Enhancing Stock Price Prediction Method Based on CNN-LSTM hybrid model

Yuqi Ding*

Zhongnan University of Economics and Laws, Wuhan, China

*Corresponding author: 120090533@link.cuhk.edu.cn

Abstract. Stock price prediction means using the previous data to predict future trend. With the development of deep learning technologies in artificial intelligence area, many works try to use single CNN or single LSTM to make prediction. However, both these two models have their own weakness and strengths. In this paper, we try to make a CNN-LSTM hybrid model to achieve a better performance on stock price prediction. It proposes that the CNN-LSTM hybrid model predicts stock prices using RMSE to measure higher accuracy than a single CNN and a single LSTM model. The experiment shows a great ability on generalization and provide reliable results, which can apply to other stocks to make prediction and give suggestions on investing strategy. The CNN-LSTM model can play the respective advantages of CNN and LSTM model to improve the accuracy of stock price prediction and has better performance in both short-term prediction and long-term trend.

Keywords: Convolutional Neural Network, Long Short Term Memory, Stock Price Prediction.

1. Introduction

With the development of modern financial theory, the popularization and improvement of computer technology, and the decline in investment transaction costs, the shortcomings of "emotional" transactions that lead to wrong judgments in decision-making and unexpected losses have also been exposed. Quantitative investment in the transaction process There is almost no human intervention, which avoids this shortcoming. It slowly appears in the financial system and becomes an indispensable and important part. The stock market has attracted a large number of investors and capital inflows. On the one hand, the stock price trend can reflect the overall development of a country or region's economy; on the other hand, the stock price tendency can also affecte the direct income of the majority of investors, so it has attracted widespread attention. Quantitative investment refers to applying mathematical models to the selection and trading of securities based on the laws and economic principles obtained from market observation, referring to historical data, combining Artificial intelligence means using computers to conduct transactions through computer programming, forming a more rational and scientific quantitative trading strategy.[1] In the 1970s and 1980s, quantitative investment gradually emerged.

The world's first quantitative trading fund was established by Edward Thorpely, and it has achieved a good performance of continuously outperforming the S&P index for 11 consecutive years. Traditional quantitative investment's prediction of stock prices is often based on the experience gained by individuals in long-term investment in the stock market, and the risk control brought by experience and the long-term prediction ability of stock prices are weak, and it is not conducive to dissemination. With artificial intelligence Continuous development, such as the introduction of deep learning technology into the field of quantitative investment for use, with its excellent predictive ability, it has gained widespread attention.

The main research work of this paper: At present, multi-factor stock strategy, futures CTA strategy, hedging arbitrage strategy and high-frequency trading strategy are commonly used quantitative investment strategies, but with the increasing number of quantitative investment competitors in the market, the use of such strategies The effect of obtaining income is gradually weakened. If you want to obtain excess income in the market, you need to continuously innovate quantitative investment strategies to achieve unexpected profits. Under the catalysis of this situation, the financial system has gradually set off the formation of deep learning algorithms. The upsurge of quantitative investment strategies, such as the quantitative investment strategies formed by using LSTM and CNN, extracting data that is not easy to observe characteristics, and assist strategy execution according to its characteristics, but due to the noise, nonlinearity and instability of the financial market, the market is not a real efficient market, so the deep learning strategy that simply analyzes historical market data can be used to deal with unexpected situations that have a major impact on the financial sector, have relatively large limitations, making the overall strategy less effective.

This paper focuses on the current status and trends, and proposes a machine learning quantitative trading strategy that takes into account both financial market sentiment factors and financial market historical transaction data. This strategy combines two different deep learning models of machine neural network LSTM and CNN, extracts multi-dimensional quantitative factors, and builds a set of balanced machine learning-based stock quantitative multi-factor investment strategies. The unique quantitative data features are extracted through the combination model, and a set of quantitative investment strategies based on deep learning stock price predictions are constructed. This strategy passes the Convolve historical data to extract relevant features, and use LSTM to better process time series data to predict stock price forecasts. Apply to backtest analysis and performance evaluation on the historical market of sp500 index constituent stocks, and find the effect of this strategy.

