Article

Neural Multivariate Grey Model and Its Applications

Qianyang Li and Xingjun Zhang *

School of Computer Science and Technology, Xi’an Jiaotong University, Xi’an 710049, China; liqianyang@stu.xjtu.edu.cn
* Correspondence: xjzhang@xjtu.edu.cn

Abstract: For time series forecasting, multivariate grey models are excellent at handling incomplete or vague information. The GM(1, N) model represents this group of models and has been widely used in various fields. However, constructing a meaningful GM(1, N) model is challenging due to its more complex structure compared to the construction of the univariate grey model GM(1, 1). Typically, fitting and prediction errors of GM(1, N) are not ideal in practical applications, which limits the application of the model. This study presents the neural ordinary differential equation multivariate grey model (NMGM), a new multivariate grey model that aims to enhance the precision of multivariate grey models. NMGM employs a novel whitening equation with neural ordinary differential equations, showcasing higher predictive accuracy and broader applicability than previous models. It can more effectively learn features from various data samples. In experimental validation, our novel model is first used to predict China’s per capita energy consumption, and it performed best in both the test and validation sets, with mean absolute percentage errors (MAPEs) of 0.2537% and 0.7381%, respectively. The optimal results for the compared models are 0.5298% and 1.106%. Then, our model predicts China’s total renewable energy with lower mean absolute percentage errors (MAPEs) of 0.9566% and 0.7896% for the test and validation sets, respectively. The leading outcomes for the competing models are 1.0188% and 1.1493%. The outcomes demonstrate that this novel model exhibits a higher performance than other models.

Keywords: GM(1, N); neural ordinary differential equations; adjoint sensitivity algorithm; energy prediction

1. Introduction

Forecasting time series has consistently been of great importance for domains such as finance, economics, markets, and social issues. Various forecasting methods emerge indefinitely as science and technology advance. There are numerous models and methods available for time series, including LR (linear regression), ARIMA [1], LSTM, and neuron models [2,3]. However, large sample sizes are required for building these prediction models. As we all know, it can be challenging to gather big samples of data for some applications in the real world. The conventional model of forecasting is ineffective when dealing with such a tiny sample set. Recently, there have been numerous methods for making predictions from small sample sizes, including neural networks [4], fuzzy theory [5], and grey system theory [6,7].

Professor Deng was considered a pioneer of grey systems theory in 1982 [8]. This theory has evolved into a valuable framework for dealing with uncertain or limited information in various fields. The theory employs mathematical tools, such as grey differential equations and grey accumulation operations, to progressively refine models and extract inherent patterns in the system. Various grey models have demonstrated remarkable predictive and analytical capabilities in numerous real-world scenarios, including in agriculture, meteorology, geology, the economy, management [6], energy [9–11], industry [12,13], smog pollution [14,15], and society [16,17].
The grey model is categorized into univariate and multivariate types. The GM(1, 1) model is a univariate model that consists of a first-order difference equation. The GM(1, N) model, within the realm of multivariate modeling, comprises a primary variable denoting the system characteristic when other N-1 variables represent the influencing factors. However, the GM(1, N) model’s accuracy has been impacted by its imprecise time response, which restricts its applicability. As a result, the GM(1, N) model has undergone enhancements by several researchers, yielding substantial advancements and the creation of various grey multivariate prediction models. A novel optimized grey prediction model was created by Zeng [18], and various experiments were conducted to illustrate the efficiency of the model. Wang [19] enhanced the flexibility of the multivariable grey model by integrating time power terms into a prediction model for multivariable grey. Zeng et al. introduced a novel multivariable grey optimizing model called OGM(1, N) [20] to address issues related to parameter defects, mechanism defects, and structural defects. By applying dynamic background values, Lao et al. [21] developed a new multivariate grey prediction model and optimized its parameters using the whale optimization method. Dang et al. [22] present a novel multivariate grey model known as the GM(1, N \sin) power model, which aims to address the limitation of grey multivariate prediction models in accurately simulating systems that exhibit periodic oscillations. Ye et al. expand the prediction theory of the DGM(1, N) model structure by introducing action duration, time delay, and an intensity coefficient, and proposes the ATIGM(1, N) [13] model, which is effective at forecasting time delay data. The outcomes of these studies have strengthened multivariate grey prediction models and substantially expanded the scope of grey system theory; both have resulted in extensive application. These variants of GM(1, N) have consistently demonstrated a superior performance in tasks related to forecasting time series data.

Neural ordinary differential equations (NODE) [23] are a class of neural networks introduced to address continuous-depth models. These networks utilize ordinary differential equations to articulate the neural network’s dynamics, enabling a flexible approach to depth rather than relying on a fixed number of layers. NODEs build on this idea by considering ResNets [24] as a specific case in which the number of layers becomes infinite. The computation of ResNet’s features can be regarded as the Euler method for solving an ordinary differential equation [25,26]. Many scholars are interested in NODEs because they link differential equations with neural networks, resulting in a novel neural network domain. They are now being used in a variety of deep learning research [27–30]. Based on NODE, the neural ordinary differential grey model (NODGM) [31] has been proposed as a novel grey model. The NODGM model incorporates an innovative whitening equation, allowing the learning of the prediction model through the training process. Recently, Lei [32] proposed a time-delayed nonlinear neural grey model based on LSTM. However, these models are limited to a single variable and do not take into account the relevant factors, so it cannot be generalized to more realistic scenarios with multiple variables. Additionally, there is also a demand to improve the grey model’s accuracy.

