | 0.6451 | 4.64% |
| Training set: 2800, Testing set: 1200 | | | | Training set: 3800, Testing set: 1200 | | |
| MSE | 4.3190 | 3.9421 | 8.73% | 4.2411 | 4.0100 | 5.45% |
| RMSE | 2.0782 | 1.9855 | 4.46% | 2.0594 | 2.0025 | 2.76% |
| MAE | 1.5621 | 1.4797 | 5.28% | 1.5292 | 1.5066 | 1.48% |
| R2 | 0.6366 | 0.6683 | 4.98% | 0.6432 | 0.6626 | 3.02% |

However, As shown in Figure 5, with the increase of sample set size, the advantages of the MLP-Reptile model gradually decrease. On larger sample sets (such as 1800, 2800, and 3800), the gap between the MLP-Reptile model and the model optimized without the Reptile algorithm decreases. On smaller sample sets, the Reptile algorithm can better utilize the information from the training tasks to help the model adapt to new tasks. However, on larger sample sets, the sample size itself is sufficient for the model to better learn task information, and the potential for performance improvement is relatively limited regardless of whether the Reptile algorithm is used for optimization. It can be seen that the MLP-Reptile model demonstrates more significant effects on smaller sample sets. Therefore, the MLP-Reptile model can better leverage meta-learning on small sample sets, addressing the challenges posed by limited data and exhibiting more prominent effects. This observation emphasizes the need to balance the applicability of different algorithms when selecting sample sizes. For small sample scenarios, the MLP-Reptile model may be a more effective choice, while in large sample scenarios, the improvement in model performance may depend more on the scale of the sample size itself.

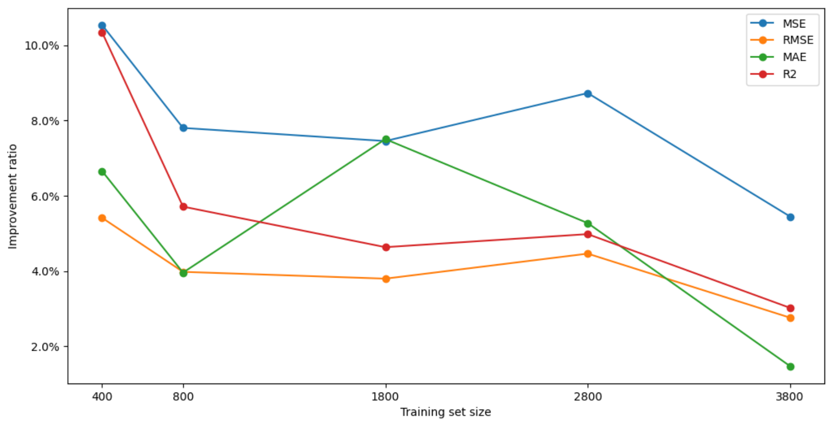


**Figure 5.** Improvement ratio of evaluation metrics before and after meta-learning on different size training sets

Our findings are consistent with recent research that applies meta-learning techniques, particularly the Reptile algorithm, to small-sample scenarios. Minot and Reddy (2024) demonstrated the effectiveness of Reptile in antibody engineering under noisy and insufficiently labeled data conditions, which is similar to the challenges we face in data pricing. Moreover, Paeedeh et al. (2024) confirmed the success of Reptile in cross-domain few-shot learning, highlighting the ability of meta-learning to generalize across different tasks with limited data. Zhang et al. (2022) showed that Reptile outperformed traditional learning algorithms in multi-agent systems, underscoring its adaptability in small-data settings. Additionally, in the field of computer vision, these results, along with our findings, emphasize the versatility and effectiveness of the Reptile algorithm in improving model generalization for small-sample problems.

### *4.3.5. Multi-platform validation*

To demonstrate the generalization ability and practicality of the model, we evaluated it using additional datasets from two platforms: JD Wanxiang (JD Cloud, n.d.) and Tianyuan Data (n.d.). These datasets differ slightly in features and size, providing a broader perspective to test the robustness and adaptability of the proposed method (Table 10).