2. Method

2.1. CNN Network Structure

Convolutional neural network (CNN), as a special deep learning model, is different from general fully connected neural networks in that the neuron structure of each layer of the CNN model is not directly connected to the neuron structure of the previous layer. The CNN model has a unique convolutional layer structure, the filter in the convolutional layer can be used to capture the features of the input information, and the weights of neurons can be shared among the convolutional layers to speed up the model training by reducing the number of model weights (w,b). The neurons' weight in each convolutional layer can be shared, which speeds up the training of the model by reducing the number of model weights (w,b). A typical CNN model network structure consists of following four parts: an Input Layer, a Convolution Layer, a Pooling Layer, and an Output Layer.[2] Figure 1 illustrates the structure of a CNN basic model.

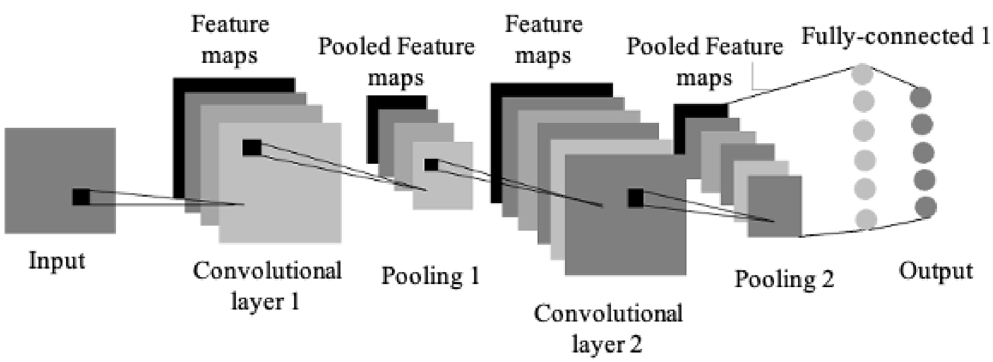

Fig 1. The structure of CNN

The most significant part of entire CNN model is the convolutional layer. Every single convolutional layer consists of plenty of convolutional units. The back propagation algorithm is used to optimize the convolutional units' parameter. The convolutional layer performs a convolutional operation on the input information covered by the filter, compresses the amount of information, and maps the result to the next layer in a certain number of steps. This process can also be defined as features extraction. After the convolutional process, a nonlinear transformation is generally used to prevent the lack of adequate expressive capability in the linear model. Most used nonlinear functions which are used to activate include following functions: ReLU function, tanh function, sigmoid function and so on [3]

The pooling layer is set to the next step after to minimize the acquired feature graph dimension, use the convolutional layer. [4] Pooling Layer is mainly for further feature sampling of the previous convolutional layer, including maximum pooling operation and average pooling operation, i.e., maximum feature extraction and average feature extraction, and finally the extracted features are output.

Output layer, also called fully-connected layer, act as a “categorizer” in the entire CNN model, which computes the final scores for each category by combining all local characteristics into global features.

In summary, the CNN model is a process of continuous dimensionality reduction and extraction of data features, which not only speeds up the training of the model, but also ensures the generalization performance by discarding irrelevant and redundant information and avoiding the learning of unnecessary features into the model. The excellent features of the CNN model also provide a new way of modeling for stock price prediction and yield enhancement.

2.2. LSTM Network Structure

Long Short Term Memory (LSTM) is a modified version of recurrent neural network (RNN). [5]RNN is greatly suitable for stock price prediction because it can examine trends in time-series data. RNN has the disadvantage of tha lack ability of saving state for long-term dependencies. Values are backpropagated in RNN, the slope decreases, and the problem of gradient vanishing follows. By storing states in the cell state, LSTM gets around this restriction. Additionally, the LSTM has a forget gate that determines whether or not prior state information is pertinent. Cell state stores the information if the Forget gate output is 1; otherwise, it ignores the information if the output is 0. Additionally employed in LSTM are input and output gates.