The modeling process in GM(1, N) contains multiple variables or factors, allowing for a more comprehensive analysis of systems involving multiple influencing factors. It is an extension of GM(1, 1). This extension enables the GM(1, N) model to handle and consider the influence of multiple variables, thereby enhancing its applicability to complex systems where interactions between various factors need to be considered in prediction or analysis. To the best of my knowledge, there has not been any research on integrating NODE with GM(1, N) models. This paper aims to introduce a new model termed the neural ordinary differential multivariate grey model (NMGM). Unlike the conventional GM(1, N) model, which uses the least squares technique for parameter estimation, the NMGM model will integrate NODE methodology, offering a new modeling method for multivariate grey models. When working with a small sample size, the approach is more likely to lead to overfitting. This study presents a novel grey prediction model that utilizes NODE to establish the whitening equation, aiming to address these issues. Neural network training will reduce overfitting and enhance forecast accuracy in the model. The model
does not require a defined structure, as the optimum model is derived through training and can be solved using advanced numerical methods. Therefore, our NMGM model can have better adaptability and generalization.

The main contributions of this study can be summarized as follows:

1. We propose the neural ordinary differential multivariate grey model (NMGM), which is a novel multivariate grey model that is based on the NODE. Our goal is to improve the accuracy of predicting data that include insufficient or limited information.
2. The NMGM model employs gradient descent as the training algorithm to obtain the model’s parameters, as opposed to the conventional least squares method. This approach enhances the precision of the model. The optimal model parameters are obtained for this study using the adjoint sensitivity method.
3. Our model is validated in two separate cases. When predicting China’s per capita energy consumption from 2012 to 2021, our model obtains a mean absolute percentage error (MAPE) of 0.2537% and 0.7381% in the test and validation sets, respectively. The MAPE of our model for predicting China’s total renewable energy from 2013 to 2022 is 0.9566% in the test set and 0.7896% in the validation set. Our prediction results are more accurate than those of other models, which is extremely valuable to national energy management and policymaking.

The subsequent sections of this work are organized in the following manner. In Section 2, we present the fundamental concept of the conventional GM(1, N) model and the neural ODEs. Section 3 presents our innovative NMGM model, referring to the model procedure, parameter optimization, and performance evaluation. The proposed model is utilized in Section 4 to simulate and predict China’s energy consumption per capita and total renewable energy. Finally, our conclusions are presented in Section 5.

2. Literature Review

This section introduces two topics, the GM(1, N) model and the NODEs model.

2.1. GM(1, N) Model

The GM(1, N) model operates as a prediction model with N variables [6]. It is designed to handle N-1 influencing factors alongside one main system behavior variable. The GM(1, N) model differs from traditional univariate models like GM(1, 1) by incorporating multiple relevant factors, thus enhancing prediction accuracy in complex systems. Then, a time series of the \( m \) dimension or an original sequence

\[
X_i^{(0)} = (x_i^{(0)}(1), x_i^{(0)}(2), \ldots, x_i^{(0)}(m)),
\]

where the system characteristic sequence is \( i = 1 \), and the relevant factor sequences are \( i = 2, 3, \ldots, N \), \( X_j^{(1)} \) is the 1-AGO (accumulating generation operator) sequences of \( X_j^{(0)} \), \( j = 1, 2, \ldots, N \), which are defined as

\[
X_j^{(1)} = (x_j^{(1)}(1), x_j^{(1)}(2), \ldots, x_j^{(1)}(m)),
\]

where \( x_j^{(1)}(i) = \sum_{k=1}^{i} x_j^{(0)}(k), i = 1, 2, \ldots, m \).

An advantageous aspect of AGO is its ability to uncover potential hidden patterns within data sequences, facilitating the identification of underlying regularities [8]. The mean sequence that is created by consecutive neighbors of \( X_1^{(1)} \) is denoted by the \( Z_1^{(1)} \):

\[
Z_1^{(1)} = (z_1^{(1)}(2), z_1^{(1)}(3), \ldots, z_1^{(1)}(m)),
\]
where \( z_1^{(1)}(k) = 0.5 * (x_1^{(1)}(k) + x_1^{(1)}(k - 1)), k = 2, 3, \ldots, m \), then the GM(1, N) model [6] can be defined as:

\[
x_1^{(0)}(k) + az_1^{(1)}(k) = \sum_{j=2}^{N} b_jx_j^{(1)}(k), k = 2, 3, \ldots, m,
\]

where \(-a\) represents the development coefficient, \(b_j\) stands for the driving coefficient, and \(b_jx_j(k)\) denotes the driving item. The parameter sequence is denoted by \(\hat{a} = [a, b_2, b_3, \ldots, b_N]^T\). Then, the GM(1, N) model’s whitening equation [6] is denoted by

\[
\frac{dx_1^{(1)}}{dt} + ax_1^{(1)} = \sum_{j=2}^{N} b_jx_j^{(1)}.
\]

