Multi-Fractional Gradient Descent: A Novel Approach to Gradient Descent for Robust Linear Regression

ROBAB KALANTARI, KHASHAYAR RAHIMI, SAMAN NADERI MEZAJIN
Finance Department
Khatam University
Tehran
IRAN

Abstract: This work introduces a novel gradient descent method by generalizing the fractional gradient descent (FGD) such that instead of the same fractional order for all variables, we assign different fractional orders to each variable depending on its characteristics and its relation to other variables. We name this method Multi-Fractional Gradient Descent (MFGD) and by using it in linear regression for minimizing loss function (residual sum of square) and apply it on four financial time series data and also tuning their hyperparameters, we can observe that unlike GD and FGD, MFGD is robust to multicollinearity in the data and also can detect the real information in it and obtain considerable lower error.

Key-Words: multicollinearity-gradient descent-fractional gradient descent -Multi-Fractional Gradient Descent-Fractional calcules

Received: April 2, 2024. Revised: August 4, 2024. Accepted: September 9, 2024. Published: October 30, 2024.

1 Introduction

Fractional Calculus is concerned with the study of fractional order integral and derivative operators over real or complex domains, as well as their applications. Its origins may be traced back to a letter from de l'Hospital to Leibniz in 1695. Questions like ”What does Fractional Derivative mean?” , for instance, ”What does the derivative of order $\frac{1}{4}$ or $\sqrt{3}$ of a function mean?” encouraged many talented scientists to concentrate their efforts on this issue. In the 18th and 19th centuries. For example, consider [1], [2], [3], [4], [5], [6], [7],[8], [9], [10], [11], [12], [13]. From a mathematical perspective, we have found many interesting publications in the last decayed related to applications of classical fixed point theorems on abstract spaces to study the existence and uniqueness of solutions to various types of initial value problems and boundary value problems for fractional operators (See, e. example., [14] [15], [16], [17], [18], [19], [20] and on the other hands there are also many applications to different science for example [21], [22], [23], [24], [25],[26], [27].

The field of machine learning and fractional calculus have each independently played vital roles in understanding and modeling complex real-life phenomena. Machine learning has emerged as a powerful tool for extracting patterns and behaviors from historical data, making it a cornerstone in various scientific disciplines, while fractional calculus provides a unique framework for describing complex dynamics with non-integer-valued derivatives. Fractional derivatives, which originated from a 17th-century inquiry into the concept of non-integer orders, have become essential in capturing the memory and inherent non-local behavior of systems. As these two modern-day topics hold substantial potential for synergistic approaches in modeling complex dynamics, this introduction sets the stage for a broader exploration of their combined potential.

Fractional calculus, often associated with its applications in physics, image processing, environmental sciences, and even biology, introduces the concept of memory into modeling processes. The fractional order of a process is closely tied to the degree of memory exhibited by that process, making it particularly relevant in fields where historical context and spatiotemporal memory are key considerations. As researchers continue to uncover the utility of fractional derivatives in modeling complex natural phenomena, it is evident that these derivatives provide valuable tools to enhance machine learning approaches [28]. The integration of machine learning and fractional calculus is a burgeoning field of research, with several recent papers exploring their combined potential. Fractional calculus, which involves derivatives and integrals of non-integer order, is gaining attention due to its ability to model complex dynamics and phenomena in various fields. Machine learning, on the other hand, is a powerful tool for data analysis and prediction. The combination of these two fields can lead to more accurate and powerful models.

A recent review paper titled ”Combining Fractional Derivatives and Machine Learning: A Review" discusses the potential of combining approaches from fractional derivatives and machine learning. The paper categorizes past combined approaches into three categories: preprocessing, machine learning and fractional dynamics, and optimization. The contributions of fractional derivatives to machine learning are manifold as they provide powerful preprocessing and feature augmentation techniques, can improve physically informed machine learning, and are capable of improving hyperparameter optimization[29]

Another paper titled "Efficient Machine Learning and Factional Calculus Based Mathematical Model for Early COVID Prediction" discusses the use of fractional calculus-based models for disease prediction and detection. The paper highlights that fractional calculus has non-local memory characteristics, which makes function approximation more accurate. The authors combined mathematical models based on fractional calculus with machine learning models for early estimation of COVID spread[30].

A paper titled "Machine Learning of Space-Fractional Differential Equations" discusses the benefits of implementing fractional derivatives. The paper highlights that fractional derivatives allow for discovering fractional-order PDEs for systems characterized by heavy tails or anomalous diffusion. The paper also mentions that a single fractional-order archetype allows for a derivative of arbitrary order to be learned, with the order itself being a parameter in the regression[31].

In the paper "Fractional differentiation and its use in machine learning", the authors discuss the implementation of fractional (non-integer order) differentiation on real data of four datasets based on stock prices. The paper concludes that fractional differentiation plays an important role and leads to more accurate predictions in the case of artificial neural networks[32].

In summary, the combination of machine learning and fractional calculus is a promising area of research that can lead to more accurate and powerful models. The use of fractional calculus in machine learning can provide powerful preprocessing and feature augmentation techniques, improve physically informed machine learning, and enhance hyperparameter optimization.

