

Extension of M Dwarf Spectra Based on Adversarial AutoEncoder

Jiyu Wei , Xingzhu Wang, Bo Li , Yuze Chen and Bin Jiang * 

School of Mechanical, Electrical & Information Engineering, Shandong University, Weihai 264209, China; weijiyu@mail.sdu.edu.cn (J.W.); wangxingzhu@mail.sdu.edu.cn (X.W.); libo18@mail.sdu.edu.cn (B.L.); cyz122699@mail.sdu.edu.cn (Y.C.)

* Correspondence: jiangbin@sdu.edu.cn

Abstract: M dwarfs are main sequence stars and they exist in all stages of galaxy evolution. As the living fossils of cosmic evolution, the study of M dwarfs is of great significance to the understanding of stars and the stellar populations of the Milky Way. Previously, M dwarf research was limited due to insufficient spectroscopic spectra. Recently, the data volume of M dwarfs was greatly increased with the launch of large sky survey telescopes such as Sloan Digital Sky Survey and Large Sky Area Multi-Object Fiber Spectroscopy Telescope. However, the spectra of M dwarfs mainly concentrate in the subtypes of M0–M4, and the number of M5–M9 is still relatively limited. With the continuous development of machine learning, the generative model was improved and provides methods to solve the shortage of specified training samples. In this paper, the Adversarial AutoEncoder is proposed and implemented to solve this problem. Adversarial AutoEncoder is a probabilistic AutoEncoder that uses the Generative Adversarial Nets to generate data by matching the posterior of the hidden code vector of the original data extracted by the AutoEncoder with a prior distribution. Matching the posterior to the prior ensures each part of prior space generated results in meaningful data. To verify the quality of the generated spectra data, we performed qualitative and quantitative verification. The experimental results indicate the generation spectra data enhance the measured spectra data and have scientific applicability.

Keywords: M-type dwarfs; adversarial AutoEncoder; spectral data generation; sky survey



check for updates

Citation: Wei, J.; Wang, X.; Li, B.; Chen, Y.; Jiang, B. Extension of M Dwarf Spectra Based on Adversarial AutoEncoder. *Universe* **2021**, *7*, 326. <https://doi.org/10.3390/universe7090326>

Academic Editor: Lorenzo Iorio

Received: 25 June 2021

Accepted: 30 August 2021

Published: 31 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The traditional method to address spectral classification of stars is to combine their photometric and spectroscopic data together. The most commonly used Harvard stellar spectral classification system was proposed by the Harvard University Observatory in the late 19th century [1]. In accordance with the order of the surface temperature of the star, the system divides the stellar spectra into O, B, A, F, G, K, M, and other types [2].

M dwarfs are the most common stars in the Galaxy [3] and are characterized by low brightness, small diameter and mass, and a surface temperature around or lower than 3500 K. With the nuclear fusion speed inside the M dwarfs being slow, M dwarfs tend to have a long life span, and they exist in all stages of the evolution of the Galaxy [4]. A huge number of spectra are obtained with the emergence of sky survey telescopes, such as Sloan Digital Sky Survey (SDSS) [5,6] and Large Sky Area Multi-Object Fiber Spectroscopy Telescope (LAMOST) [7,8].

However, the distribution of the subtypes is unbalanced. In the SDSS-DR15, as shown in Figure 1, the spectra of M0–M4 are relatively greater in number, whereas that of M5–M9 is limited. The generation of specific subtype spectra of M dwarfs is helpful to solve the problem of unbalanced distribution and provide more reliable samples for research. For example, in the SDSS dataset, the number of M5–M9 is very low. When the data are limited, it is difficult for astronomers to analyze them using machine learning or deep learning methods, such as classification, clustering, dimensionality reduction, etc. If we can effectively expand the data, we can improve the M Dwarf dataset to better understand

these stars. In this study, we select M-class stars with unbalanced distribution of subtypes M0–M6 (signal-to-noise ratio > 5) to verify the effectiveness of our method.