Ordinary least squares (OLS) are used by the GM (1,N) to calculate \(a, b_2, b_3, \ldots, b_N\) [6]:

\[
[a, b_2, \ldots, b_N] = (B^TB)^{-1}B^T Y,
\]

where \( B = \begin{bmatrix}
-\hat{z}_1^{(1)}(2) & \hat{x}_2^{(1)}(2) & \cdots & \hat{x}_N^{(1)}(2)
-\hat{z}_1^{(1)}(3) & \hat{x}_2^{(1)}(3) & \cdots & \hat{x}_N^{(1)}(3)
\vdots & \vdots & \ddots & \vdots
-\hat{z}_1^{(1)}(m) & \hat{x}_2^{(1)}(m) & \cdots & \hat{x}_N^{(1)}(m)
\end{bmatrix}, Y = \begin{bmatrix}
x_1^{(0)}(2)
x_1^{(0)}(3)
\vdots
x_1^{(0)}(m)
\end{bmatrix}.
\]

With the initial condition, \(x_1^{(0)}(0)\) is taken to \(x_1^{(0)}(1)\). The whitening equation (5), has a solution [6] that can be obtained by:

\[
x_1^{(1)}(t) = e^{-at} \left[ x_1^{(0)} - t \sum_{j=2}^{N} b_jx_j^{(1)}(0) + \sum_{j=2}^{N} \int b_jx_j^{(1)}(t)e^{at} dt \right].
\]

When there are tiny changes in \(X_i^{(1)}(i = 1, 2, \ldots, N), \sum_{j=2}^{N} b_jx_j^{(1)}(0)\) represents a grey constant. We can obtain the approximate time-response sequence [6] \(\hat{x}_1^{(1)}\) of the GM(1, N) model:

\[
\hat{x}_1^{(1)}(1) = \left[ x_1^{(0)} - \frac{1}{a} \sum_{j=2}^{N} b_jx_j^{(1)}(k + 1) \right] e^{-ak} + \frac{1}{a} \sum_{j=2}^{N} b_jx_j^{(1)}(k + 1).
\]

Ultimately, by employing the inverse accumulated generation operation (IAGO), the forecasting value for the raw data \(\hat{x}_1^{(0)}(k)\) of \(x_1^{(0)}(k)\) can be reconstructed using the following equation:

\[
\hat{x}_1^{(0)}(k) = \hat{x}_1^{(1)}(k) - \hat{x}_1^{(1)}(k - 1).
\]

By extensively investigating the definition and modeling procedure of the GM(1, N) model, it has been demonstrated that the model’s structure, parameter application, and modeling mechanism have several shortcomings [20]. The conventional GM(1, N) model faces limitations in its application and stability primarily because of these critical challenges. From the GM(1, N) model, numerous improved models have been developed by combining novel techniques including the time factor, fractional order, or discretization, just as those described in the first chapter augmented the grey system theory and improved the multivariate grey prediction mode. However, its application and generalization need to be enhanced due to the GM(1, N) model’s inherent defects.

### 2.2. Neural ODEs

Neural ordinary differential equations (NODEs) [23] can be seen as a conceptual extension of residual networks (ResNets [24]) when considering an infinite number of layers. NODEs view the entire network as a continuous process governed by ordinary
differential equations. We can use traditional gradient descent to train ODEs in continuous neural networks. To illustrate this, let us investigate how ResNet’s hidden states transition from current layer $t$ to the next layer, $t + 1$:

$$z_{t+1} = z_t + f_t(z_t), \quad (10)$$

where $z_t \in \mathbb{R}^d$ represents the hidden state in $\mathbb{R}^d$ at $t$, while $f_t : \mathbb{R}^d \rightarrow \mathbb{R}^d$ denotes a differentiable function maintaining the dimensionality of $z_t$. Consider a neural network denoted by $f_{\theta}$, where $\theta$ is a parameter set, $z(t)$ represents the evolution data of the neural network at time $t$, and $t \in [t_0, t_1]$. Then, the NODE [23] is specified as

$$\frac{dz(t)}{dt} = f_{\theta}(z(t), t). \quad (11)$$

At time $t$, $z(t)$ represents the hidden state of the NODE, evolving through the model from the initial condition $z(t_0)$. For any given value of $t$, the value of $z(t)$ can be ascertained by solving an ordinary differential equation (ODE) coupled with initial value problems (IVP). The forward propagation of data can be obtained through the following integral equation:

$$z(t_1) = z(t_0) + \int_{t_0}^{t_1} f_{\theta}(z(t), t). \quad (12)$$

With the same dimensions, the input $z(t_0)$ of the neural network at time $t_0$ can generate the solution $z(t_1)$ of the equation at time $t_1$. As shown in Figure 1, with only one ODE layer, this model architecture is straightforward.

![Figure 1](image)

**Figure 1.** Diagram of neural ODE architecture. $f_{\theta}$ is the neural network function with parameters $\theta$ defined in $\mathbb{R}^d$. Maintain a time-invariant mapping between $\mathbb{R}^d$ to $\mathbb{R}^d$ from time $t_0$ to $t_1$.

Naturally, the gradient of the solution trajectory at any given point in space is defined by the network $f_{\theta}$ itself. With the initial value $z(t_0)$, $\theta$, start point $t_0$, and final point $t_1$, we can use typical numerical ODE solvers (e.g., the Runge–Kutta method [33]) to compute the above integral.

$$z(t_1) = \text{ODESolver}(z(t_0), f_{\theta}, t_0, t_1). \quad (13)$$

In order to train the parameters, we choose the MSELoss $\mathcal{L}$ between the real value $G$ and the ODESolver output as the loss function:

$$\mathcal{L}(z(t_1)) = \left( \text{ODESolver}(z(t_0), f_{\theta}, t_0, t_1) - G \right)^2. \quad (14)$$

When optimizing the loss function, it is possible to backpropagate using the ODE solver’s operations. Nevertheless, this approach results in a significant memory overhead and introduces extra numerical inaccuracies due to the potential expansion of the computational graph. Furthermore, the neural ODE model was trained by solving one extra ODE backward in time using the adjoint sensitivity approach [34]. This approach has a direct relationship with the size of the data, requires low memory usage, and effectively minimizes...
numerical inaccuracies. This approach is particularly suitable for predicting small sample data, as discussed in this paper.