This work serves as the initial step in unraveling the synergy between Gradient Descent Algorithm as part of machine learning and fractional calculus with stochastic nature. In section 2, we introduce Gradient Descent Algorithm then in section 3, we explain the fractional Calculus and FGD. In section 4 we define MFGD. In numerical experiment we show the result of implementing the main idea on four finance time series data(Gold, S& P500, NASDAQ, and Dowjones) set and compare three methods (Gradient Descent, FGD, and MFGD) with classical linear regression and represent how MFGD to find informative feature. And at last, in section 6 we get a conclusion and shortly speak about future challenges and works.

2 Gradient Descent Algorithm in Linear Regression

In its simplest form, a linear regression model can be expressed as:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p + \varepsilon$$

The cost function for linear regression is often defined as the mean squared error (MSE), which measures the average squared difference between the predicted ($\hat{Y}_i$) and actual (Y_i) values:

$$J(\beta) = \frac{1}{2N} \sum_{i=1}^{N} (\hat{Y}_i - Y_i)^2$$

where N is the number of observations.

The goal of gradient descent is to minimize this cost function by adjusting the coefficients (β).

The gradient descent algorithm starts with an initial guess for the coefficients (β) and iteratively updates them by moving in the direction of the steepest decrease in the cost function. The update rule for the coefficients is given by:

$$\boldsymbol{\beta} = \boldsymbol{\beta} - \alpha \nabla J(\boldsymbol{\beta})$$

where: α is the learning rate, a small positive constant that determines the step size, $\nabla J(\boldsymbol{\beta})$ is the gradient vector of the cost function with respect to the coefficients.

The components of the gradient vector are the partial derivatives of the cost function with respect to each coefficient:

$$\nabla J(\boldsymbol{\beta}) = \left[\frac{\partial J(\boldsymbol{\beta})}{\partial \beta_0}, \frac{\partial J(\boldsymbol{\beta})}{\partial \beta_1}, \ldots, \frac{\partial J(\boldsymbol{\beta})}{\partial \beta_p} \right]^T$$

$$\frac{\partial J(\boldsymbol{\beta})}{\partial \beta} = -\frac{1}{N} \mathbf{X}^T (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$$

The algorithm continues this process until convergence, where the changes in the coefficients become negligible, or a predefined number of iterations is reached.

2.1 Advantages and Considerations

Gradient descent offers several advantages in the context of linear regression:

1. Scalability: It is particularly useful for large datasets as it processes data in small batches, making it computationally efficient.

2. Flexibility: It can be applied to a wide range of cost functions, not limited to the mean squared error.

However, the choice of the learning rate is crucial. A learning rate that is too small may result in slow convergence, while a learning rate that is too large may cause overshooting or divergence. Additionally, the cost function should be convex to ensure that gradient descent converges to the global minimum.
In summary, gradient descent is a powerful optimization algorithm applied to linear regression, enabling the model to find optimal coefficients efficiently and effectively.

3 Fractional Calculus

In this section, we'll explain Fractional Gradient Descent and more information about fractional derivative such as Riemann–Liouville fractional derivative, Caputo's fractional derivatives, Grünwald–Letnikov derivative see[33].
Theorem 1. [33]Let (c, d), $-\infty < c < d < +\infty$, be an open interval in $\mathbb{R}$, and $[a, b] \subset (c, d)$ be such that for each $t \in [a, b]$ the closed ball $B_{b-a}(t)$, with center at t and radius $b - a$, lies in (c, d). If $x(\cdot)$ is analytic in (c, d), then

$$ {}_aD_t^{\alpha} X(t) = \sum_{k=0}^{\infty} \frac{(-1)^{k-1} \alpha x^{(k)}(t)}{k!(k-\alpha)\Gamma(1-\alpha)} (t-a)^{k-\alpha}. $$

3.1 Fractional Gradient Descent

In various research studies, the utilization of fractional derivatives in optimization has been explored, particularly in the formulation of a gradient vector that incorporates fractional partial derivatives.[34] To be more precise, fractional gradient descent in linear regression can be defined as follows:

$$ \nabla^{\alpha} J(\boldsymbol{\beta}) = \left[{}_aD_{\beta_0}^{\alpha} J(\boldsymbol{\beta}), {}_aD_{\beta_1}^{\alpha} J(\boldsymbol{\beta}), \dots, {}_aD_{\beta_p}^{\alpha} J(\boldsymbol{\beta}) \right]^T $$

$$ \boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^k - \gamma \nabla^{\alpha} J(\boldsymbol{\beta}^k) $$

Corollary 1. Riemann–Liouville fractional derivative of $J(\boldsymbol{\beta})$ can be calculated by the finite terms of Theorem 1 approximation.

