HA NOI UNIVERSITY OF SCIENCE AND TECHNOLOGY.

SCIENTIFIC COMPUTING

IT4110E

INSTRUCTOR: PROF. DR VU VAN THIEU

Class ID: 147831

Application of Fractional-Order Gradient Descent and Integer-Order Gradient Descent : Comparison and Evaluation.

Group Members:
Nguyen Cong Huan - 20227974
Phan Gia Do - 20226026
Nguyen Cong Dat - 20226022
Tran Gia Khanh - 20226048
Nguyen Trung Hieu - 20225971
Nguyen Dang Truong Giang - 20226036

Abstract

This report establishes a model of economic growth of the United States from 1978 to 1997, in which the gross domestic product (GDP) is related to arable land, population, gross capital formation, exports of goods and services, general government final consumer spending and broad money.The ***fractional-order gradient descent*** and ***integer-order gradient descent*** are used to estimate the model parameters to fit the GDP and forecast GDP from 1998 to 2000.

1 Introduction

In recent years, fractional model has become a research hotspot because of its advantages. Fractional calculus has developed rapidly in academic circles, and its achievements in the fields include.

Gradient descent is generally used as a method of solving the unconstrained optimization problems, and is widely used in evaluation and in other aspects. The rise in fractional calculus provides a new idea for advances in the gradient descent method. Although numerous achievements have been made in the two fields of fractional calculus and gradient descent, the research results combining the two are still in their infancy. Recently, ref. applied the fractional order gradient descent to image processing and solved the problem of blurring image edges and texture details using a traditional denoising method, based on integer order. Next, ref. improved the fractional-order gradient descent method and used it to identify the parameters of the discrete deterministic system in advance. Thereafter, ref. Applied the fractional-order gradient descent to the training of neural networks' backpropagation (BP), which proves the monotony and convergence of the method.

Compared with the traditional integer-order gradient descent, the combination of fractional calculus and gradient descent provides more freedom of order; adjusting the order can provide new possibilities for the algorithm. In this paper, economic growth models of seven countries are established, and their cost functions are trained by gradient descent (fractional- and integer-order). To compare the performance of fractional- and integer-order gradient descent, we visualize the rate of convergence of the cost function, evaluate the model with MSE, MAD and R^2 indicators and predict the GDP of the USA in 1998-2000 according to the trained parameters.

2 Knowledge

2.1 Model describes

The prediction of variables generally uses time series models (for example, ARIMA and SARIMA), or artificial neural networks, which have been very popular in recent years. The time series model mainly predicts the future trend in variables, but it is difficult to reflect the change in unexpected factors in the model. Additionally, the neural network model needs to adjust more parameters, the network structure selection is too large, the training efficiency is not high enough, and easy to overfit. Although the linear model is simple in form and easy to model, its weight can intuitively express the importance of each attribute, so the linear model has a good explanatory ability. It is reasonable to build a linear regression model of economic growth, which can clearly learn which factors have an impact on the economy. Next, we chose six explanatory variables to describe the economic growth. The explained variable is y, where y refers to GDP and is a function. The expression for y is as follows:

$$y(t) = \sum_{j=1,2,3,4,5,6} \theta_j x_j(t) + \theta_0 + \varepsilon \tag{1}$$

where t is year, θ_0 is the intercept. ε is an unobservable term of random error. θ_j represents the weight of each variable. The six explanatory variables are:

x_1 :arable land (hm^2)
x_2 :population
x_3 :gross capital formation (USD)
x_4 :exports of goods and services (USD)
x_5 :general government final consumer spending (USD)
x_6 :broad money (USD)

2.2 Fractional-Order Derivative

Due to the differing conditions, there are different forms of fractional calculus definition, the most common of which are Riemann–Liouville, and Caputo. In this article, we chose the definition of fractional-order derivative in terms of the Caputo form. Given the function $f(t)$, the Caputo fractional-order derivative of order α is defined as follows:

$$^{Caputo}D_x^{\alpha} f(t) = \frac{1}{\Gamma(1-k)} \int_c^t (t-\tau)^{-\alpha} f'(\tau) d\tau$$

where $^{Caputo}D_x^{\alpha}$ is the Caputo derivative operator. α is the fractional order, and the interval is $\alpha \in (0,1)$. $\Gamma(.)$ is the gamma function. c is the initial value.