3. NMGM Model

This section presents our innovative multivariate grey model, NMGM. The NMGM model contains a function $f_\theta(X_1(t), X_2(t), \cdots, X_N(t), t)$ that replaces the derivative of the 1-AGO sequence with respect to $t$ in the conventional GM model. The whitening equation denoted by Equation (5) can be reformulated as a neural ordinary differential equation (NODE), allowing the equation to be trained with neural networks to make a better performance model. Compared with the NODGM [31], our novel model takes N-1 relevant factors as the input of the model; so it is possible to fully utilize all of the information on relevant factors in order to achieve better predictive outcomes.

3.1. Modeling Procedure

In the NMGM model, we assume that $X^{(0)}_i(t)$ represents the system’s characteristic sequence when $i = 1$, and the pertinent factor sequences when $i = 2, 3, \cdots, N$. This assumption is based on the definition of GM(1, N) in Section 2. Subsequently, the 1-AGO (accumulating generation operator) sequences of $X^{(0)}_i$ are denoted as $X^{(1)}_1(t), X^{(1)}_2(t), \cdots, X^{(1)}_N(t)$. Considering that $f_\theta$ is a function of $(X^{(1)}_1(t), X^{(1)}_2(t), \cdots, X^{(1)}_N(t), t)$, the coefficient $\theta$ is the dependent variable. Then, the function $f_\theta$ is defined by the deep neural network, which can represent a series of transformations on $t$ and the 1-AGO sequence $X^{(1)}_1(t), X^{(1)}_2(t), \cdots, X^{(1)}_N(t)$. A deep neural network can be understood as a function that executes a series of operations on its input data. We define $z(t)$ as the hidden state of the neural network. The NMGM model’s whitening equation can be represented as a NODE:

$$\frac{dz(t)}{dt} = f_\theta(z(t), t),$$

where $\theta$ denotes the associated weights for neural networks. Once the neural network is defined, employing the initial value $z(0)$, time span $t_s$, and $\theta$, we can proceed to invoke the ODESolver, enabling the computation of the forecast sequence $Z$:

$$Z = \left[z(0), \hat{z}(1), \cdots, \hat{z}(n-1)\right] = \text{ODESolver}(z(t), t, f_\theta(z(t), t), z(0), t_s, \theta),$$

where $t_s = [0, 1, 2, \cdots, n-1]$. After constructing the model, we can use forward propagation and backpropagation to optimize parameters to obtain the optimal prediction sequence. Based on the above theoretical explanations, the flowchart for the new proposed model can be seen in Figure 2. The computational steps are summarized below:

1. Collecting the raw data set $X^{(0)}_1, X^{(0)}_2, \cdots, X^{(0)}_N$;
2. The 1-AGO series $X^{(1)}_1, X^{(1)}_2, \cdots, X^{(1)}_N$ of raw data are computed and used as input to the NMGM model;
3. Constructing the NMGM model, using forward propagation and backpropagation to optimize parameters;
4. Computing the simulated values and errors where the predictive value can be generated by the IAGO;
5. Predicting the future value and analyzing the development trend of the system.
Step 1: Data Preparation
(Collecting raw data set)

Step 2: Data Processing
1. Data Normalization
2. Accumulate generating sequences (1-AGO)

Step 3: Constructing the NODMGM model
1. Using forward propagation and backpropagation to optimize parameters
2. Training model with adjoint method

Step 4: Computing the simulated values and errors
1. Computing the simulated values
2. Computing the reduction values (IAGO)
3. Computing the MAPE

Meeting the accuracy requirement?

Step 5: Predicting the future value
1. Forecasting the future values of the dependent variable
2. Analyzing the development trend of the system (Optional)

Figure 2. The flowchart of NMGM.

3.2. Parameter Optimization

In fact, provided with an initial value and a specific timeframe, the NODE is designed to generate a prediction sequence matching the length of the given time span. This capability ensures the production of a prediction sequence that aligns precisely with the designated duration, allowing for accurate forecasting based on the initial conditions and temporal scope provided. This sequence is then used to calculate the loss. As the 1-AGO series $X^{(1)}_1, X^{(1)}_2, \ldots, X^{(1)}_N$ are used as input to our model, the output also has $N$ dimensions in order to optimize the weight $\theta$ and obtain the optimal prediction sequence of the raw system characteristic sequence $X^{(0)}_1$. To train NODE, the following loss function is defined:

$$
\mathcal{L} = \sum_{t=0}^{n} \left( z - z(t) \right)^2 = \left( Z - X^{(1)}_1 \right)^2,
$$

where $X^{(1)}_1$ is the 1-AGO sequences of system characteristic sequence $X^{(0)}_1$. Based on the backpropagation algorithm, the model parameters $\theta$ can be updated through gradients:

$$
\theta^{(t+1)} = \theta^t - \lambda \frac{d\mathcal{L}}{d\theta}.
$$

(17)
Equation (16) can be used to obtain the prediction sequence \( \hat{x}^1 \) after training the parameter \( \theta \). Eventually, we can obtain the predicted sequence of the initial data prior to accumulation by employing the restoration equation, Equation (19)

\[
\hat{x}^0(k) = \hat{x}^1(k) - \hat{x}^1(k-1), \text{s.t. } k = 1, 2, \ldots, m - 1.
\]  

(19)

The process of the algorithm for the NMGM model is illustrated in Algorithm 1. A full adjoint sensitivities algorithm can be used to optimize the parameters. In the appendix of NODE [23], more details about the algorithm are provided.