Proof. Since there is a integer-order derivative in that approximation, by calculating it we have:

$$ \frac{\partial^0 J(\boldsymbol{\beta})}{\partial \beta^0} = \frac{1}{2N} (\mathbf{Y} - \mathbf{X}\beta)^T (\mathbf{Y} - \mathbf{X}\beta) $$
$$ \frac{\partial^1 J(\boldsymbol{\beta})}{\partial \beta^1} = -\frac{1}{N} \mathbf{X}^T (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) $$
$$ \frac{\partial^2 J(\boldsymbol{\beta})}{\partial \beta \partial \beta^T} = \frac{1}{N} \mathbf{X}^T \mathbf{X} $$

And obviously for higher partial orders $i > 2$

$$ \frac{\partial^i J(\boldsymbol{\beta})}{\partial \beta^i} = 0 $$

Therefore

$$ {}_aD_{\beta}^{\alpha} J(\boldsymbol{\beta}) = \sum_{k=0}^{2} \frac{(-1)^{k-1} \alpha}{k!(k-\alpha)\Gamma(1-\alpha)} \frac{\partial^k J(\boldsymbol{\beta})}{\partial \beta^k} (t-a)^{k-\alpha}. $$

□

4 Multi-Fractional Gradient Descent

Different studies shows that fractional gradient descent performs better than standard gradient descent in different senses like model accuracy, CPU/GPU time consumption and number of iteration for convergence.
These improvements stem from changing strict one-order gradient to more flexible fractional-orders. So, a natural idea for generalizing fractional gradient descent with order α for all p partial derivative is to dedicate different fractional orders to each of them with respect to their natures and other properties that are influenced by the order of fractional derivative.
For mathematical formulation we can define the multi-fractional gradient descent as follows:

$$ \nabla^{\mathcal{A}} J(\boldsymbol{\beta}) = \left[{}_aD_{\beta_0}^{\alpha_0} J(\boldsymbol{\beta}), {}_aD_{\beta_1}^{\alpha_1} J(\boldsymbol{\beta}), \dots, {}_aD_{\beta_p}^{\alpha_p} J(\boldsymbol{\beta}) \right]^T $$

$$ \boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^k - \gamma \nabla^{\mathcal{A}} J(\boldsymbol{\beta}^k) $$

Where $\mathcal{A} = (\alpha_0, \alpha_1, \dots, \alpha_p)$.
We can generalize this definition even more by changing the starting point a with p different points and forming the following gradient vector:

$$ \nabla^{\mathcal{A}, \mathbb{A}} J(\boldsymbol{\beta}) = \left[{}_{a_0}D_{\beta_0}^{\alpha_0} J(\boldsymbol{\beta}), {}_{a_1}D_{\beta_1}^{\alpha_1} J(\boldsymbol{\beta}), \dots, {}_{a_p}D_{\beta_p}^{\alpha_p} J(\boldsymbol{\beta}) \right]^T $$

Where $\mathbb{A} = (a_0, a_1, \dots, a_p)$.

5 Numerical Experiment

To assess the effectiveness of the suggested approach, which involves using multi-fractional gradient descent, we conducted experiments on four financial datasets: S&P 500, Dow Jones, NASDAQ, and Gold. These datasets encompass diverse sets of features. The data spans daily closing prices from January 1, 2000, to January 1, 2023.

The dataset was divided into training and test sets, with over 56 percent allocated for training purposes. Specifically, the first 3000 records of the data were used for training, and the rest were reserved for testing. This partitioning strategy allows for robust evaluation and validation of the proposed framework's performance on the financial datasets.

5.1 Multi-Fractional Gradient Descent is Robust to Multicollinearity

We designed an experiment to investigate the performance of multi-fractional gradient descent in Linear Regression model in dealing with multicollinearity in the data. For this we add the first to the tenth lag of four mentioned financial time series and evaluate the classic, fractional and multi-fractional gradient descent Linear Regression models on each of them and at last comparing their error rates.

It is obvious that there as we have shown in the following plots,unlike classic LR, and somehow fractional version, our Multi-Fractional model is perfectly robust to the multicollinearity and its performance does not ruined by increasing collinear features and in some cases it gets better(lower) mean absolute error rate. In the presented heatmaps, it is evident that the correlation scores between each pair of variables exceed 0.99, indicating a high degree of correlation between these variables. This observation prompted us to investigate the performance of three gradient descent algorithms applied to each of the four time series. The ensuing analysis includes the assessment of error rates. Remarkably, our proposed model demonstrates robustness in the face of collinearity, showcasing consistent performance even when new correlated lags are introduced to the dataset. Notably, the standard fractional gradient descent outperforms the classical gradient descent approach, emphasizing its superiority in handling complex relationships within the data. Despite this enhanced performance, it is worth noting that the standard fractional gradient descent remains susceptible to the challenges posed by collinearity, presenting a nuanced trade-off between performance and sensitivity to correlated variables.

The outcomes of the aforementioned experiment are presented below. It is essential to highlight that each of the three models underwent a tuning process to optimize their hyperparameters, ensuring the attainment of the best possible performance for each model.

5.2 Multi-Fractional Gradient Descent Detects the Information

In the next numerical experiment, we delve into assessing the effectiveness of our proposed MFGD LR model, highlighting its superior performance relative to the other two models. The datasets comprises four financial time series: S&P 500, Gold, Dow Jones, and NASDAQ. Notably, within the domain of financial time series analysis, the extraction of meaningful features presents a substantial challenge. It is essential to underscore that feature extraction falls outside the scope of this work. The datasets for each financial instrument are constructed with the following features:

1. **Fractional_diff_lag1:** Fractional Difference at Lag 1 - A transformation applied to time series data to enhance stationarity and capture long-term dependencies, promoting stability in the dataset.