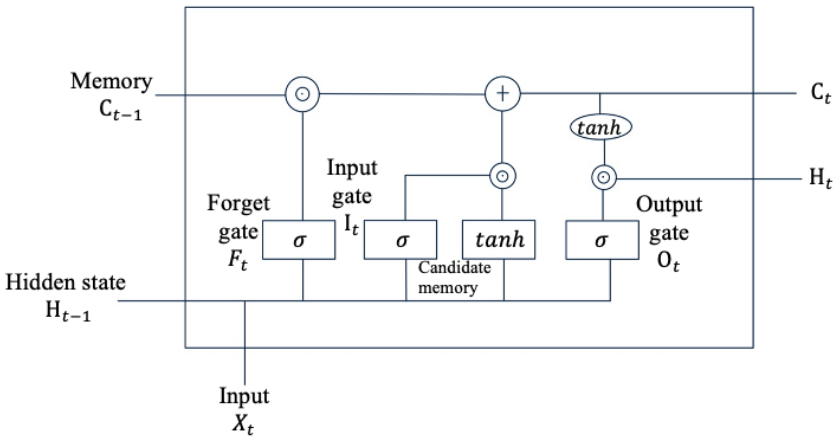

Fig 2. The structure of LSTM

Forget gate, the prior node h_{t-1} and incoming input X_t. are primarily intentionally forgotten.

$$f_t = \Sigma(w_f \cdot [h_{t-1}, X_t] + b_f$$

Some "information" may become "obsolete" over time for the unit state in the LSTM at the previous period. Some components are forgotten by selection of the LSTM unit from the former unit state, this is known as the "forget gate". To prevent too much memory from affecting how the neural network processes the current input.

Input gate mainly controls the input data feature X and the extent of the internal memory unit.

$$i_t = \Sigma(w_i \cdot [h_{t-1}, X_t] + b_i)$$

$$C_t = \tanh(w_c \cdot [h_{t-1}, X_t] + b_c)$$

The data at time t whether is contained into the control department in the unit state is determined by the input gate. In order to control these memories which are incorporated into the cell state, tanh activation function layer is first used to capture the useful information from the current vector.

In Output gate, cells which will be defined as output under the current stage. The LSTM unit is generally used to compute through control of an output gate, the neural layer's current output value. Add the results of multiplying it and C_t in the following equation to the result of f_t and C_{t-1} to get the value of $C_t$transferred to the following state, which is:

$$C_t = C_{t-1} \cdot f_t + i_t \cdot C_{t-1}$$

Firstly, the output layer converts the following cell state to the interval (-1,1) using the tanh function compression. Secondly, the output layer multiplies the cell state S_t of the C_t managed by the tanh function O_t to obtain the output h_t of the LSTM model at time t, which is:

$$O_t = \Sigma(W_o \cdot [h_{t-1}, X_t] + b_o$$

$$S_t = \tanh(C_t)$$

The deployment of input, forget, and output gates in the LSTM is clearly seen in Fig. 2. The information from earlier levels is filtered by the input gate, and the output that will be transferred to the following layer is filtered by the output gate.[6] Cell state in LSTM is

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \widetilde{C}_t$$

3. Experiment

3.1. Data Description

The data we are using to train our model is obtained from the Kaggle, shown as Table1. We are going to focus on one piece of stock –AAPL(Apple in America). This data set includes the everyday features of the stock from 1980-12-12 to 2022-12-12. The features include Date, Low, Open, Volume, High, Close and Adjusted Closing Price. Because the stock market in 2020 is often affected by the Covid19 epidemic, the phenomenon stock market crash, so to make the model's prediction performance more stable and enhance the model's generalization performance, this paper will not use the stock market trading data after 2020. For the experiment, we mainly focus on the data from 2013-01-01 to 2019-12-31.

Table 1. The example of data set

Date	Low	Open	Volume	High	Close	Adjust Close
2013-01-02	19.343929	19.779285	560518000	19.821428	19.608213	16.862829
2013-01-03	19.321428	19.567142	352965200	19.631071	19.360714	16.649979
2013-01-04	18.779642	19.177500	594333600	19.236786	18.821428	16.186195
2013-01-07	18.400000	18.642857	484156400	18.903570	18.710714	16.090986
2013-01-08	18.616072	18.900356	458707200	18.996071	18.761070	16.134289
...	...	...	...	...	...	...
2019-12-24	70.730003	71.172501	48478800	71.222504	71.067497	69.623230
2019-12-26	71.175003	71.205002	93121200	72.495003	72.477501	71.004585
2019-12-27	72.029999	72.779999	146266000	73.492500	72.449997	70.977646
2019-12-30	71.305000	72.364998	144114400	73.172501	72.879997	71.398903
2019-12-31	72.37997	72.482498	100805600	73.419998	73.412498	71.920563