**Algorithm 1 Algorithm of the NMGM.**

**Input:** raw data \((X_1^{(0)}, X_2^{(0)}, \ldots, X_N^{(0)})\) where \(X_1^{(0)} = (x_1^{(0)}(1), x_1^{(0)}(2), \ldots, x_1^{(0)}(m))\); initialize \(\theta^{(0)}\); function \(f_\theta; t_s; \lambda; \text{MaxIterations MI; Threshold Th;}

**Output:** \(\hat{X}_1^{(0)}, \hat{X}_2^{(0)}, \ldots, \hat{X}_N^{(0)}\)

1. initialization
2. Compute 1-AGO sequence: \(X_j^{(1)} = (x_j^{(1)}(1), x_j^{(1)}(2), \ldots, x_j^{(1)}(m))\) where \(x_j^{(1)}(i) = \sum_{k=1}^{m} x_j^{(0)}(k), i = 1, 2, \ldots, m\)
3. **while** \(k \leq \text{MI} \text{ AND Loss} > \text{th} \textbf{do}
4. Solve \(\frac{dx_j^{(1)}}{dt} = f_\theta(x_j^{(1)}, t_s)\) to get \((\hat{X}_1^{(0)}, \hat{X}_2^{(0)}, \ldots, \hat{X}_N^{(0)})\)
5. Get \(a(t) = \text{ODESolver}(a(t_n), t, \frac{\partial f_\theta(z(t), \theta)}{\partial \theta})\)
6. Get \(\nabla_\theta \text{Loss} = \int_{t_1}^{t_0} a(t)^\top \frac{\partial f_\theta(z(t), \theta)}{\partial \theta} dt\)
7. Iterate \(\theta : \theta^{(k+1)} = \theta^{(k)} - \lambda \nabla_\theta \text{Loss}\)
8. Solve \(\frac{dx_j^{(1)}}{dt} = f_\theta(x_j^{(1)}, t_s)\) to get \((\hat{X}_1^{(1)}, \hat{X}_2^{(1)}, \ldots, \hat{X}_N^{(1)})\)
9. Obtain prediction value \(\hat{X}_1^{(0)} : \hat{x}_1^{(0)}(k) = \hat{x}_1^{(1)}(k) - \hat{x}_1^{(1)}(k-1), k = 1, 2, \ldots, m - 1\)
10. **return** \((\hat{X}_1^{(0)}, \hat{X}_2^{(0)}, \ldots, \hat{X}_N^{(0)})\).

In summary, the conventional GM(1, N) model adopts the least squares method for parameter training, whereas the NMGM model relies on a neural network design and thus applies gradient descent for training. Furthermore, as the neural network defines the function \(f_\theta\) within the whitening equation, integrating the attributes of dependent variables and associated factors into the network’s architecture and training greatly enhances the model’s accuracy.

### 3.3. Performance Evaluation

Once a model is used to generate predictions, it is necessary to employ suitable validation criteria to evaluate the model’s prediction accuracy and performance. The chosen verification criteria should effectively capture the difference between predicted and actual values. To validate and compare the accuracy of model fitting and prediction, the absolute percentage error (APE) and mean absolute percentage error (MAPE) are employed [11,22]. \(x_1^{(0)}\) refers to the initial sequence values input into the model, while \(\hat{x}_1^{(0)}\) represents the fitted value obtained from the model.

APE computes the absolute difference between predicted and actual values, normalized by the actual value. MAPE calculates the average of these absolute differences across the dataset. These measures aid in evaluating the model’s performance in accurately fitting observed data and predicting future trends. The formulas for calculating APE and MAPE are as follows [22]:

\[
\text{APE} = \frac{\left| x_1^{(0)}(i) - \hat{x}_1^{(0)}(i) \right|}{x_1^{(0)}(i)} \times 100\%, i = 2, 3, \ldots, n
\]  

(20)

\[
\text{MAPE} = \frac{1}{m - 1} \sum_{i=2}^{m} \left| \frac{x_1^{(0)}(i) - \hat{x}_1^{(0)}(k)}{x_1^{(0)}(i)} \right| \times 100\%.
\]  

(21)
4. Experiments

In this section, two cases are used, with reference to [10], to evaluate the performance of NMGM. We compared NMGM’s simulation and prediction errors to those of other widely used grey prediction models, including GM(1, N) [6], FGM(1, N) [35], FDGM(1, N) [36], and NODGM [31]. In our experiments, we adopt the MAPE and APE, which were defined in Section 3.3 as the criterion to evaluate the performance of the models. We implement these experiments using the torchdiffeq libraries of the Python language [23].

