Artificial neural network for constructing type Ia supernovae spectrum evolution model

Qiao-Bin Cheng, Chao-Jun Feng,* Xiang-Hua Zhai,† and Xin-Zhou Li‡

Shanghai United Center for Astrophysics (SUCA) and Department of Physics, Shanghai Normal University, 100 Guilin Road, Shanghai 200234, People's Republic of China

(Received 11 January 2018; published 20 June 2018)

We construct and train an artificial neural network called the backpropagation neural network to describe the evolution of the type Ia supernova spectrum by using the data from the CfA Supernova Program. This network method has many attractive features, and one of them is that the constructed model is differentiable. Benefiting from this, we calculate the absorption velocity and its variation. The model we constructed can well describe not only the spectrum of SNe Ia with wavelength range from 3500 Å to 8000 Å but also the light-curve evolution with phase time from −15 to 50 with different colors. Moreover, the number of parameters needed during the training process is much less than the usual methods.

DOI: 10.1103/PhysRevD.97.123530

I. INTRODUCTION

Type Ia supernova (SN Ia) is regarded as the standard candle to measure the distance on cosmological scales, since all of them have almost the same intrinsic brightness. By using the SN Ia distance indicator, the accelerating expansion of the universe was discovered in 1998 [1,2]. The standard model of SN Ia involves the thermonuclear disruption of a carbon-oxygen white dwarf star as it approaches the Chandrasekhar mass, and most type Ia supernovae are very similar in their photometric and spectroscopic properties. The SN Ia may be generated through the merging of components in close binaries [3]. A correlation between the peak luminosity and the shape of the early light curve was also found, with brighter objects having a lower rate of decline than dimmer ones [4], which is matched by a spectroscopic sequence.

The model of the spectroscopic sequence is constructed by a training process, such as the SALT2 model [5], in which the mean evolution of the spectral energy distribution (SED) sequence of SN Ia and its variation with color are modeled as a functional form; see Eq. (1) in [5]. During the training process of SALT2, one ends up with more than 3000 parameters [5] to fit, due to the obvious nonlinearities of the SED.

In past decades, among the various machine intelligence procedures, artificial neural network (ANN) methods have been established as powerful techniques to solve a variety of real-world problems because of its excellent learning capacity [6,7]. ANN is one of the popular areas of artificial intelligence research and also an abstract computational model based on the organizational structure of the human brain [6]. In this paper, we will construct and train an ANN called the backpropagation neural network (BPNN) to describe the evolution of the SN Ia spectrum. The inputs of our ANN are the phase (or time), the wavelength, and also the color from the data that will be described later, while the output is just the corresponding flux of the SN Ia. The SED we trained is almost the same as that in the SALT2 model. However, the employment of a neural architecture adds many attractive features:

(i) One does not need to assume a functional form of the SED model during the ANN's training process.
(ii) The SED via ANNs is differentiable, and then it is easily used in any subsequent calculations, e.g., the calculation of the absorption velocity gradients of Si II λ6355 line of SN Ia in the following.
(iii) The required number of model parameters is far less than the traditional methods.
(iv) The method can be realized in hardware, using neuroprocessors, and hence offer the opportunity to handle the real-time analysis of the SN Ia's spectra.
(v) The method can also be efficiently implemented on parallel architectures, such as the data parallelism, in which the data set is split into small ones and each of them is feed to a processor.

Si II λ6355 is one of the strongest features in optical/near-infrared spectra of SN Ia; the blueshift of its absorption minimum has often been used to diagnose the diversity among SN Ia [8]. Different subclass evolutions of Si II λ6355 absorption velocity ($v_{\rm abs}$) have been compared in Ref. [9], while the relation between its gradient and the parameter Δm_{15} of SN Ia has been studied in [10]. In this

*fengcj@shnu.edu.cn
†zhaixh@shnu.edu.cn
‡kychz@shnu.edu.cn

paper, we have studied the properties of $v_{\rm abs}$ and its gradients by using the spectrum evolution model after ANN's training. The results are fully consistent with those in [11,12].

The structure of this paper is as follows. In Sec. II, we construct the neural network, and then train it in Sec. III. The spectra data set we used for training is described in Sec. IV, while the training results are presented in Sec. V, and the $v_{\rm abs}$ and its gradient are also computed in this section. In Sec. VI the relation between $v_{\rm abs}$ and color is shown. Finally, discussions and conclusions are given in Sec. VII.

II. NETWORK CONSTRUCTION

In the following, the neural network will be constructed, which is called the backpropagation neural network. This kind of ANN has already been used in astronomy and physics, for example, to classify the type of a supernova (e.g., Ia, Ib, II, etc.) by using the neural network from the multifrequency observations of its light curve; see [13] and references therein. The structure (or topology) of the BPNN could be described as Fig. 1: where a few hidden layers and a few neural neurons in each layer are presented for limited space. In general, there could be totally $L+2$ layers in the network, and we will use $l=0,1,2,\ldots,L,L+1$ to denote them, where $l=0$ is called the input layer, $l=L+1$ is called the output layer, and others are all called the hidden levels.