2.3 The Cost Function

The cost function (also known as the loss function) is essential for a majority of algorithms in machine learning. The model's optimization is the process of training the cost function, and the partial derivative of the cost function with respect to each parameter is the gradient mentioned in gradient descent. To select the appropriate parameters θ for the model (1) and minimize the modeling error, we introduce the cost function:

$$C(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2 \tag{2}$$

where $h_\theta(x^{(i)})$ is a modification of model (1), $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + ... + \theta_j x_j$, which represents the output value of the model. $x^{(i)}$ are the sample features. $y^{(i)}$ is the true data, and t represents the number of samples.

2.4 The Integer-Order Gradient Descent

Integer-Order Gradient Descent is the traditional gradient descent method, where first-order or higher-order derivatives are used to update parameters.

- Application in optimization: Integer-order Gradient Descent uses information from first-order derivatives to adjust parameters, helping to find the minimum of the objective function.
- Update formula: The update formula in IOGD uses first-order derivatives. For example:

$$\theta_{\text{new}} = \theta_{\text{old}} - \alpha \times \nabla f(\theta_{\text{old}})$$

Where $\nabla f(\theta_{\text{old}})$ is the gradient (first-order derivative) of the objective function at θ_{old}

The first step of the integer-order gradient descent is to take the partial derivative of the cost function $C(\theta)$:

$$\frac{\partial C(\theta)}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)x_j^{(i)}, \quad j = 1,\ldots,6. \tag{3}$$

and the update function is as follows:

$$\theta_{j+1} = \theta_j - \eta\frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)x_j^{(i)} \tag{4}$$

where η is learning rate, $\eta > 0$.

2.5 The Fractional-Order Gradient Descent

The first step of fractional-order gradient descent is to find the fractional derivative of the cost function $C(\theta)$. According to Caputo's definition of fractional derivative, we know that if *g(h(t))* is a compound function of *t*, then the fractional derivation of α with respect to *t* is

$${}_cD_t^\alpha g(h) = \frac{\partial(g(h))}{\partial h} \cdot {}_cD_t^\alpha h(t). \tag{5}$$

It can be known from (5) that the fractional derivative of a composite function can be expressed as the product of integral and fractional derivatives. Therefore, the calculation for ${}_cD_{\theta_j}^\alpha C(\theta)$ is as follows:

$${}_cD_{\theta_j}^\alpha C(\theta) = \frac{1}{m(1-\alpha)\Gamma(1-\alpha)}(\theta_j - c)^{(1-\alpha)}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)x_j^{(i)}, j = 1,2,...6$$

and the update function is as follows:

$$\theta_{j+1} = \theta_j - \eta\frac{1}{m(1-\alpha)\Gamma(1-\alpha)}(\theta_j - c)^{(1-\alpha)}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)},$$

where η is the learning rate, $\eta > 0$. α is the fractional order, $0 < \alpha < 1$. C is the initial value of Caputo's fractional derivative, and $c < \min\{\theta_j\}$.

3 Model Evaluation Indexes

We use the absolute relative error (*ARE*) to measure the prediction error:

$$ARE_i = \frac{|y_i - \widehat{y_i}|}{y_i}.$$

To evaluate the fitting quality of gradient descent on the model, the following three indicators can be calculated: The mean square error ((MSE)):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_1 - \widehat{y_i})^2.$$

The coefficient of determination (R^2):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \widehat{y_i})^2}{\sum_{i=1}^{n}(y_i - \overline{y_i})^2}.$$

The mean absolute deviation (*MAD*):

$$MAD = \frac{\sum_{i=1}^{n}|y_i - \widehat{y_i}|}{n}.$$

In these formulas, n is the number of years (n = 44). y_i and $\widehat{y_i}$ are the real value and the model output, respectively. $\overline{y_i}$ is the mean of the GDP.