In our experiment, we define $f_\theta$ as a three-layer fully connected neural network, incorporating the exponential linear unit (ELU) as the activation function. Consequently, we use the NODE as the whitening equation for the NMGM model. We employ the dopri5 algorithm as our ODE solver to obtain the solution of the equation. Dopri5 is an adaptive step solver that uses the Runge–Kutta (RK) [33] numerical approach, which is one of the most-used neural ODE solvers. The torchdiffeq library automates the parameter updates for training the loss function by seamlessly executing forward passes and backpropagation. This process ensures the automatic adjustment of parameters, streamlining the optimization of the loss function during training. The ADAM gradient optimization technique was used for this study. Considering the generalization of the neural network model, like the NODGM and NCDMGM models, we ran these models ten times and took the mean value as the result.

4.1. China’s per Capita Energy Consumption Prediction

The experiment in this section was carried out to predict energy consumption per capita in China using publicly available historical data. As shown in Table 1, we used China’s per capita energy consumption data from 2012 to 2021, which comes from the Energy Statistical Yearbook 2022 [37]. The total energy unit is kilos of standard coal, while electricity is measured in kilowatt-hours and coal and oil in kilograms. Before constructing our models, we divided the dataset into two distinct segments. We used the period from 2012 to 2018, shown in Table 1, as the test dataset. Subsequently, the latest data from 2019 to 2021 are used as the test dataset to evaluate and compare the prediction accuracy. When selecting predictive indicators, we prioritize total energy consumption per capita as the primary system characteristic sequence, categorizing the remaining indicators as influencing factors.

Table 1. Per capita energy consumption in China (2012–2021).

<table>
<thead>
<tr>
<th>Year</th>
<th>Total Energy</th>
<th>Electricity</th>
<th>Coal</th>
<th>Oil</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012</td>
<td>2977</td>
<td>3684</td>
<td>3018</td>
<td>354</td>
</tr>
<tr>
<td>2013</td>
<td>3058</td>
<td>3976</td>
<td>3113</td>
<td>367</td>
</tr>
<tr>
<td>2014</td>
<td>3122</td>
<td>4215</td>
<td>3015</td>
<td>378</td>
</tr>
<tr>
<td>2015</td>
<td>3146</td>
<td>4205</td>
<td>2998</td>
<td>406</td>
</tr>
<tr>
<td>2016</td>
<td>3181</td>
<td>4410</td>
<td>2802</td>
<td>416</td>
</tr>
<tr>
<td>2017</td>
<td>3285</td>
<td>4721</td>
<td>2803</td>
<td>433</td>
</tr>
<tr>
<td>2018</td>
<td>3364</td>
<td>5098</td>
<td>2833</td>
<td>444</td>
</tr>
<tr>
<td>2019</td>
<td>3463</td>
<td>5318</td>
<td>2855</td>
<td>458</td>
</tr>
<tr>
<td>2020</td>
<td>3531</td>
<td>5501</td>
<td>2869</td>
<td>463</td>
</tr>
<tr>
<td>2021</td>
<td>3724</td>
<td>6032</td>
<td>3042</td>
<td>484</td>
</tr>
</tbody>
</table>

Table 2 displays the forecast results of the GM, FDGM, FGM, NOGDM, and NMGM models for the data presented in Table 1. The most optimal outcomes are indicated in bold, while the second most optimal outcomes are underlined. In addition to that, it shows the MAPE of all models based on the data used for training and testing. As indicated in Table 2, the NMGM model outperforms all other grey models in the MAPE. The NMGM model achieves an MAPE of 0.2537% on the training data and 0.7381% on the test data, surpassing all other tested models. The NMGM model exhibits a superior generalization capacity in comparison with the other grey models. Simultaneously, we specifically noted
that the predictive accuracy of the multi-variable NMGM surpassed that of the uni-variable NODGM in both the training and test sets. The major reason for this is that the model takes into account the impact factor. In order to make a more accurate comparison of the discrepancies between these models, the MAPE findings are compared in Figure 3, while the APE results are displayed in Figure 4. When comparing MAPE and APE, it is evident that the NMGM model exhibits superior prediction accuracy in contrast to other grey models.

### Table 2. Modeling result of per capita energy consumption in China. The best is bold, the next best is underlined.

<table>
<thead>
<tr>
<th>Year</th>
<th>GM(1,N)</th>
<th>FDGM(1,N)</th>
<th>FGM(1,N)</th>
<th>NODGM</th>
<th>NMGM</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012</td>
<td>2977</td>
<td>2977</td>
<td>2977</td>
<td>2977</td>
<td>2977</td>
</tr>
<tr>
<td>2013</td>
<td>2910.35</td>
<td>2993.94</td>
<td>2993.62</td>
<td>3033.55</td>
<td>3054.94</td>
</tr>
<tr>
<td>2014</td>
<td>3156.44</td>
<td>3072.23</td>
<td>3071.91</td>
<td>3093.55</td>
<td>3098.41</td>
</tr>
<tr>
<td>2015</td>
<td>3136.18</td>
<td>3152.59</td>
<td>3149.20</td>
<td>3157.22</td>
<td>3137.33</td>
</tr>
<tr>
<td>2016</td>
<td>3160.87</td>
<td>3234.98</td>
<td>3216.73</td>
<td>3224.75</td>
<td>3194.37</td>
</tr>
<tr>
<td>2017</td>
<td>3280.37</td>
<td>3319.57</td>
<td>3321.35</td>
<td>3296.41</td>
<td>3271.18</td>
</tr>
<tr>
<td>2018</td>
<td>3387.67</td>
<td>3406.40</td>
<td>3376.2</td>
<td>3372.41</td>
<td>3367.31</td>
</tr>
</tbody>
</table>

MAPE: 1.4503% 1.1589% 0.9082% 0.5298% 0.2537% 0.2537% 1.2702% 1.106% 0.9927% 1.794% 0.7381%

As illustrated in Figure 5, the per capita energy consumption in China exhibits a clear upward trend. Along with societal development, energy consumption continues to persist, and the sustained growth in energy usage is expected to remain stable. The predictions in this study can help government departments make better decisions and policies by providing more accurate data.