Let N^l denote the number of neurons at level l, with N^0 and N^{L+1} also being called the dimensions of the input and the output, respectively. In this paper, we consider two cases with $N^0=2$ and 3, and $N^{L+1}=1$ for both.

Between the lth and the $(l-1)$th level, we have the weights

$$\mathbf{W}^l \equiv (W^i_j)^l \in \mathcal{R}^{N^L\times(N^{L-1}+1)} \tag{1}$$

for $l\geq 1$, where $i=1,2,\ldots,N^l$, $j=0,1,2,\ldots,N^{l-1}$, and here we have included the bias vector in $j=0$, namely,

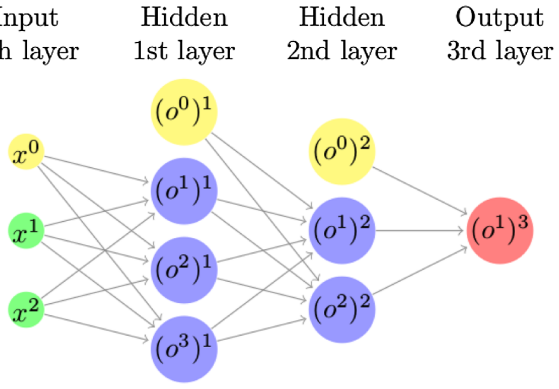

FIG. 1. Typical structure of an ANN.

$$(W^i_0)^l = (b^i)^l. \tag{2}$$

By construction, the input of the lth level is just the output of the $(l-1)$th level plus the bias, so we set the output of the lth level $(o^i_m)^l$ as the following:

$$(o^i_m)^0 = (n^i_m)^0 = X^i_m, \qquad l=0, \tag{3}$$

$$(n^i_m)^l = \sum_{j=0}^{N^{l-1}} (W^i_j)^l (o^j_m)^{l-1}, \qquad l\geq 1, \tag{4}$$

$$(o^i_m)^l = g_l((n^i_m)^l), \qquad l\geq 1, \tag{5}$$

where $i=1,2,\ldots,N^l$, $m=1,2,\ldots,M$, and M denotes the total number of the samples. Here, the function $g_l(x)$ is the activation function, which is often taken as a sigmoid function or the tangent hyperbolic function.

$$g(x) = \begin{cases} \frac{1}{1+e^{-x}} & \text{for sigmoid function,} \\ \tanh(x) & \text{for } \tanh(x), \end{cases} \tag{6}$$

between the hidden layers, i.e., $l\leq L$. Straightforward, we also have

$$\dot{g} = \frac{dg}{dx} = \begin{cases} g(1-g) & \text{for sigmoid function,} \\ 1-g^2 & \text{for } \tanh(x). \end{cases} \tag{7}$$

For the output layer, we take the activation function as a linear function $g(x)=x, \dot{g}=1$.

III. NETWORK TRAINING

To train the network, one needs to minimize a cost function after feeding training samples. The cost function is also called the error function, which describes the error between the output and training samples. In our cases, the cost function is given by $E=\mathbf{e}^{\mathbf{T}}\mathcal{C}^{-1}\mathbf{e}/2$, or more specifically,

$$E = \frac{1}{2}\left(\vec{F}^{\rm obs}-\vec{F}^{\rm ANN}\right)^T \mathcal{C}^{-1}\left(\vec{F}^{\rm obs}-\vec{F}^{\rm ANN}\right), \tag{8}$$

where $\mathcal{C}$ is the covariance matrix of the observational flux $\vec{F}^{\rm obs}$ and the flux $\vec{F}^{\rm ANN}$ is the output of BPNN for a supernova at the redshift z that depends on the phase $p\equiv(t-t^B_{\rm max})/(1+z)$, the wavelength λ, and maybe the color C. In the rest frame of a supernova, p is just the days after $(p>0)$ or before $(p<0)$ the date on which it has a maximum luminosity. Furthermore, the error for a given sample m in Eq. (8) reads

$$e_m = F^{\rm obs}_m - F^{\rm ANN}_m. \tag{9}$$

In the following, we will perform the Levenberg-Marquardt (LM) algorithm to train the BPNN. In each step s, the following weighted normal equations will be solved:

$$(\mathbf{J}^{\mathbf{T}}\mathcal{C}^{-1}\mathbf{J}+\mu\mathbf{I})(W_{s+1}-W_s)=-\mathbf{J}^{\mathbf{T}}\mathcal{C}^{-1}\mathbf{e}, \quad (10)$$

to update each weight between the layers of the network for next step $s+1$. Here $\mathbf{I}$ is the identity matrix, and μ is the combination coefficient that could be changed during the training procedure as the following: If the updated parameters $\Delta W = W_{s+1} - W_s$ computed from Eq. (10) lead to a reduction of the error, the updated parameters are accepted and μ is decreased by factor of 2 or so in the next. Otherwise, μ is increased by a factor of 2 or so, and Eq. (10) is solved again to process until the decreased error is found. This process repeats until the required precision is reached.