   In time series analysis, fractional differencing is a technique used to enhance stationarity by applying fractional differences to a time series. The fractional difference operator, denoted as Δ^d, is defined as follows:

   For a given time series $\{X_t\}$, the d-th fractional difference is calculated as:

$$\begin{aligned}(1-B)^d &= \sum_{k=0}^{\infty} \binom{d}{k} (-B)^k \\ &= \sum_{k=0}^{\infty} \frac{\prod_{i=0}^{k-1}(d-i)}{k!} (-B)^k \\ &= 1 - dB + \frac{d(d-1)}{2!} B^2 \\ &- \frac{d(d-1)(d-2)}{3!} B^3 + \dots\end{aligned}$$

   The parameter d determines the degree of differencing. When d is an integer, fractional differencing reduces to the regular differencing operator.

   Fractional differencing is particularly useful for capturing long-term dependencies in time series data. It allows for a flexible approach to achieving stationarity without relying on traditional differencing methods.

2. **WMA (Weighted Moving Average):** Weighted Average - Averages time series data with higher

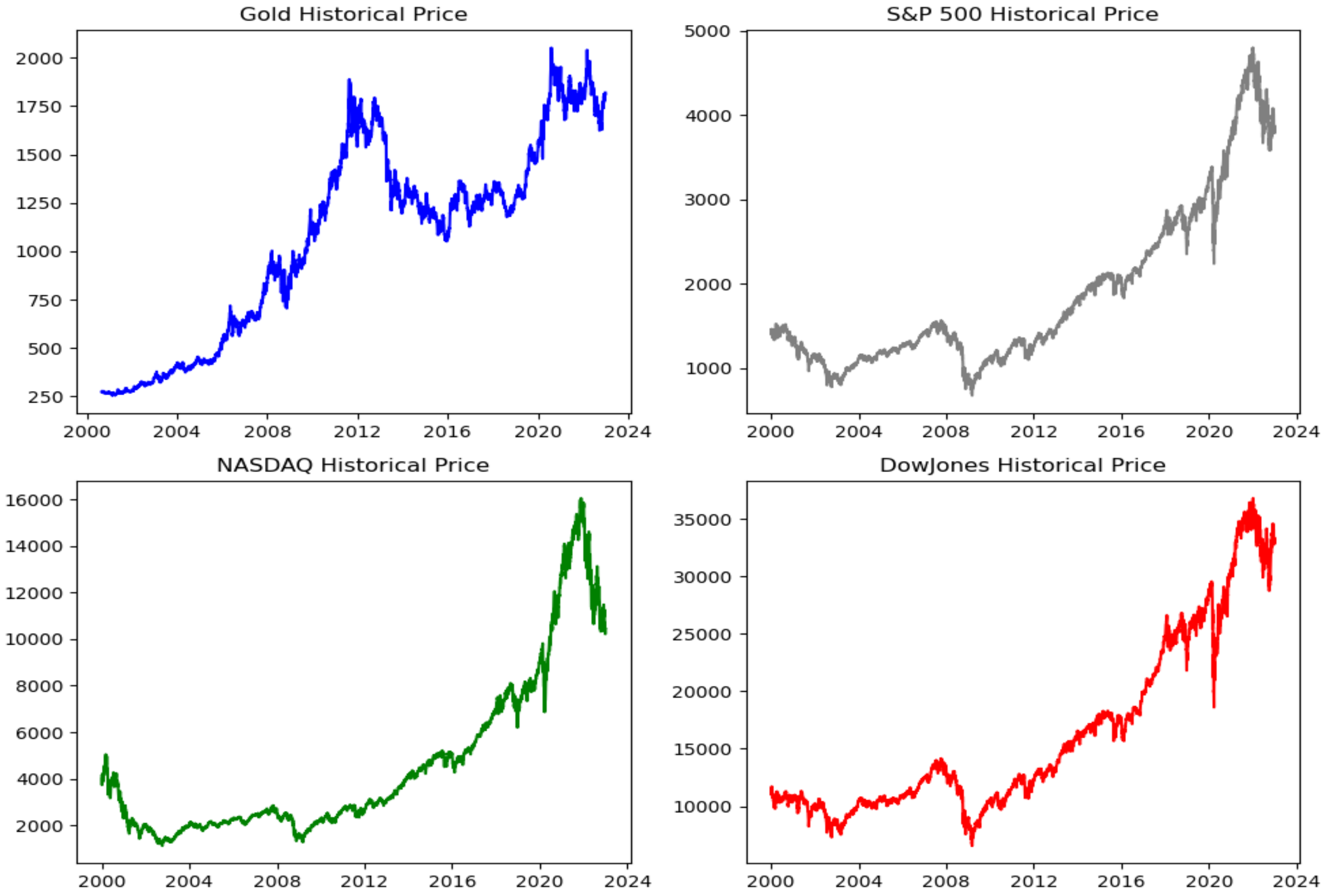

Figure 1: Historical Price Plots

weights assigned to recent observations, providing emphasis on more recent trends.