**Table 10.** Data volume and features of Tianyuan Data and JD Wanxiang Platforms

| Platform | Tianyuan Data | JD Wanxiang |
|---|---|---|
| Data volume | 120 | 798 |
| Features | Price, pageviews, data type, data format, data size, update frequency/collection time, supplier, tags, product category, etc. | Price, rating, transaction volume, pageviews, favorites, data size, data type, data format, source, start time, end time, category, etc. |

The results from the two platforms are summarized in Table 11.

**Table 11.** Model prediction performance on JD-Wanxiang and Tianyuan Data

| Metrics | MSE | RMSE | MAE | $R^2$ |
|---|---|---|---|---|
| JD Wanxiang | 15.51 | 3.94 | 3.12 | 0.67 |
| Tianyuan Data | 7.76 | 2.79 | 2.03 | 0.54 |

Despite differences in feature sets and data volume, the model achieved reasonable predictive performance on both datasets, confirming its cross-platform adaptability and robustness. The JD Wanxiang dataset offers a richer feature set, which may account for the higher $R^2$ value and more comprehensive predictions. This underscores the importance of including diverse relevant features in data pricing models. In contrast, the Tianyuan dataset is relatively smaller in size and feature diversity, potentially limiting the model's ability to capture complex patterns, as reflected in its lower $R^2$ value. The multi-platform validation highlights the generalizability of the proposed approach, ensuring its applicability across different datasets and platforms.

## 5. Conclusions

Fair and accurate pricing of data is essential for facilitating their circulation in the data trading market, thereby maximizing their value. Due to the presence of the small sample problem, accurately modeling and predicting data resource prices become challenging, necessitating the exploration of methods to accurately model and predict even in small sample scenarios. This study proposes the MLP-Reptile model to address the small sample data resource pricing problem. Our approach aggregates data resource information and related price data from multiple industries to improve generality and prediction accuracy. Evaluation on real data resource confirms the superior performance of our proposed predictive model.

The main contributions are as follows:

1. Analysis of the main factors influencing data prices in specific industries under small sample conditions. The study identifies factors such as data size, applicability value, completeness, and consistency as the main common factors influencing prices. The primary influencing factors vary across different industries, providing references for data pricers from various industries. Considering factors such as consistency and completeness can help products achieve higher selling prices.
2. This paper provides an innovative solution to the small sample data resource pricing problem by proposing the MLP-Reptile model. Traditional machine learning methods typically require a large amount of labeled data to train models, learning the mapping

relationship from input to output. Instead, we introduce the Reptile algorithm, aiming to learn common features and patterns between tasks rather than the specific details of each task, requiring minimal labeled data. By training on multiple related tasks, the model adapts better to new tasks, significantly reducing prediction errors and providing an effective method for small sample data pricing problems. The following are the advantages of the proposed model:

Consideration of multidimensional factors: Comprehensive consideration of the influence of both intrinsic data factors and market factors on data resource prices.

Accurate pricing in small sample scenarios: Capable of accurately pricing data resources, with unique advantages over other models in small sample scenarios.

Adaptation to different industry characteristics: Experimental results on data from different industries demonstrate the proposed model's ability to accurately adapt to the unique requirements of each industry and predict data prices accurately, offering practicality and personalized adaptability, providing reliable support for decision-making in various industries.

Efficiency: The operation of adding batch normalization layers to the MLP model and improving the meta-objective function reduces model training time, achieving better prediction results in a shorter time.

Even in small sample scenarios, data pricers can optimize pricing strategies, maximizing the value of data resources by accurately and effectively pricing and understanding the factors influencing prices. Theoretically, our study not only enriches the application research of small-sample learning and meta-learning but also fills gaps in the existing literature, further advancing the theoretical development of this field. Practically, this research has broad application prospects. The proposed MLP-Reptile model is not only applicable to data pricing problems but can also be applied to various fields such as financial forecasting and medical data analysis, providing enterprises and organizations with a reliable data analysis tool in small-sample scenarios. Furthermore, by optimizing training efficiency, this study offers a practical solution for machine learning model training in real-world applications, enabling efficient training under limited computational resources and enhancing the accuracy and efficiency of decision support.