The Jacobi matrix $\mathbf{J} \in \mathcal{R}^{M\times N}$ is defined as

$$\begin{pmatrix} \frac{\partial e_1}{\partial W_1} & \frac{\partial e_1}{\partial W_2} & \cdots & \frac{\partial e_1}{\partial W_N} \\ \frac{\partial e_2}{\partial W_1} & \frac{\partial e_2}{\partial W_2} & \cdots & \frac{\partial e_2}{\partial W_N} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial e_M}{\partial W_1} & \frac{\partial e_M}{\partial W_2} & \cdots & \frac{\partial e_M}{\partial W_N} \end{pmatrix}, \quad (11)$$

where $N=\sum_{l=1}^{L+1} N^l(N^{l-1}+1)$ is the total number of weights (including bias).

For a given sample m we also have the following recurrence relation:

$$\frac{\partial e_m}{\partial \mathbf{W}^l}=\frac{\partial(F_m^{\rm obs}-F_m^{\rm ANN})}{\partial \mathbf{W}^l}=\frac{-\partial(o_m^{L+1})}{\partial \mathbf{W}^l}, \quad (12)$$

$$\frac{\partial(o_m^{L+1})}{\partial \mathbf{W}^l}=\delta^l(\mathbf{o}_m^{l-1})^T, \quad (13)$$

$$\delta^{L+1}=1, \quad (14)$$

$$\delta^l=\dot{G}^l(\mathbf{n}_m^l)(\bar{\mathbf{W}}^l)^T\delta^{l+1}, \qquad l\le L, \quad (15)$$

where $(\bar{\mathbf{W}}^l)$ are the weights not including the bias vector, i.e., $(\bar{W}_j^i)^l=(\bar{W}_j^i)^l, j\neq 0$, and

$$\dot{G}^l(\mathbf{n}_m^l)=\begin{pmatrix} \dot{g}^l((n_m^1)^l) & 0 & \cdots & 0 \\ 0 & \dot{g}^l((n_m^2)^l) & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \dot{g}^l((n_m^{N^l})^l) \end{pmatrix}. \quad (16)$$

In summary, by using the LM algorithm, the update rule of weights can be presented as

$$W_{s+1}=W_s-(\mathbf{J}^{\mathbf{T}}\mathcal{C}^{-1}\mathbf{J}+\mu\mathbf{I})^{-1}\mathbf{J}^{\mathbf{T}}\mathcal{C}^{-1}\mathbf{e}. \quad (17)$$

Actually, we also have tried the famous resilient back-propagation (RPROP) algorithm [14] to train BPNN, but it turns out that the LM one is much better, especially when the input dimension is higher than 2.

IV. DATA ANALYSIS

The data used in this paper are obtained through the CfA Supernova Program, and there are 2603 spectra of 462 low-z SNIa in total; see Refs. [9,15–21]. It should be noticed that some of the spectra in the CfA data have no flux error information and in this case we estimate the flux error by averaging the errors of the other data. For the training, the recalibration (or renormalization) of spectra from the corresponding supernova needs the knowledge of $t^B_{\rm max}$, i.e., the time when the flux has a maximum luminosity, so that spectra without the information of $t^B_{\rm max}$ will be disregarded. We only focus on $-15<p<50$, which is the most important phase range when dealing with the spectra. Beyond this range, we have either a few spectra or slowly varying ones. The spectra of SN Ia with color $C>0.8$ and those without a spectrum in range $-10<p<15$ are ignored, since the supernova has strong extinctions for large color. Actually, the impact is limited on training that includes these outside ranges in the sense that the trained weights of the network do not change much, but we do not think that the network could well describe the range outside of $[-15,50]$ in which there are only a few data points, and the color range $C>0.8$ in which the data have too strong extinctions to deal with when the covariance matrices are unknown. Here, the color of a SN Ia could be defined through the spectra:

$$C=2.5\log\left[\frac{\int_\lambda \lambda T_V(\lambda)S_{\rm SN}(\lambda)d\lambda}{\int_\lambda \lambda T_B(\lambda)S_{\rm SN}(\lambda)d\lambda}\right], \quad (18)$$

where $S_{\rm SN}(\lambda)$ is the de-redshift spectrum nearest $t^B_{\rm max}$, and $T_B(\lambda)$ and $T_V(\lambda)$ are the effective instrument transmissions in photometric bands B and V, respectively. The color defined from spectroscopy in the equation above has almost the same properties as that defined from the photometry; see Ref. [15].

Finally, there are 1787 spectra of 238 SN Ia (about 4600 thousand data points) that will be left for the BPNN training. We have shown the number and redshift distributions of these samples in Figs. 2 and 3. From the number distribution of the spectra in Fig. 2 (unfilled histogram), we find that many spectra were obtained near the max date $t^B_{\rm max}$ ($p=0$), but the average phase is about $\bar{p}\sim 10.5$ because the distribution has a long tail. The height of each column of the filled histogram in Fig. 2 indicates the number of SN Ia whose earliest spectrum is obtained at the phase p. It shows that more than half of the supernovae have the spectra earlier than the max date. From Fig. 3, one can see that almost all the SN Ia are below $z<0.05$, and the average value is $\bar{z}\sim 0.02$.