3. **EMA (Exponential Moving Average):** Exponential Average - A moving average that assigns greater weight to recent observations, enabling the capture of short-term trends in the data.

4. **SMA (Simple Moving Average):** Simple Average - An average calculated over a specified number of past data points, effectively smoothing fluctuations and revealing underlying trends.

5. **RSI (Relative Strength Index):** RSI - A momentum oscillator that gauges the speed and magnitude of price movements, signaling potential overbought or oversold conditions in the market.

6. **Close_lag:1:** Lag 1 of Closing Prices - Represents the previous day's closing price, providing insight into the immediate historical performance of the financial instrument.

7. **Close_difference:** Closing Price Difference - The discrepancy between consecutive closing prices, offering a measure of the directional movement in the financial instrument.

In our analysis, the target variable is defined as the closing price of the day. Notably, the sum of **Close_lag:1** and **Close_difference** directly corresponds to the closing price, encapsulating the entirety of relevant information. This combination serves as a comprehensive and informative set of features in our datasets.

However, our experimental results demonstrate a noteworthy observation: the Multi-Fractional Gradient Descent Linear Regression model uniquely exhibits the capability to fully discern and exploit this intrinsic relationship between **Close_lag:1**, **Close_difference**, and the closing price. In contrast, the other two models in consideration do not exhibit

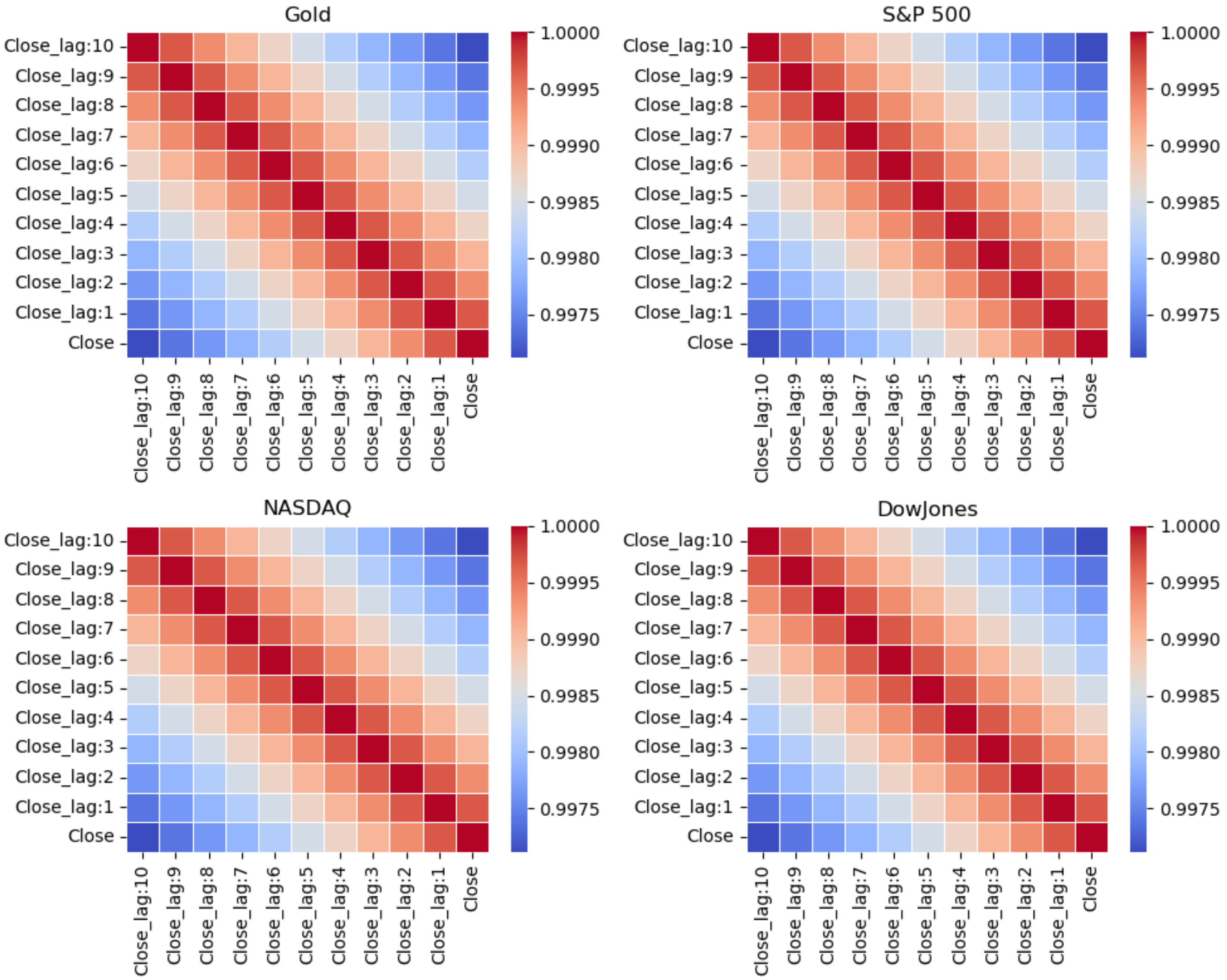

Figure 2: Correlation Tables

the same level of proficiency in capturing this vital feature. This highlights the distinctive effectiveness of the MFGD LR model in recognizing and utilizing the inherent information encapsulated within these features.