This study primarily focuses on modeling the small-sample problem in the early stages of data market development, where data transaction volumes are low and features are sparse. Therefore, although we did not conduct experiments on large-scale datasets, the preliminary results demonstrate that the model performs well in small-sample scenarios. However, to evaluate the model's scalability and performance boundaries, future research will aim to test it on larger-scale datasets to further validate its applicability and robustness. Other known small sample solution methods can be attempted to address small sample data resource pricing problems, such as model fine-tuning, data augmentation, metric learning, etc. Combining different methods can further enhance the ability to solve small sample data resource pricing problems. Furthermore, future studies should consider the complexities introduced by real-world data trading environments, such as dynamic market conditions and regulatory changes. Integrating these factors will make the model more flexible and practical in real-world applications.

## Funding

This work was supported by the National Natural Science Foundation of China under Grant 71964018; Applied Basic Research Key Project of Yunnan under Grant 202401AS070112.

## Disclosure statement

This research does not have any competing financial, professional, or personal interests from other parties.

## References

Ali, S., Akhlaq, F., Imran, A. S., Kastrati, Z., Daudpota, S. M., & Moosa, M. (2023). The enlightening role of explainable artificial intelligence in medical & healthcare domains: A systematic literature review. *Computers in Biology and Medicine*, *166*, Article 107555. https://doi.org/10.1016/j.compbiomed.2023.107555

Bjorck, N., Gomes, C. P., Selman, B., & Weinberger, K. Q. (2018). Understanding batch normalization. *Advances in Neural Information Processing Systems*, *31*, 1–12.

Chen, F., Yan, J., Liu, Y., Yan, Y., & Tjernberg, L. B. (2024). A novel meta-learning approach for few-shot short-term wind power forecasting._ *Applied Energy*, *362*, Article 122838. https://doi.org/10.1016/j.apenergy.2024.122838

Demetzou, K., Zanfir-Fortuna, G., & Barros Vale, S. (2023). The thin red line: Refocusing data protection law on ADM, a global perspective with lessons from case-law. *Computer Law & Security Review*, *49*, Article 105806. https://doi.org/10.1016/j.clsr.2023.105806

Du, X., Li, X., Zhang, S., Zhao, T., Hou, Q., Jin, X., & Zhang, J. (2022). High-accuracy estimation method of typhoon storm surge disaster loss under small sample conditions by information diffusion model coupled with machine learning models. *International Journal of Disaster Risk Reduction*, *82*, Article 103307. https://doi.org/10.1016/j.ijdrr.2022.103307

Finn, C., Abbeel, P., & Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. *Proceedings of the 34th International Conference on Machine Learning*, *70*, 1126–1135.

Ge, J., Tang, H., Yang, N., & Hu, Y. (2023). Rapid identification of damaged buildings using incremental learning with transferred data from historical natural disaster cases. *ISPRS Journal of Photogrammetry and Remote Sensing*, *195*, 105–128. https://doi.org/10.1016/j.isprsjprs.2022.11.010

Guoxin Youyi (n.d.). https://infobase.deeprocknetzero.com/detail/449?

Guo, Z., Hao, J., & Kennedy, L. (2024). Protection path of personal data and privacy in China: Moving from monism to dualism in civil law and then in criminal law. *Computer Law & Security Review*, *52*, Article 105928. https://doi.org/10.1016/j.clsr.2023.105928

Hao, J., Deng, Z., & Li, J. (2023a). The evolution of data pricing: From economics to computational intelligence. *Heliyon*, *9*(9), Article e20274. https://doi.org/10.1016/j.heliyon.2023.e20274

Hao, J., Yuan, J., Li, J., Liu, M., & Liu, Y. (2023b). Ensemble pricing model for data assets with ranking-pruning-averaging strategy. *Procedia Computer Science*, *221*, 813–820. https://doi.org/10.1016/j.procs.2023.08.055

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, *12*(1), 55–67. https://doi.org/10.1080/00401706.1970.10488634

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, *37*, 448–456.

Jia, W., Collins, A., & Liu, W. (2023). Digitalization and economic growth in the new classical and new structural economics perspectives. *Digital Economy and Sustainable Development*, *1*, Article 5. https://doi.org/10.1007/s44265-023-00007-0

JD Cloud. (n.d.). *JD Wanxiang Data Platform*. https://wx.jdcloud.com/

Liu, J. (2021). Development practice of data factor market in China: Achievements, problems and suggestions. *Price: Theory & Practice*, (12), 18–22.