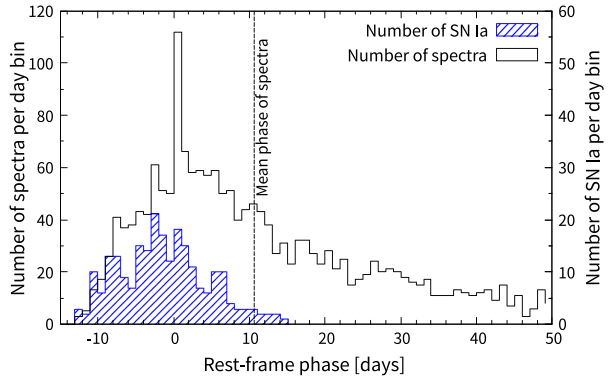

FIG. 2. Number distributions of the spectra and SN Ia.

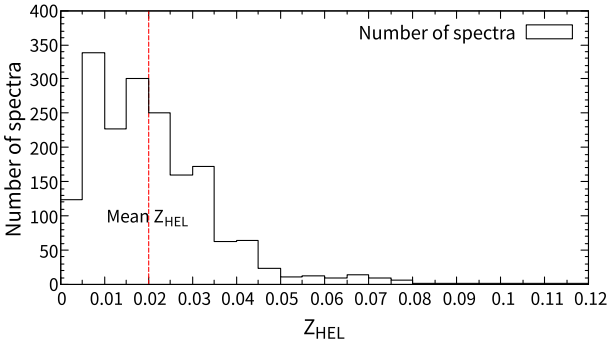

FIG. 3. Redshift distribution of the spectra.

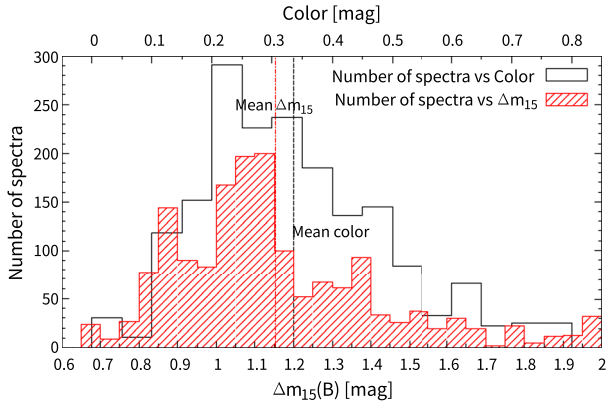

FIG. 4. Parameter distribution of the spectra.

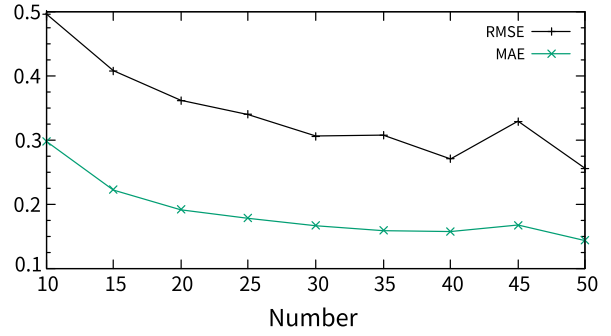

FIG. 5. The RMSE and MAE vs the neuron number in the hidden level for the BPNN with only one hidden level.

TABLE I. Results in Case I with different network structures. The first column is the topology of the network, the second column is the total number of the weights (including the bias), and the third and forth columns are the final RMSE and MAE, respectively. Note that the values of RMSE and MAE are already divided by the number of data points.

Structure	No.	RMSE	MAE
2-10-10-1	151	0.4964	0.2981
2-15-15-1	301	0.4077	0.2220
2-20-20-1	501	0.3549	0.1871
2-10-10-10-1	261	0.4063	0.2206
2-15-15-15-1	541	0.3491	0.1866
2-20-20-20-1	921	0.3059	0.1690
2-10-10-10-10-1	371	0.3769	0.1988
2-15-15-15-15-1	781	0.3016	0.1705
2-20-20-20-20-1	1341	0.2780	0.1665

The Δm_{15} and color C distributions of the spectra are also plotted in Fig. 4. The filled histogram tells us the average value of Δm_{15} is about $\Delta\bar{m}_{15} \sim 1.15$ mag, while the unfilled one shows the average value of $\bar{C} \sim 0.34$ mag.

To make the best use of the spectroscopic data, one needs to recalibrate or renormalize the data by using the corresponding SN Ia photometry, since in general the photometry is much better calibrated and uniform than its spectra. The recalibration is performed by multiplying each input spectrum by the factor

$$f_{\rm SN}^{p} = \frac{\bar{F}_R^{\rm obs}(p)}{\int_\lambda \lambda T_R(\lambda) S_{\rm SN}(\lambda)_p d\lambda}, \tag{19}$$

where $\bar{F}_R^{\rm obs}(p)$ is the average value of the flux in the R band filter.