The following four tables present comprehensive summaries of the performance of three gradient descent methods employed in linear regression. It is important to highlight that the numerical values in the weights and derivative order columns correspond to the order of the features mentioned earlier. Additionally, it is essential to emphasize that the hyperparameters of all three models have been meticulously tuned to achieve optimal results, aiming for the lowest possible error for each respective model.

6 Conclusion

The paper under review provides a comprehensive examination of linear regression models utilizing the gradient descent method to determine optimal values for the loss function. The initial section of the study delves into the fundamental aspects of linear regression, emphasizing the significance of the gradient descent approach in optimizing the loss function.

A critical aspect of the paper involves an in-depth exploration of various definitions of fractional derivatives, setting the stage for a subsequent discussion on their application in the context of fractional gradient descent. This analysis contributes to the theoretical foundation of the study and establishes the groundwork for proposing a novel definition for fractional gradient descent.

The innovative approach introduced in the paper is

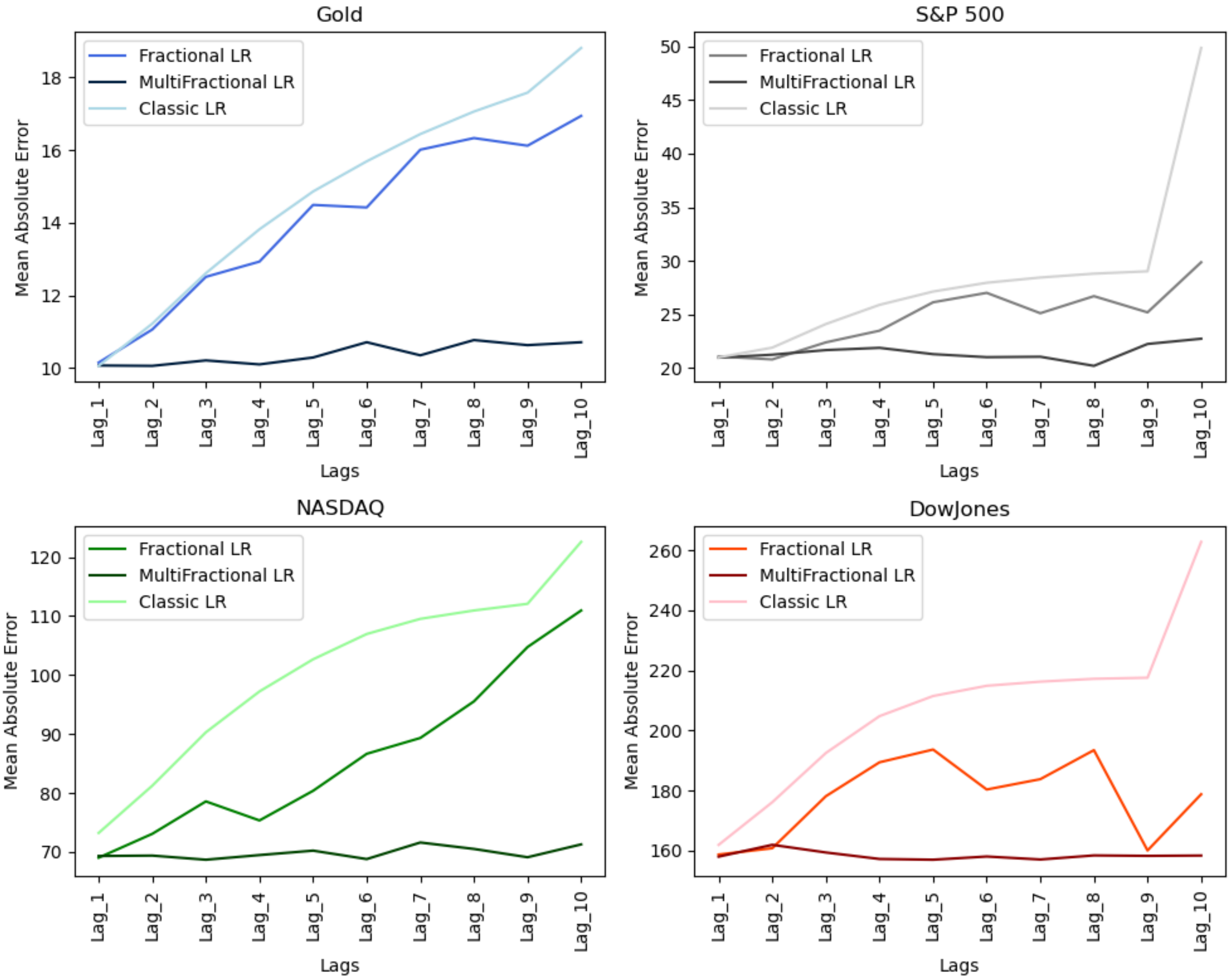

Figure 3: Performance Comparison of Tuned Models: The plot illustrates the results of the experiment, highlighting the robustness of the MFGD LR to multicollinearity, the degradation of GD LR performance due to collinearity, and the intermediate performance of the FGD LR model between these two scenarios. MFGD LR: Multi-Fractional Gradient Descent Linear Regression, FGD LR:Fractional Gradient Descent Linear Regression,GD LR: Gradient Descent Linear Regression