Since this experiment uses stock data from the history transaction day to predict the closing price of stocks in the following days, by comparing the model results of whether to add volume and rise and fall, it is found that model performance is not improved by adding more features, but has been greatly affected, possibly because the addition of features with low correlation causes data redundancy, which reduces the stability of the model and leads to unsatisfactory results in the final prediction. Therefore, this article does not use the number of daily transactions of stocks with low correlation with the closing price and the daily stock rise and fall, and the final trading data of the S&P500 is the daily opening price, closing price, highest price and lowest price.

3.2. Model design

CNN is widely used in feature extraction because of its unique convolutional structure, which can better extract the features of the input data. However, Most often, classification problems are processed using CNN, and time series are processed less frequently. The most common application of LSTM is in the time series process, although it is not optimal for the extraction of useful features from data. LSTM can selectively save the input data to be used in the back-order of the model to prevent the early signals from vanishing during the processing..[7] LSTM can save the input data by selection to use during later process, thus preventing the early signal from fading out during processing, and is mostly used in time series processing, but it is not ideal for extracting effective features from the data. [8]The advantages of each model can make up for the limitations of the other model, therefore, by combining the characteristics of CNN and LSTM, this paper innovatively constructs a stock price forecasting model based on CNN-LSTM, in order to utilize the advantages of CNN for the extraction of the effective features of the data to reduce the noise of the time series, and then utilize the extracted data to improve the forecasting accuracy by using the advantages of the LSTM model for the processing of serial data.[9] Then the extracted data will be utilized to increase the prediction's precision by the superiority of the LSTM model in serial data processing to provide a more reliable basis for investment.

The structure of the CNN-LSTM hybrid model is shown in Figure 3, and the main structures are CNN and LSTM, including input layer, one-dimensional convolutional layer, pooling layer, LSTM layer, Dropout layer, fully connected layer and output layer.

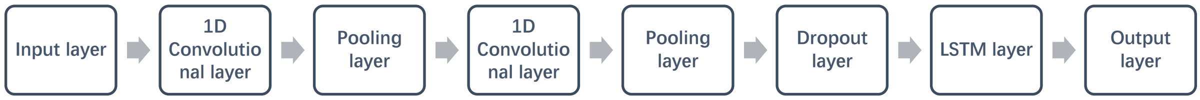

Fig 3. The structure of the CNN-LSTM hybrid model

After using python to form the above CNN-LSTM neural network model, its prediction ability cannot satisfy the prediction of future stock prices, so it is necessary to optimize the model performance by further parameterization to increase the prediction's precision. After parameterization, the final CNN-LSTM model is constructed with the following layer structure and parameter settings:

a) Input layer: For the input layer, we input the data (Open, Low, High, Close) into the CNN-LSTM model, where the numbers of neurons in this layer is 128.

b) 1D Convolutional layer: After input data, we use 1D convolutional layer to make features extraction through the Convolutional structure. The filters are set 256, and the kernel size is set 1. In order to keep the output dimension of the convolutional layer the same as the input dimension, we set the convolutional layer padding to the same. Then we choose the relu activation function and set the initialization as Glorot uniform.

c) Pooling layer: Compressing the data of the previous layer without destroying the main data features can effectively prevent model overfitting and improve the generalization ability of the model. The pooling window size (pool_size) is set to 2 and no padding is used which means the padding parameter is set to valid.

d) LSTM layer: the neurons number is set to 128, and the parameter return sequences is set to False in order to return a single value in the output sequence based on the current input data.

e) Fully-connected layer: the neurons number is set to 8, sigmoid function is selected for activation, and kernel initializer uses uniform initialization method.

f) Dropout layer: Dropout refers to the randomly discarded features in the model calculation process to improve the robustness of the model. In order to prevent the model from overfitting, this paper adds a dropout layer to the model, in which the ratio of randomly dropped features, i.e., the dropout ratio, is set as 0.1.

g) output layer: the neurons number is 1, the activation function is a sigmoid function, and the kernel_initializer uses a uniform initialization method.