V. TRAINING RESULTS

The output flux of BPNN $\vec{F}^{\rm ANN}$ in Eq. (8) could depend on either two inputs, i.e., phase p and wavelength λ, or three inputs, phase p, wavelength λ, and color C. In the following, the networks of these two cases will be trained.

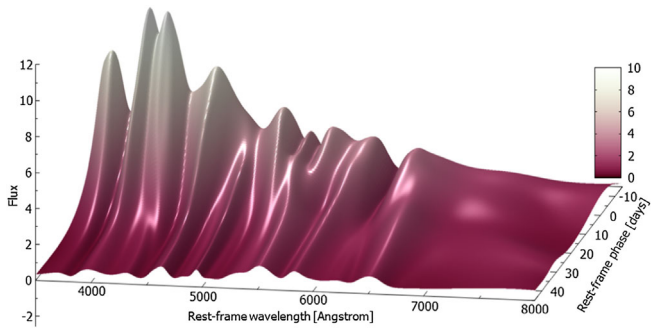

FIG. 6. Three-dimensional vision: A best-trained spectrum evolution model.

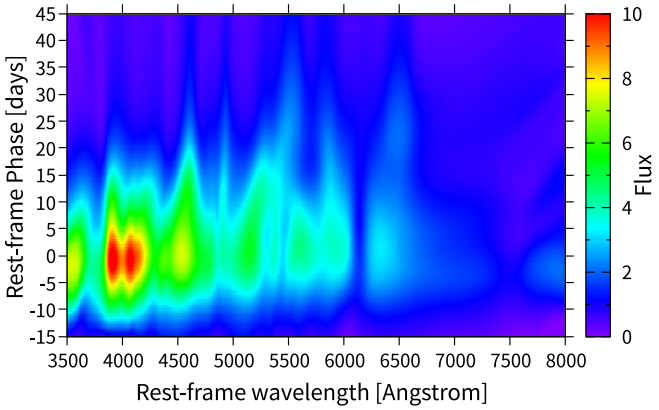

FIG. 7. Contour: A best-trained spectrum evolution model.

For simplicity, we assume each hidden level of the network has the same number of neurons in both cases.

A. Case I: $\vec{F}^{\mathrm{ANN}} = \vec{F}^{\mathrm{ANN}}(p,\lambda)$

In this case, the network will be trained by using the spectra in the wavelength range between 3500 and 8000 Å with a bin interval of 10 Å. In each interval, we average the flux weighted by its variation. There are about 57 000 data points in total.

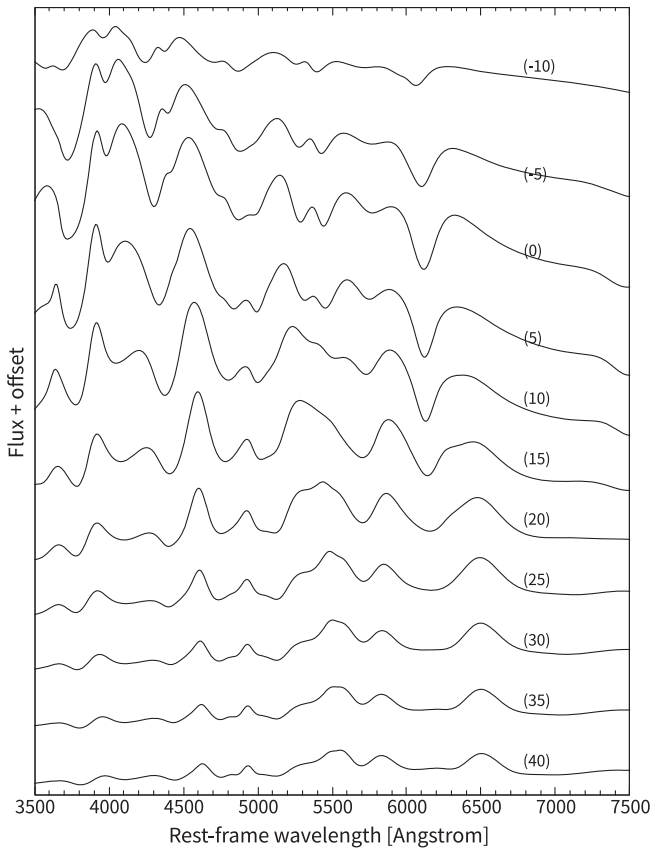

FIG. 8. Evolution of spectrum with phase, where the number in parentheses denotes the phase.

TABLE II. The data are split randomly to the training and test sets accordingly (80% to 20% or 70% to 30% split).

	Percent (%)	Number	RMSE	MAE
Training set	80	1416	0.4471	0.2699
Test set	20	371	0.4297	0.2643
Training set	70	1240	0.4303	0.2657
Test set	30	547	0.4778	0.2790

Figure 5 indicates the variation of the root mean squared error (RMSE) and the mean absolute error (MAE) vs the neuron number, where the BPNN has only one hidden level. One can clearly see that both RMSE and MAE decrease with the increase of the neuron number. We have also compared different structures of the network with three or four hidden levels; see Table I. The total number of parameters including weights and bias in the training network is less than 1000 except the last one, which has the most complicated structure. However, during the training process of SALT2, one usually ends up with more than 3000 parameters [5] to fit. To illustrate the ability of the BPNN method, a best-trained spectrum evolution model is presented in Figs. 6–8.