supported by a series of numerical experiments. The results of these experiments reveal the efficacy of the multi-fractional gradient descent method in enhancing the robustness of linear regression models, particularly in the presence of multicollinearity within the dataset. This finding is crucial as it addresses a common challenge in regression analysis and provides a potential solution to improve model performance under such conditions.

Furthermore, the paper demonstrates that the multi-fractional gradient descent method exhibits a capability to identify and leverage information embedded in the dataset. This adaptability leads to improved training and prediction outcomes when there is valuable information present in the data. The emphasis on leveraging information for enhanced model performance aligns with contemporary trends in machine learning and contributes to the broader understanding of gradient descent methods.

6.1 Future Works

In conclusion, our examination of the presented tables reveals a discernible correlation between the fractional derivative order (α) and the resulting weights in the model. This implies a connection between α and the information encoded in the independent variables (features). The investigation of this relationship offers promising avenues for future research.

	Alpha	Weights	MAE
Classic LR	1	0.137 0.218 0.223 0.215 0.007 0.226 -0.014	10.88
Fractional LR	0.97	0.087 0.223 0.290 0.210 0.035 0.200 0.021	9.62
Multi-Fractional LR	0.98 0.85 0.74 0.82 0.76 0.53 0.8	0.018 0.055 0.129 0.069 0.032 0.728 0.023	2.84

Table 1: Gold

	Alpha	Weights	MAE
Classic LR	1	0.129 0.214 0.226 0.198 0.041 0.262 -0.00023	23.65
Fractional LR	0.97	0.079 0.184 0.187 0.163 0.051 0.420 0.032	21.03
Multi-Fractional LR	0.61 0.58 0.96 0.82 0.96 0.32 0.49	0.011 0.057 0.011 -0.022 -0.007 0.944 0.075	1.97

Table 2: S&P 500

	Alpha	Weights	MAE
Classic LR	1	0.097 0.219 0.226 0.198 0.070 0.275 0.012	110.77
Fractional LR	0.94	0.096 0.198 0.208 0.187 0.076 0.332 0.047	81.04
Multi-Fractional LR	0.93 0.75 0.98 0.76 0.73 0.41 0.60	0.010 0.040 0.026 0.035 0.009 0.891 0.071	11.72

Table 3: NASDAQ

	Alpha	Weights	MAE
Classic LR	1	0.138 0.213 0.226 0.194 0.044 0.265 -0.004	217.40
Fractional LR	0.99	0.082 0.182 0.173 0.146 0.040 0.442 0.017	97.57
Multi-Fractional LR	0.81 0.94 0.97 0.96 0.98 0.41 0.60	0.012 0.024 0.029 0.026 0.004 0.911 0.073	20.64

Table 4: Dow Jones

Moreover, the applicability of the proposed gradient descent method extends beyond linear regression. It can be seamlessly implemented in other machine learning models, such as deep learning architectures like recurrent neural networks (RNNs).

Another avenue for future exploration involves the rigorous analysis of the convergence properties of the multi-fractional gradient descent method. Understanding the convergence behavior is essential for establishing the reliability and efficiency of the method across various scenarios.

It is worth noting that, due to the computation of first and second-order derivatives in the loss function for fractional derivative calculation, the presented method involves a higher computational burden compared to classic gradient descent. Future endeavors should focus on investigating methods to mitigate computational demands, making the approach more feasible for real-world applications. These considerations underscore the potential for refinement and enhancement in the proposed gradient descent method, paving the way for its broader adoption and practical utility.

References:

[1] L. Euler, De progressionibus transcendentibus seu quarum termini generales algebraice dari nequeunt, Commentarii academiae scientiarum Petropolitanae (1738) 36–57.

[2] P. Laplace, Théorie analytique des probabilités, courcier, paris, Oeuvres Complètes de Laplace 7 (1812) 523–525.

[3] J. B. J. Fourier, Théorie analytique de la chaleur, Gauthier-Villars et fils, 1888.

[4] N. H. Abel, œuvres complètes de Niels Henrik Abel, Vol. 1, Grøndahl, 1881.

[5] A. Letnikov, : On historical development of differentiation theory with an arbitrary index. mat. sb. 3, 85-112 (1868).

[6] A. Letnikov, Theory of differentiation with an arbitrary index, moscow mat (1868).

[7] A. Letnikov, On explanation of the main propositions of differentiation theory with an arbitrary index, Sb. Math 6 (1872) 413–445.

[8] J. Liouville, Mémoire sur quelques questions de géométrie et de mécanique, et sur un nouveau genre de calcul pour résoudre ces questions, 1832.