After all parameters are set, we use python to train the train set and make prediction on the test set of the whole dataset. From the Figure4 and Figure5 shown below, it can be discovered that the differences between prediction stock price and the real stock price are small, and the performance of the CNN-LSTM model has a great generalization ability.

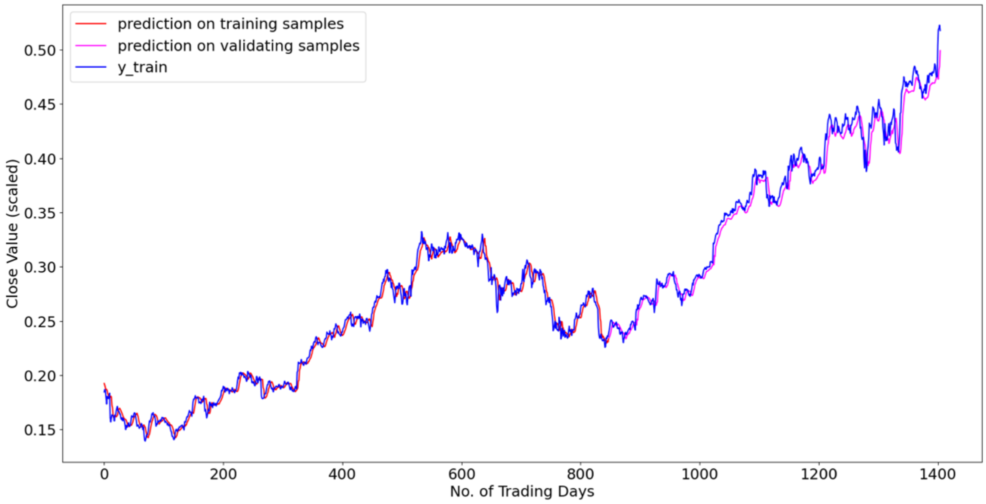

Fig 4. Train set prediction

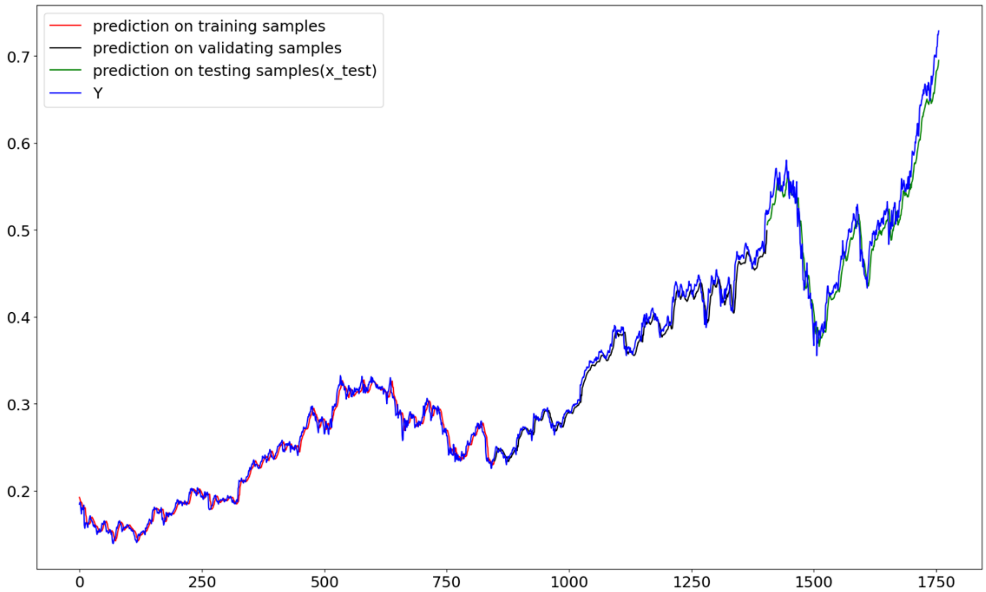

Fig 5. Train and Test ser prediction

4. Model Result

In this paper, RootMeanSquareError (RMSE) is used as the evaluation index of model prediction accuracy, and the calculation method of RMSE is shown in Equation below, the result shown on Table2.