![Figure 3. MAPE of five grey models on China’s per capita energy consumption.](image-url)
4.2. Forecast of China’s Total Amount of Renewable Energy

This section uses the new MGM model and four other comparative models to predict China’s total renewable energy production. The citation for the data can be found in the IRENA Renewable Capacity Statistics 2023 report [38] and the units for all data are Mw. This study forecasts by using data from the renewable energy statistics table from 2013 to 2022, as displayed in Table 3. As in the previous case, we compare our model with the GM, FDGM, FGM, and NOGDM models. The dataset spans from 2013 to 2019 for training and 2020 to 2022 for testing. Total renewable energy serves as the core feature, while other factors complement the analysis in both the training and testing phases. This setup allows a holistic examination of renewable energy’s interrelation with various supplementary factors, enriching the model’s predictive capacity.

<table>
<thead>
<tr>
<th>Year</th>
<th>Total</th>
<th>Hydropower</th>
<th>Solar Energy</th>
<th>Bioenergy</th>
<th>Wind Energy</th>
<th>Biogas</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>2013</td>
<td>359,516</td>
<td>280,440</td>
<td>17,759</td>
<td>6089</td>
<td>76,731</td>
<td>193</td>
<td>210.53</td>
</tr>
<tr>
<td>2014</td>
<td>414,651</td>
<td>304,860</td>
<td>28,399</td>
<td>6653</td>
<td>96,819</td>
<td>310</td>
<td>249.13</td>
</tr>
<tr>
<td>2015</td>
<td>479,103</td>
<td>319,530</td>
<td>43,549</td>
<td>7977</td>
<td>131,048</td>
<td>331</td>
<td>297.73</td>
</tr>
<tr>
<td>2016</td>
<td>541,016</td>
<td>332,070</td>
<td>77,809</td>
<td>9269</td>
<td>148,517</td>
<td>350</td>
<td>342.69</td>
</tr>
<tr>
<td>2017</td>
<td>620,856</td>
<td>343,775</td>
<td>130,822</td>
<td>11,234</td>
<td>164,374</td>
<td>454</td>
<td>370.39</td>
</tr>
<tr>
<td>2018</td>
<td>695,463</td>
<td>352,261</td>
<td>175,237</td>
<td>13,235</td>
<td>184,665</td>
<td>630</td>
<td>401.15</td>
</tr>
<tr>
<td>2019</td>
<td>788,844</td>
<td>358,040</td>
<td>204,971</td>
<td>16,537</td>
<td>209,582</td>
<td>799</td>
<td>431.91</td>
</tr>
<tr>
<td>2020</td>
<td>899,625</td>
<td>370,280</td>
<td>253,964</td>
<td>23,583</td>
<td>282,113</td>
<td>1285</td>
<td>462.67</td>
</tr>
<tr>
<td>2021</td>
<td>1,020,234</td>
<td>390,920</td>
<td>306,973</td>
<td>29,753</td>
<td>328,973</td>
<td>1711</td>
<td>493.43</td>
</tr>
<tr>
<td>2022</td>
<td>1,160,799</td>
<td>413,500</td>
<td>393,032</td>
<td>34,088</td>
<td>365,964</td>
<td>1928</td>
<td>524.19</td>
</tr>
</tbody>
</table>

The MAPE for each model in both the training and test datasets is shown in Table 4. The table also shows the predictions made by the GM, FDGM, FGM, NOGDM, and NMGM models using the data from Table 3. The optimal results are presented in bold, and the second-best results are underlined. Table 4 shows that, out of all the evaluated models, the NMGM model has the best MAPE with 0.9566% on the training data and 0.7896% on the test data. Thus, when contrasted with other grey models, the NMGM model demonstrates a superior performance. For a more intuitive comparison of the distinctions between these models, the MAPE and APE results are displayed in Figure 6 and Figure 7, respectively. In contrast to alternative grey models, the NMGM model, as defined in this paper, exhibits superior predictive accuracy when compared to MAPE and APE.

Table 4. Modeling result of renewable energy data in China. The best is bold, the next best is underlined.