To see whether the training is overfit, we split the data into training and test sets by randomly choosing the training points from the all data set. A typical value of RMSE and MAE between these two sets is presented in Table II.

From Table II, one can see that the difference between the training and test sets is small. Therefore, it seems that there is no overfit issue for the networks we have trained.

B. Case II: $\vec{F}^{\mathrm{ANN}} = \vec{F}^{\mathrm{ANN}}(p,\lambda,C)$

It is well known that the K correction of the light curve mainly depends on spectral color [22,23], so in this case, the color will be included as one input of the network and

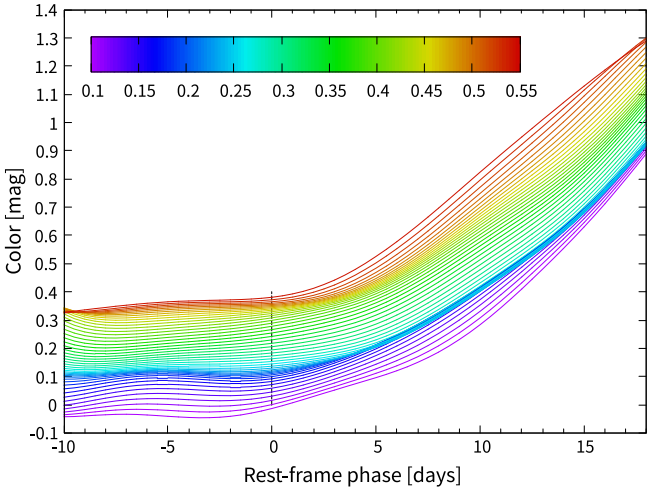

FIG. 9. Color vs phase.

the full set of spectra data will be used instead of the bin one.

After training, we find the behavior of the RMSE and MAE is the same as that in Case I. It is interesting that the color is almost a constant before the maximum luminosity day, after which it is proportional to the phase time; see Fig. 9. We also compared the trained spectra with those in SALT2 at different phases; see Fig. 10. One can see that the difference between them is very small at long wavelength, whereas at the short wavelength, the difference mainly depends on the color value; when $C \sim 0.15$, the model we trained is almost the same as that in SALT2. The RMSE/MAEs over the training and test sets for the SALT2 and BPNN models are also given in Table III.

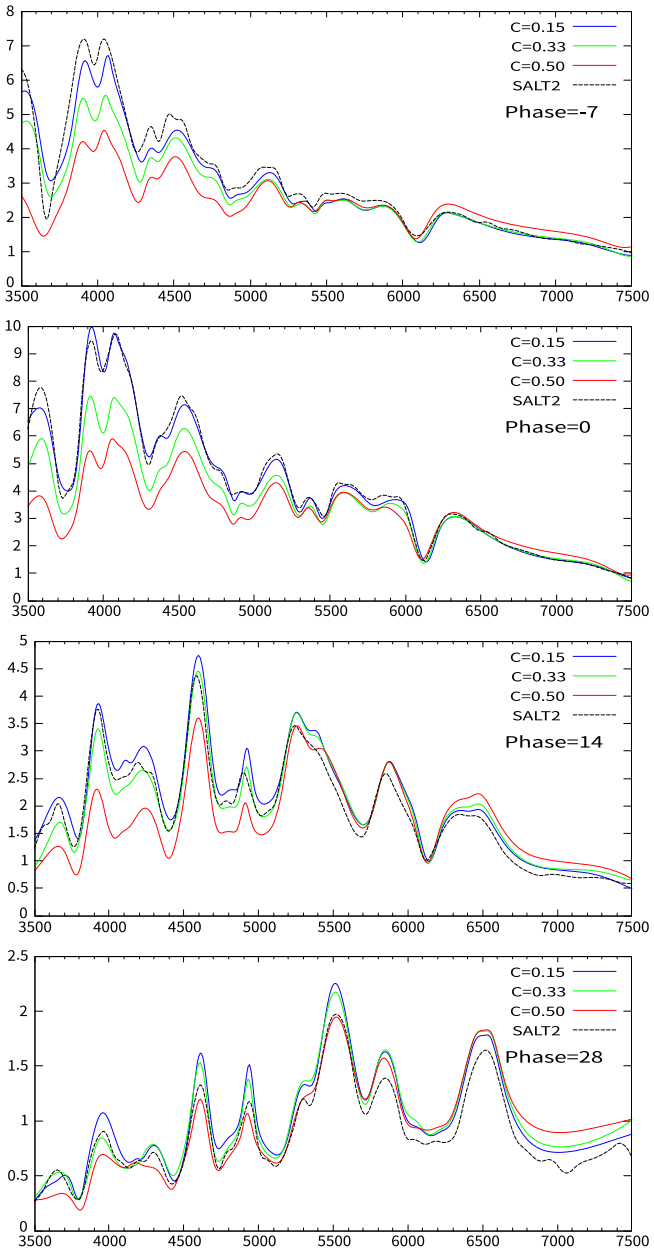

FIG. 10. Comparison of different color spectra. The black dashed line is SALT2 template spectra.