Lee, J., & Yang, C. (2022). Deep neural network and meta-learning-based reactive sputtering with. *Journal of Manufacturing Systems*, *62*, 703–717. https://doi.org/10.1016/j.jmsy.2022.02.004

Li, L., Fan, Y., Tse, M., & Lin, K.-Y. (2020). A review of applications in federated learning. *Computers & Industrial Engineering*, *149*, Article 106854. https://doi.org/10.1016/j.cie.2020.106854

Li, X., Chen, K., Yang, J., Wang, C., Yang, T., Luo, C., Li, N., & Liu, Z. (2023). TLDA: A transfer learning based dual-augmentation strategy for traditional Chinese Medicine syndrome differentiation in rare disease. *Computers in Biology and Medicine*, *169*, Article 107808. https://doi.org/10.1016/j.compbiomed.2023.107808

Li, X., Su, H., Xiang, L., Yao, Q., & Hu, A. (2024). Transformer-based meta learning method for bearing fault identification under multiple small sample conditions. *Mechanical Systems and Signal Processing*, *208*, Article 110967. https://doi.org/10.1016/j.ymssp.2023.110967

Liao, J., & Li, R. (2023). Establishing a two-way transaction pricing model of “platform-individual” co-creation data property rights. *Journal of Innovation & Knowledge*, *8*(4), Article 100427. https://doi.org/10.1016/j.jik.2023.100427

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, *30*, 1–10.

Lv, Z., Qiao, L., Hossain, M., & Choi, B. (2021). Analysis of using blockchain to protect the privacy of drone big data. *IEEE Network*, *35*(1), 44–49. https://doi.org/10.1109/MNET.011.2000154

Miao, X., Gao, Y., Chen, L., Peng, H., Yin, J., & Li, Q. (2021, April 19–22). Towards query pricing on incomplete data. In *Proceedings of the 2021 IEEE 37th International Conference on Data Engineering* (*ICDE*) (pp. 2348–2349). Chania, Greece. IEEE. https://doi.org/10.1109/ICDE51399.2021.00260

Minot, M., & Reddy, S. T. (2024). Meta learning addresses noisy and under-labeled data in machine learning-guided antibody engineering. *Cell Systems*, *15*(1), 4–18. https://doi.org/10.1016/j.cels.2023.12.003

National Standards Committee. (2018, June 07). *Information technology Data Transaction Service transaction data Description* (Standard No. GB/T 36343-2018). https://openstd.samr.gov.cn/bzgk/gb/newGbInfo?hcno=445371F13A0DED107EF13579D018ACEA

Ng, A. Y. (2004). Feature selection, L1 vs. L2 regularization, and rotational invariance. In *Proceedings of the Twenty-first International Conference on Machine Learning* (p. 78). Association for Computing Machinery. https://doi.org/10.1145/1015330.1015435

Nichol, A., Achiam, J., & Schulman, J. (2018). *On first-order meta-learning algorithms*. Arxiv. https://doi.org/10.48550/arXiv.1803.02999

Niyato, D., Alsheikh, M. A., Wang, P., Kim, D. I., & Han, Z. (2016 May 22–27). Market model and optimal pricing scheme of big data and Internet of Things (IoT). In *2016 IEEE International Conference on Communications* (*ICC*) (pp. 1–6). Kuala Lumpur, Malaysia. IEEE. https://doi.org/10.1109/ICC.2016.7510922

Paeedeh, N., Pratama, M., Ma’sum, M. A., Mayer, W., Cao, Z., & Kowlczyk, R. (2024). Cross-domain few-shot learning via adaptive transformer networks. *Knowledge-Based Systems*, *288*, Article 111458. https://doi.org/10.1016/j.knosys.2024.111458

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). “Why should I trust you?”: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1135–1144). Association for Computing Machinery. https://doi.org/10.1145/2939672.2939778

Rong, K. (2022). Research agenda for the digital economy. *Journal of Digital Economy*, *1*(1), 20–31. https://doi.org/10.1016/j.jdec.2022.08.004

Santurkar, S., Tsipras, D., Ilyas, A., & Madry, A. (2018). How does batch normalization help optimization? In *Proceedings of the 32nd International Conference of Neural Information Processing Systems* (pp. 2488–2498). Curran Associates Inc.