4 Main Results

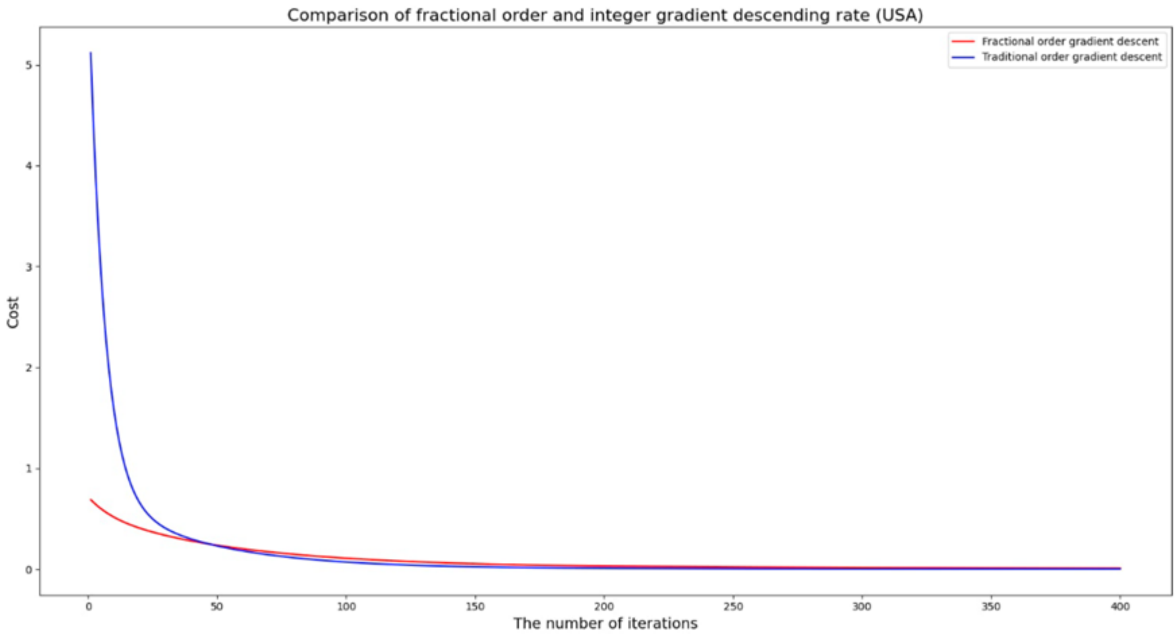
In here, we standardize the data for each country before running the algorithm, and each iteration to update θ uses m samples. The grid search method was used to select the appropriate learning rate and initial weight interval, and the effects of different fractional orders are compared to select the best order (Table 1). The learning rate and the initial weight interval are applicable to both fractional-order gradient descent and integer-order gradient descent.

Table 1: Parameters for the United States.

Country	α	**Learning Rate**	**Initial Interval**
The United States	0.8	0.03	$(-0.1, 0.1)$