TABLE III. The RMSE/MAEs over the training and test sets for the SALT2 and BPNN models.

RMSE/MAE	SALT2	BPNN model
Training set (80%)	0.7098/0.4656	0.4471/0.2699
Test set (20%)	0.6685/0.4489	0.4297/0.2643
Training set (70%)	0.6900/0.4612	0.4303/0.2657
Test set (30%)	0.7270/0.4642	0.4778/0.2790

From Table III, one can see that the values of RMSE/MAE are slightly smaller for the BPNN model than that for the SALT2 model.

VI. APPLICATION: ABSORPTION VELOCITY

As we stated in the Introduction, one of the attractive features of applying the ANN method is that the output of the network is differentiable. One can benefit from this property to perform the consequence calculation, e.g., to calculate the absorption velocity gradients of Si II λ6355 line of SN Ia.

The absorption velocity is defined by using the relativistic Doppler formula:

$$\frac{v_{\rm abs}}{c} = \frac{(\lambda_{\rm abs}/\lambda_0)^2 - 1}{(\lambda_{\rm abs}/\lambda_0)^2 + 1}, \tag{20}$$

where λ_0 is the wavelength of the corresponding transition, and c is the speed of light. The wavelength corresponding to the maximum absorption $\lambda_{\rm abs}(p)$ can be obtained from the trained network; i.e., the value of wavelength minimizes $\vec{F}^{\rm ANN}$ at phase p (see Fig. 11). And then the changes of $v_{\rm abs}$ can be more efficiently obtained than those in Ref. [24] (see Fig. 12). The evolution is much faster at early times of the

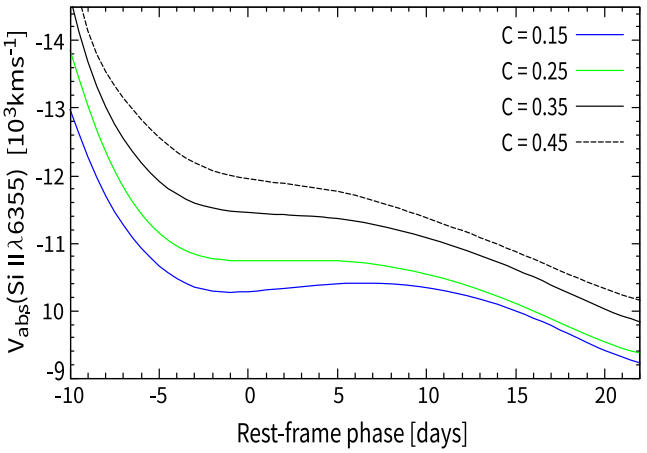

FIG. 11. Evolution of the Si II λ6355 absorption velocity with time for the spectra of different colors.

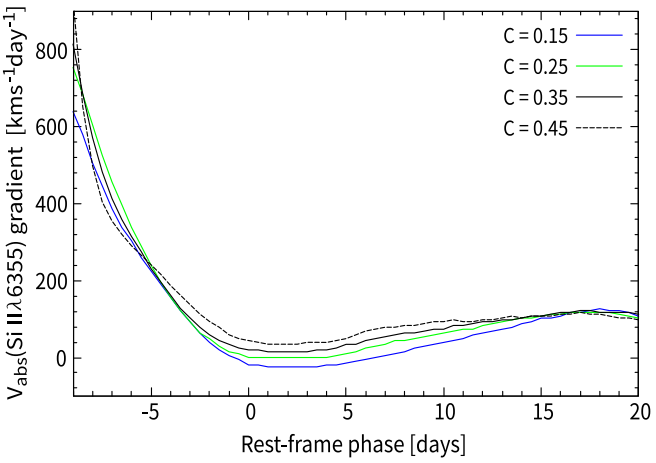

FIG. 12. $\partial V_{\rm abs}/\partial t$ as a function of phase.

supernova's explosion, which is about $v_{\rm abs} \sim 13,000$ km/s at phase $p \sim -8$. Later, the speed decreases to $v_{\rm abs} \sim 11,000$ km/s at the maximum date $p \sim 0$ with range ~2000 km/s for different colors, and it is nearly a constant during these phases. Finally, it will continue to decrease to $v_{\rm abs} \sim 9,000$ km/s at $p \sim 25$ with range 1,000 km/s. The speed is almost a constant during $0 < p < 5$, i.e., $\partial_t v_{\rm abs} \sim 0$, while $\partial_t v_{\rm abs} \sim 110\,{\rm km/s/day}$ after phase $p \sim 10$.