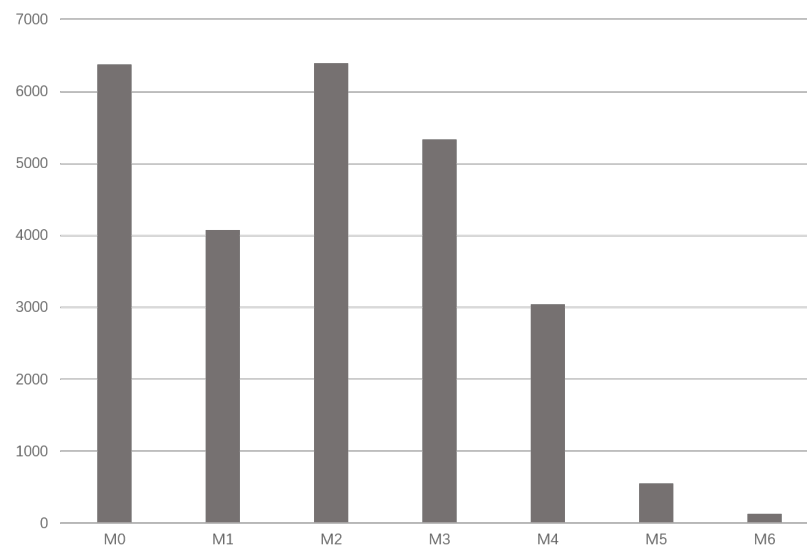


Figure 1. Distribution of M0–M6 dwarfs with signal-to-noise ratio > 5 of Sloan Digital Sky Survey (SDSS) 15.

With the development of machine learning and deep learning, generation models were remarkably improved. An increasing number of methods are proposed to solve the lack of data, and all kinds of data, especially the two-dimensional data from the real world, were expanded effectively. AutoEncoder is a kind of neural network. After encoding and decoding, it can obtain output similar to the input. The Variational AutoEncoder (VAE), a model proposed by Kingma and Welling [9], combines variational Bayesians with neural network and achieves good results with data generation. Generative Adversarial Nets (GAN) is a model proposed by Goodfellow et al. [10] to solve the lack of data, especially for the 2D data from the real world [10–14]. GAN consists of a discriminator and a generator. The discriminator is designed to determine whether the input data are real or fake data generated by the generator, and the task of the generator is to generate fake data that can confuse the discriminator as much as possible. Through such a dynamic game process, similar data are generated.

The Adversarial AutoEncoder (AAE) is a model proposed by Makhzani et al. [15]. The AAE replaces the generator of the traditional generation model, GAN, with an AutoEncoder that can better learn the feature of discrete data. At the same time, the discriminator is used to correct the distribution after encoding. By doing this, the problem of traditional GAN with the generation of discrete data is solved effectively. However, due to the restriction of traditional GAN structure, the AAE also has problems, such as unstable training and model collapse, and training a good AAE with a small amount of data is difficult.

For the quality of the generated spectrum, it is necessary to qualitatively test its similarity with the original spectrum. Principal Component Analysis (PCA) [16] is a widely used data dimensionality reduction algorithm that can extract features of high-dimensional data. T-Distributed Stochastic Neighbor Embedding (t-SNE) is a visualization method for high-dimensional data proposed by Arjovsky et al. [17]. These two methods can visually demonstrate the similarity between the observational spectra and the generated spectra.

Simultaneously, we use the generated spectrum to enhance the data of the classifier to further quantitatively verify the value of the generated spectrum. Fully connected neural network is a commonly used feature extraction method; through multilayer full connection, the feature of the spectrum can be effectively extracted. Training the classifier through two methods can visually show the performance improvement of the classifier

after data expansion using the generated spectrum. The contribution of this paper could be summarized as three-fold:

1. We used AAE to generate spectral data, and the model performed well with various kinds of spectral data, providing new ideas for the generation of spectral data.
2. From a qualitative and quantitative perspective, we proved the high quality of the generated spectra and the effectiveness and robustness of the AAE.
3. Our work provides a new direction for the combination of astronomy and machine learning.

2. Method

In this work, we propose to use an Adversarial AutoEncoder (AAE) to generate spectral data. The model is composed of a generator and a discriminator; the generator is an AutoEncoder composed of an encoder and a decoder, and the discriminator is implemented by a GAN discriminant network. AAE does not directly train the network to generate spectral data. Instead, the output of the encoder in the AutoEncoder is constrained to conform to a preselected prior distribution by the game process between the discriminator and the generator. The network parameters of autoencoder are continuously optimized to make the output of the decoder as consistent as possible with the input of the AutoEncoder. Finally, a decoder is obtained as the generator of spectral data, which can stably decode the vector that conforms to a prior distribution into high-quality spectral data.

In this study, we use two fully connected neural networks to form the encoder and decoder of the AutoEncoder, and the GAN discriminant network is constituted of a two-layer, fully connected neural network. The model is shown in Figure 2. The training process is divided into two stages: the reconstruction stage, which aims to obtain a decoder that can stably reconstruct the encoding vector into high-quality spectral data, and the regularization stage, which aims to constrain the encoding vector generated by the encoder to an artificially selected prior distribution through GAN's confrontation process.