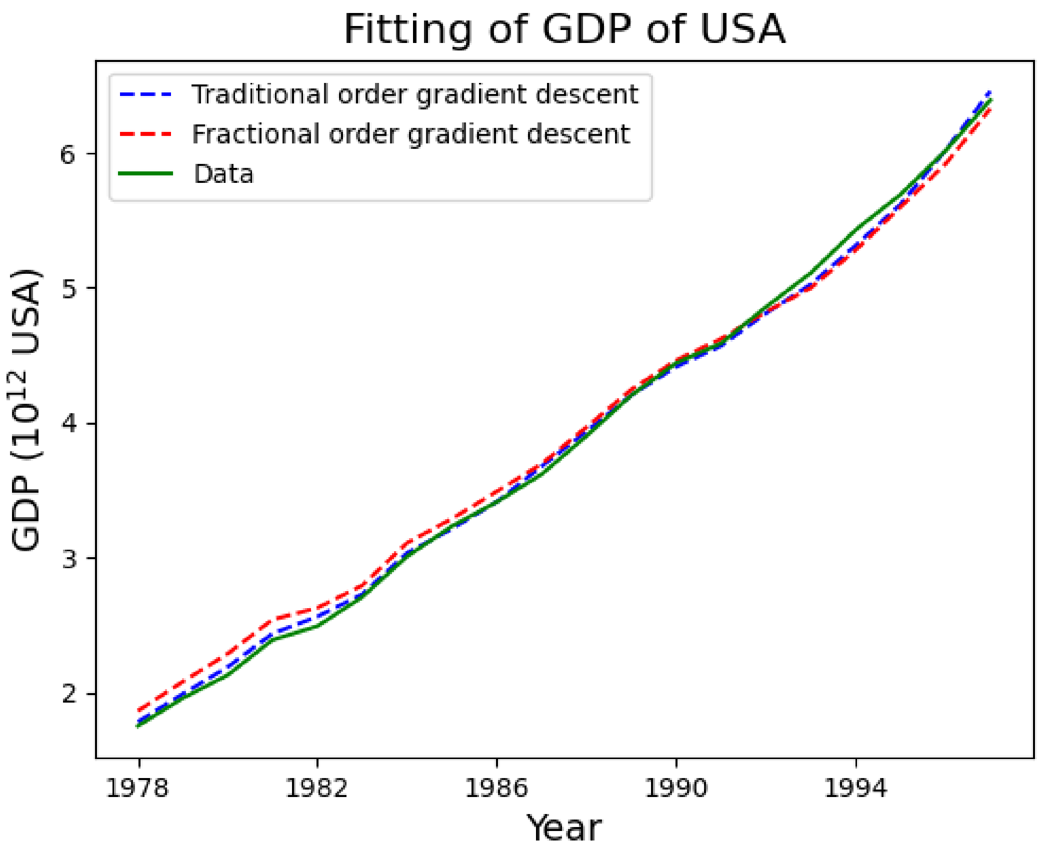
4.1 *Comparison of Convergence Rate of Fractional and Integer Order Gradient Descent*

As shown in the figure, after the same number of iterations, the convergence rate of fractional-order gradient descent is faster than that of integer-order gradient descent, which indicates that the method combining fractional-order and gradient descent is better than the traditional integer-order gradient descent in the convergence rate of update equation.



4.2 *Fitting Result*

It can be seen from figure that the *MSE* results of GDP fitted by fractional-order gradient descent are better than that fitted by integer-order gradient descent, which indicates that, under the same iteration number, learning rate and initial weight interval, the fitting performance of the data fitted by fractional-order gradient descent is better than that of integer-order.



4.3 *Predicted Results*

Finally, in order to test the prediction effect of fractional and integer-order gradient descent on GDP, we forecast the GDP from 1998 to 2000, and used the absolute relative error (ARE) index to measure the prediction error.

Country	Year	Actual Value	Predicted Value		ARE	
			Integer	Fractional	Integer	Fractional
The United States	1998	6752705842310	6802758580679.91	6649228925311.35	0.0074	0.0153
	1999	7176188111753	7270804905986.7	7078790768071.62	0.0132	0.0136
	2000	7637981352565	7750315862802.78	7577128756098.85	0.0147	0.0079

5 Conclusions

The results show that, the gradient descent method can also solve the regression analysis problem by iterating the cost function, and obtain good results, a without complicating the mode.

We apply the fractional differential to gradient descent, and compare the performance of fractional-order gradient descent with that of integer-order gradient descent. It was found that the fractional-order has a faster convergence rate, higher fitting accuracy and lower prediction error than the integer-order. This provides an alternative method for fitting and forecasting GDP and has a certain reference value.

References

- Peshawa J. Muhammad Ali, Haval Abdulkarim Ahmed; "Gradient Descent Algorithm: Case Study", Machine Learning Technical Reports, 2021, 2(1), pp 1-7.

- Wang, X.; Feckan, M.; Wang, J. Forecasting Economic Growth of the Group of Seven via Fractional-Order Gradient Descent Approach. Axioms 2021.

- Rukshan Pramoditha. Linear Regression with Gradient Descent. 2020