[9] J. Liouville, Mémoire sur le changement de la variable indépendante, dans le calcul des différentielles a indices quelconques, 1835.

[10] A. K. Grunwald, Uber" begrente" derivationen und deren anwedung, Zangew Math und Phys 12 (1867) 441–480.

[11] B. Riemann, Versuch einer allgemeinen auffassung der integration und differentiation, Gesammelte Werke 62 (1876) (1876).

[12] H. Laurent, Sur le calcul des dérivées à indices quelconques, Nouvelles annales de mathématiques: journal des candidats aux écoles polytechnique et normale 3 (1884) 240–252.

[13] O. Heaviside, Iii. on operators in physical mathematics. part i., Proceedings of the Royal Society of London 52 (315-320) (1893) 504–529.

[14] P. Kulczycki, J. Korbicz, J. Kacprzyk, Fractional Dynamical Systems: Methods, Algorithms and Applications, Vol. 402, Springer, 2022.

[15] R. P. Agarwal, Y. Zhou, Y. He, Existence of fractional neutral functional differential equations, Computers & Mathematics with Applications 59 (3) (2010) 1095–1100.

[16] R. P. Agarwal, D. O'Regan, S. Staněk, Positive solutions for dirichlet problems of singular nonlinear fractional differential equations, Journal of Mathematical Analysis and Applications 371 (1) (2010) 57–68.

[17] R. P. Agarwal, M. Benchohra, S. Hamani, A survey on existence results for boundary value problems of nonlinear fractional differential equations and inclusions, Acta Applicandae Mathematicae 109 (2010) 973–1033.

[18] N.-e. Tatar, Mild solutions for a problem involving fractional derivatives in the nonlinearity and in the non-local conditions, Advances in Difference Equations 2011 (2011) 1–12.

[19] K. Diethelm, N. J. Ford, Volterra integral equations and fractional calculus: do neighboring solutions intersect?, The Journal of Integral Equations and Applications (2012) 25–37.

[20] D. Baleanu, K. Diethelm, E. Scalas, J. J. Trujillo, Fractional calculus: models and numerical methods, Vol. 3, World Scientific, 2012.

[21] C. Ionescu, A. Lopes, D. Copot, J. T. Machado, J. H. Bates, The role of fractional calculus in modeling biological phenomena: A review, Communications in Nonlinear Science and Numerical Simulation 51 (2017) 141–159.

[22] J. S. Jacob, J. H. Priya, A. Karthika, Applications of fractional calculus in science and engineering, J. Crit. Rev 7 (13) (2020) 4385–4394.

[23] T.-Q. Tang, Z. Shah, R. Jan, E. Alzahrani, Modeling the dynamics of tumor–immune cells interactions via fractional calculus, The European Physical Journal Plus 137 (3) (2022) 367.

[24] T. Alinei-Poiana, E.-H. Dulf, L. Kovacs, Fractional calculus in mathematical oncology, Scientific Reports 13 (1) (2023) 10083.

[25] M. Joshi, S. Bhosale, V. A. Vyawahare, A survey of fractional calculus applications in artificial neural networks, Artificial Intelligence Review (2023) 1–54.

[26] D. Baleanu, Y. Karaca, L. Vázquez, J. E. Macías-Díaz, Advanced fractional calculus, differential equations and neural networks: analysis, modeling and numerical computations, Physica Scripta 98 (11) (2023) 110201.

[27] S. Shahmorad, R. Kalantari, A. Assadzadeh, Numerical solution of fractional black-scholes model of american put option pricing via a nonstandard finite difference method: Stability and convergent analysis, Mathematical Methods in the Applied Sciences 44 (4) (2021) 2790–2805.

[28] M. S. Raubitzek, K., T.Neubauer, Combining fractional derivatives and machine learning: A review., Entropy 25 (1) (2023) 462–467.

[29] S. Raubitzek, K. Mallinger, T. Neubauer, Combining fractional derivatives and machine learning: A review, Entropy 25 (1) (2022) 35.

[30] S. K. Chandra, M. K. Bajpai, Efficient machine learning and factional calculus based mathematical model for early covid prediction, Human-Centric Intelligent Systems (2023) 1–13.

[31] M. Gulian, M. Raissi, P. Perdikaris, G. Karniadakis, Machine learning of space-fractional differential equations, SIAM Journal on Scientific Computing 41 (4) (2019) A2485–A2509.

[32] R. Walasek, J. Gajda, Fractional differentiation and its use in machine learning, International Journal of Advances in Engineering Sciences and Applied Mathematics 13 (2-3) (2021) 270–277.

[33] R. Almeida, S. Pooseh, D. F. Torres, Computational methods in the fractional calculus of variations, World Scientific Publishing Company, 2015.

[34] Y. Chen, Q. Gao, Y. Wei, Y. Wang, Study on fractional order gradient methods, Applied Mathematics and Computation 314 (2017) 310–321.

Contribution of individual authors to the creation of a scientific article (ghostwriting policy)

Robab Kalantari and Khashayar Rahimi developed a novel method for gradient descent, conducted simulations, performed optimization, and were responsible for the writing and implementation of the proposed approach.
Saman Naderi has some edit in writing.