<table>
<thead>
<tr>
<th>Year</th>
<th>GM(1,N)</th>
<th>FDGM(1,N)</th>
<th>FGM(1,N)</th>
<th>NODGM</th>
<th>NMGM</th>
</tr>
</thead>
<tbody>
<tr>
<td>2013</td>
<td>359,516</td>
<td>359,516</td>
<td>359,516</td>
<td>359,516</td>
<td>359,516</td>
</tr>
<tr>
<td>2014</td>
<td>328,586</td>
<td>416,479</td>
<td>415,830</td>
<td>415,805</td>
<td>415,042</td>
</tr>
<tr>
<td>2015</td>
<td>483,737</td>
<td>473,119</td>
<td>472,310</td>
<td>477,083</td>
<td>477,580</td>
</tr>
<tr>
<td>2016</td>
<td>532,637</td>
<td>537,463</td>
<td>536,461</td>
<td>543,976</td>
<td>544,277</td>
</tr>
<tr>
<td>2017</td>
<td>607,781</td>
<td>610,558</td>
<td>609,326</td>
<td>617,399</td>
<td>617,482</td>
</tr>
<tr>
<td>2018</td>
<td>683,109</td>
<td>693,594</td>
<td>692,087</td>
<td>698,438</td>
<td>697,311</td>
</tr>
<tr>
<td>2019</td>
<td>748,340</td>
<td>787,923</td>
<td>786,090</td>
<td>788,297</td>
<td>788,556</td>
</tr>
<tr>
<td></td>
<td>MAPE</td>
<td>4.7564%</td>
<td>1.351%</td>
<td>1.4129%</td>
<td>0.9566%</td>
</tr>
<tr>
<td>2020</td>
<td>926,350</td>
<td>895,080</td>
<td>882,860</td>
<td>888,266</td>
<td>892,905</td>
</tr>
<tr>
<td>2021</td>
<td>1,061,307</td>
<td>1,036,811</td>
<td>1,004,132</td>
<td>999,732</td>
<td>1,011,086</td>
</tr>
<tr>
<td>2022</td>
<td>1,212,928</td>
<td>1,176,097</td>
<td>1,145,877</td>
<td>1,124,203</td>
<td>1,152,380</td>
</tr>
<tr>
<td></td>
<td>MAPE</td>
<td>3.8291%</td>
<td>1.1493%</td>
<td>1.5758%</td>
<td>0.7896%</td>
</tr>
</tbody>
</table>

Recently, the Chinese government has actively promoted the growth of the new energy sector, resulting in a consistent annual increase in China’s overall renewable energy capacity. As shown in Figure 8, our experimental results can verify this very well. More accurate prediction results can provide more effective support for the government’s correct decision-making.

According to the above analysis, we can conclude that the NMGM model shows excellent adaptability and accuracy. In summary, depending on the generalization ability of the neural network, the proposed model can successfully recognize and understand the nonlinear features of sparse data, enabling the model to function well in most situations.
Figure 6. MAPE of five grey models to China’s renewable energy.

Figure 7. APE of five grey models to China’s renewable energy.

Figure 8. Predictive curves from five grey models for China’s renewable energy.
5. Conclusions

This paper introduces the neural ordinary differential multivariate grey model (NMGM) as a novel grey model for time series forecasting, which is particularly designed to tackle the challenges posed by sparse data in grey prediction. NMGM is distinguished by its utilization of neural ordinary differential equations (NODE) during training, ensuring a robust structural compatibility that significantly enhances its predictive capabilities. The NMGM model is more adaptive than prior grey models with artificially specified structures, and it can train an appropriate model from a range of sample data. Meanwhile, unlike the least squares technique, the method optimizes parameters using forward and backpropagation, preventing overfitting and improving the model's prediction performance. As a result, the NMGM performs better in terms of generalization and accuracy than models built using the least squares method.

Our new model outperforms four existing grey prediction models in two practical cases, affirming its superior efficacy and performance. Similar to previous grey models, our model can be used in scenarios where the data have some regularity and trends, and the amount of data is relatively small, such as an air pollution index, traffic flow, sales forecast, energy consumption, and demographics. The experimental results further highlight the ability of the NMGM model to accurately predict time series with a small number of samples. With more accurate predictions, our models can provide governments or commercial organizations with more accurate data to make better decisions.

Our novel model enhances the accuracy of multivariate grey prediction models, contributing to the advancement of the theoretical underpinnings of grey prediction models. The advantages of this novel approach will be comprehensively explored across various real-world scenarios. In the future, we plan to promote the model’s application in areas such as sales forecasting and traffic analysis. Furthermore, we will investigate greater-depth model optimization based on challenging scenarios, such as typical situations including data with time delay or periodicity.

Author Contributions: Conceptualization, Q.L. and X.Z.; methodology, Q.L.; software, Q.L.; validation, X.Z.; investigation, Q.L.; writing—original draft preparation, Q.L.; writing—review and editing, X.Z.; supervision, X.Z.; funding acquisition, X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the [National Natural Science Foundation of China] grant number [62372366].

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. These data can be found here: https://data.stats.gov.cn/easyquery.htm/ (accessed on 8 November 2023) and https://www.irena.org/Publications/2023/Mar/Renewable-capacity-statistics-2023 (accessed on 9 November 2023). Due to URL permission issues, it is available by the authors on request.

Acknowledgments: The authors would like to thank Liu Bingqing and Wang Chuntao for their invaluable contribution to project management and the validation of the results.

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

References


11. Li, X.; Li, N.; Ding, S.; Cao, Y.; Li, Y. A novel data-driven seasonal multivariable grey model for seasonal time series forecasting. *Inf. Sci.* 2023, **642**, 119165. [CrossRef]. [CrossRef]


14. Xiong, P.; He, Z.; Chen, S.; Peng, M. A novel GM(1, N) model based on interval gray number and its application to research on smog pollution. *Kybernetes* 2020, **49**, 753–778. [CrossRef]


18. Zeng, L. Analysing the high-tech industry with a multivariable grey forecasting model based on fractional order accumulation. *Kybernetes* 2018, **48**, 1158–1174. [CrossRef]


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.