Sun, J., Wei, D., Wang, L., & Zheng, Y. (2024). Hybrid unsupervised representation learning and pseudo-label supervised self-distillation for rare disease imaging phenotype classification with dispersion-aware imbalance correction. *Medical Image Analysis*, *93*, Article 103102. https://doi.org/10.1016/j.media.2024.103102

Tian, C., Zhu, X., Hu, Z., & Ma, J. (2021). A transfer approach with attention reptile method and long-term generation mechanism for few-shot traffic prediction. *Neurocomputing*, *452*, 15–27. https://doi.org/10.1016/j.neucom.2021.03.068

Tian, G., & Wu, W. (2023). Big data pricing in marketplace lending and price discrimination against repeat borrowers: Evidence from China. *China Economic Review*, *78*, Article 101944. https://doi.org/10.1016/j.chieco.2023.101944

Tianyuan Data. (n.d.). https://www.tdata.cn/

Vo, S.-N., Vo, T.-T., & Le, B. (2024). Interpretable extractive text summarization with meta-learning and BI-LSTM: A study of meta learning and explainability techniques. *Expert Systems with Applications*, *245*, Article 123045. https://doi.org/10.1016/j.eswa.2023.123045

Weng, Y., & Paal, S. G. (2024). Extrapolating wind pressures on roof soffits of low-rise buildings using few-shot learning. *Journal of Building Engineering*, *82*, Article 108110. https://doi.org/10.1016/j.jobe.2023.108110

Xu, J., Hong, N., Xu, Z., Zhao, Z., Wu, C., Kuang, K., Wang, J., Zhu, M., Zhou, J., Ren, K., Yang, X., Lu, C., Pei, J., & Shum, H. (2023). Data-driven learning for data rights, data pricing, and privacy computing. *Engineering*, *25*, 66–76. https://doi.org/10.1016/j.eng.2022.12.008

Xu, L., Jiang, C., Qian, Y., Zhao, Y., Li, J. & Ren, Y., (2016). Dynamic privacy pricing: A multi-armed bandit approach with time-variant rewards. *IEEE Transactions on Information Forensics and Security*, *12*(2), 271–285. https://doi.org/10.1109/TIFS.2016.2611487

Yanik, E., Intes, X., Kruger, U., Yan, P., Diller, D. Van Voorst, B., Makled, B., Norfleet, J., & De, S. (2022). Deep neural networks for the assessment of surgical skills: A systematic review. *The Journal of Defense Modeling and Simulation*, *19*(2), 159–171. https://doi.org/10.1177/15485129211034586

Yanik, E., Schwaitzberg, S., Yang, G., Intes, X., Norfleet, J., Hackett, M., & De, S. (2024). One-shot skill assessment in high-stakes domains with limited data via meta learning. *Computers in Biology and Medicine*, *174*, Article 108470. https://doi.org/10.1016/j.compbiomed.2024.108470

Ye, Y., Zhou, A., Shi, X., & Huang, C. (2022). A SEED model for constructing the data factor market: Evidence from Guiyang Global Big Data Exchange (GBDEx) in China. *Journal of Digital Economy*, *1*(3), 273–283. https://doi.org/10.1016/j.jdec.2023.03.002

You, J., Lou, S., Mao, R., & Xu, T. (2022). An improved FMEA quality risk assessment framework for enterprise data assets. *Journal of Digital Economy*, *1*(3), 141–152. https://doi.org/10.1016/j.jdec.2022.12.001

Yu, B., & Zhao, H. (2019). Research on the construction of big data trading platform in China. In *Proceedings of the 2019 4th International Conference on Intelligent Information Technology* (pp. 107–112). Association for Computing Machinery. https://doi.org/10.1145/3321454.3321474

Zhang, X., Hu, C., He, B., & Han, Z. (2022). Distributed reptile algorithm for meta-learning over multi-agent systems. *IEEE Transactions on Signal Processing*, *70*, 5443–5456. https://doi.org/10.1109/TSP.2022.3222734

Zhao, H., Qiu, S., Bai, M., Wang, L., & Wang, Z. (2024). Toxicity prediction and classification of Gunqile-7 with small sample based on transfer learning method. *Computers in Biology and Medicine*, *173*, Article 108348. https://doi.org/10.1016/j.compbiomed.2024.108348