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

Where $\hat{y}_i$ *is* the predicted value of the neural network, y_i is the true value, and n is the number of predictions, the value of RMSE can objectively reflect the degree of deviation between the predicted value and the true value of the model. The prediction performance of the model improves as the RMSE value decreases because the value predicted approaches the real value more closely. On the other hand, the prediction performance of the model degrades as the RMSE value increases because the forecasted value deviates further from the real value.[10]

Table 2. RMSE on Datasets

Model	Train set RMSE	Test set RMSE
CNN	0.012121978	0.076785950
LSTM	0.009935872	0.054784795
CNN-LSTM	0.008376274	0.018235675

When CNN-LSTM is used for modeling and prediction, and batch_size is set to 128, the average loss size which is related to the average MSE size, is 0.0084. It proves that the CNN-LSTM model has good fitting effect for the SP500 index. Observing the above two sets of fitted curves and comparing the predicted values with the real values, the following conclusions can be obtained:

The CNN-LSTM model predicts the closing price of AAPL stock price from SP500 and the real value of the index, which is effectively consistent with the trend, and the predicted and real values are close to each other with a low degree of deviation. S&P 500 is a market index composed of 500 stocks with the largest market capitalization and the most active turnover in American stock exchanges, which usually indicates the trend of the market's heavy-weighted stocks, and has a certain degree of representativeness for the change of the whole market's stock price, so this paper believes that the CNN-LSTM model has a certain degree of general applicability to the prediction of American stock prices in the stock market. Therefore, this paper believes that the CNN-LSTM model has some general applicability to the prediction of stock prices in American stock market.

5. Conclusion

Through the empirical comparative study in the previous section, the following conclusions can be drawn:

(1) Deep learning techniques in the field of finance through the use of deep learning can handle financial data more flexibly, provide more effective solutions to the problems under study, and also promote the possibility of discipline crossover to a certain extent.

(2) This paper tries to combine the superiority of CNN model for data feature extraction and LSTM model for sequential data processing in deep learning. The hybrid CNN-LSTM model is constructed for stock price prediction. The empirical results show that the model is effective, and it can be found that the CNN model has a higher prediction of the long-term trend of the stock, while the short-term upward and downward trend of the stock is poorly predicted. On the contrary, the LSTM model is less accurate in predicting the long-term trend of stocks but fits better for the short-term trend of stocks.

References

[1] E. Guresen, G. Kayakutlu and T. U. Daim, "Using artificial neural network models in stock market index prediction", Expert Systems with Applications, vol. 38, pp. 10389-10397, 2011.

[2] Y. Lecun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, et al., "Backpropagation applied to handwritten zip code recognition", Neural Computation, vol. 1, pp. 541-551, 1989.

[3] Y. Lecun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, et al., "Backpropagation applied Fangfang. Research on power load forecasting based on Improved BP neural network. Harbin Institute of Technology, 2011.

[4] Yamashita, R., Nishio, M., Do, R.K.G. et al. Convolutional neural networks: an overview and application in radiology. Insights Imaging 9, 611–629 (2018). https://doi.org/10.1007/s13244-018-0639-9

[5] Shi X, Chen Z, Wang H, et al. Convolutional LSTM network: A machine learning approach for precipitation nowcasting[J]. Advances in neural information processing systems, 2015, 28.

[6] S. Hochreiter and J. Schmidhuber, "Long short-term memory", Neural Computation, vol. 9, pp. 1735-1780, 1997.

[7] Y. Le Cun, Y. Bengio and G. Hinton, "Deep learning", Nature, vol. 521, pp. 436, 2015.

[8] H. Zhuang, R. Zhao, X. Luo and C. Yang, "Research on Quantitative Stock Selection Strategy Based on CNN-LSTM," 2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI), Chengdu, China, 2022, pp. 1142-1147, doi: 10.1109/PRAI55851.2022.9904291.

[9] X. Zhou, "Stock Price Prediction using Combined LSTM-C Model," 2021 3rd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI), Taiyuan, China, 2021, pp. 67-71, doi: 10.1109/MLBDBI54094.2021.00020.

[10] Willmott, C. and Matsuura, K.: Advantages of the Mean Abso- lute Error (MAE) over the Root Mean Square Error (RMSE) in assessing average model performance, Clim. Res., 30, 79–82, 2005.