VII. CONCLUSION AND FUTURE WORK

We have constructed an SN Ia spectrum evolution model by training a backpropagation neural network with the observed nearby spectra from the CfA SN Program. The model can well describe not only the spectrum of SNe Ia with wavelength range from 3500 to 8000 Å but also the light-curve evolution with phase time from −15 to 50 for different colors. We also compare the results with the SALT2 template and find that the model we constructed by using ANN is almost the same as the SALT2 template but with fewer parameters during the training process.

From Fig. 5, one can see that the RMSE and MAE appear to still be decreasing with increasing network complexity. Actually, we have tried a few more complicated networks with more layers and hidden units, but the RMSE and MAE do not decrease very much, something like $\lesssim 0.1\%$. However, the computing time increases very much. It costs even a few months to train a network with the most complicated network during our research. Besides, the number of parameters will increase when the network becomes complicated, and it may lead to an overfit issue.

By taking advantage of the network method, we calculated Si II λ6355 absorption velocity and its gradient in different phases and colors, and our results are consistent with the previous works in the literature.

It also shows that the Levenberg-Marquardt algorithm for training the network is much faster than others such as the RPROP algorithm.

With the artificial neural network, the evolution of the universe could be obtained with the trained spectra model, which we will tackle in our future work. Actually, the history of the universe could be observed without assumption of any cosmological model, for example, the model given in Refs. [25,26]. Such a model-independent attempt has been put forward in Ref. [27]. We will try to achieve this goal using the network method directly.

ACKNOWLEDGMENTS

This work is supported by National Science Foundation of China Grants No. 11105091, No. 10671128, and No. 11047138, the Key Project of Chinese Ministry of Education Grant No. 211059, "Chen Guang" project supported by Shanghai Municipal Education Commission and Shanghai Education Development Foundation Grant No. 12CG51, and Shanghai Natural Science Foundation, China, Grant No. 10ZR1422000. This research has made use of the CfA Supernova Archive, which is funded in part by the National Science Foundation through Grant No. AST 0907903.

[1] A. G. Riess *et al.* (Supernova Search Team), Astron. J. **116**, 1009 (1998).
[2] S. Perlmutter *et al.* (Supernova Cosmology Project Collaboration), Astrophys. J. **517**, 565 (1999).
[3] A. V. Bogomazov and A. I. Tutukov, Astronomy Reports **53**, 214 (2009).
[4] M. M. Phillips, Astrophys. J. **413**, L105 (1993).
[5] J. Guy *et al.* (SNLS Collaboration), Astron. Astrophys. **466**, 11 (2007).
[6] W. S. McCulloch and W. Pitts, Bull. Math. Biophys. **5**, 115 (1943).
[7] M. Minsky and S. Papert, *Perceptrons* (MIT Press, Cambridge, MA, 1969).
[8] X. Wang *et al.*, Astrophys. J. **699**, L139 (2009).
[9] S. Blondin *et al.*, Astron. J. **143**, 126 (2012).
[10] S. Benetti *et al.*, Astrophys. J. **623**, 1011 (2005).
[11] D. Branch *et al.*, Publ. Astron. Soc. Pac. **118**, 560 (2006).
[12] S. Blondin *et al.*, Astron. J. **131**, 1648 (2006).

[13] P. Graff, F. Feroz, M. P. Hobson, and A. N. Lasenby, Mon. Not. R. Astron. Soc. **441**, 1741 (2014).
[14] M. Riedmiller and H. Braun, in *Proceedings of the IEEE International Conference on Neural Networks* (IEEE Press, New York, 1993), pp. 586–591.
[15] T. Matheson *et al.*, Astron. J. **135**, 1598 (2008).
[16] D. Branch *et al.*, Astron. J. **126**, 1489 (2003).
[17] S. Jha *et al.*, Astrophys. J. Suppl. Ser. **125**, 73 (1999).
[18] K. Krisciunas *et al.*, Astron. J. **142**, 74 (2011).
[19] W. Li *et al.*, Publ. Astron. Soc. Pac. **115**, 453 (2003).
[20] R. J. Foley, G. Narayan, P. J. Challis, A. V. Filippenko, R. P. Kirshner, J. M. Silverman, and T. N. Steele, Astrophys. J. **708**, 1748 (2010).
[21] M. Hicken, P. M. Garnavich, J. L. Prieto, S. Blondin, D. L. DePoy, R. P. Kirshner, and J. Parrent, Astrophys. J. **669**, L17 (2007).
[22] A. Kim, A. Goobar, and S. Perlmutter, Publ. Astron. Soc. Pac. **108**, 190 (1996).
[23] P. Nugent, A. Kim, and S. Perlmutter, Publ. Astron. Soc. Pac. **114**, 803 (2002).
[24] R. J. Foley, N. E. Sanders, and R. P. Kirshner, Astrophys. J. **742**, 89 (2011).
[25] C. J. Feng, X. Y. Shen, P. Li, and X. Z. Li, J. Cosmol. Astropart. Phys. 09 (2012) 023.
[26] C. J. Feng, Phys. Lett. B **672**, 94 (2009).
[27] C. J. Feng and X. Z. Li, Astrophys. J. **821**, 30 (2016).