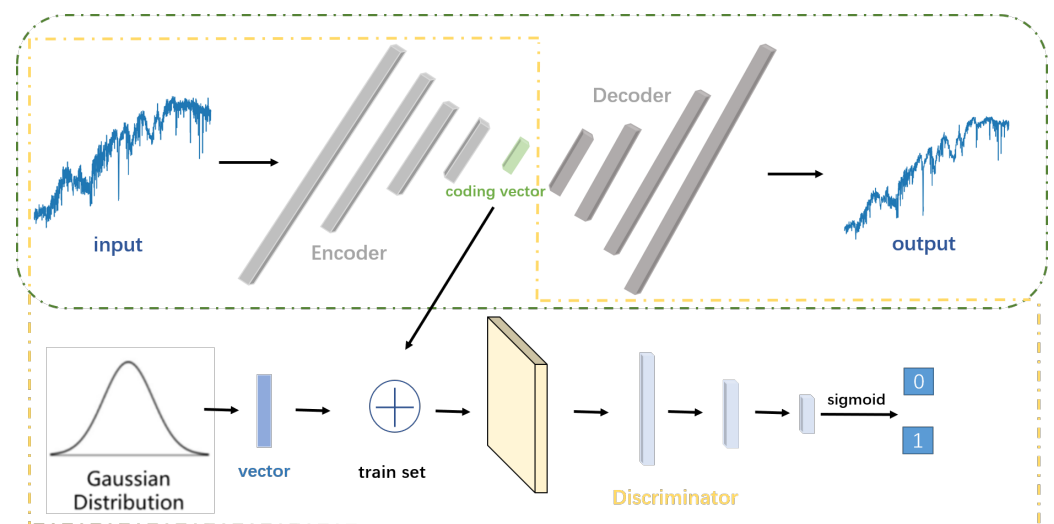


Figure 2. Structure of the Adversarial AutoEncoder (AAE). AAE is composed of an autoencoder and a discriminator. Green box part is autoencoder composed of an encoder and a decoder, and yellow box part is a GAN composed of a discriminator and an autoencoder. Firstly, spectral data are encoded and then decoded to generate reconstructed data similar to the input data. Secondly, encoding vector of input spectral data and a randomly selected vector which conforms to the normal distribution are used as false data and real data as input of the discriminator, respectively.

Formally, we denote the encoder as Q , P as the decoder and D as the generator of the GAN. The input of the spectral data and the output of decoder are denoted as x and \hat{x} . The encoding vectors generated by encoder are denoted as z and are constrained to a prior

distribution $p(z)$; in this paper, $p(z)$ adopts standard normal distribution. The parameter of our model is shown in Table 1:

Table 1. Parameters of AAE.

Encoder			
Layer	Input Shape	Output Shape	Activation Function
Input	(20, 3522)	(20, 1024)	-
Linear	(20, 1024)	(20, 512)	ReLU
BatchNorm1d	(20, 512)	(20, 512)	-
Linear	(20, 512)	(20, 256)	ReLU
BatchNorm1d	(20, 256)	(20, 256)	-
Linear	(20, 256)	(20, 128)	ReLU
BatchNorm1d	(20, 128)	(20, 128)	-
Output	(20, 128)	(20, 32)	-
Decoder			
Layer	Input Shape	Output Shape	Activation Function
Input	(20, 32)	(20, 128)	-
Linear	(20, 128)	(20, 256)	ReLU
BatchNorm1d	(20, 256)	(20, 256)	-
Linear	(20, 256)	(20, 512)	ReLU
BatchNorm1d	(20, 512)	(20, 512)	-
Linear	(20, 512)	(20, 1024)	ReLU
BatchNorm1d	(20, 1024)	(20, 1024)	-
Output	(20, 1024)	(20, 3522)	-
Discriminator			
Layer	Input Shape	Output Shape	Activation Function
Linear	(20, 3522)	(20, 128)	ReLU
Linear	(20, 128)	(20, 1)	Sigmoid

2.1. Reconstruction Stage

In the reconstruction stage, the encoder encodes the discrete spectral data x with 3522 dimensional features into a 32-dimensional code vector z , and then restores it to a 3522-dimensional reconstruction output \hat{x} through a symmetrical decoder. To ensure the data reconstructed by the decoder are similar to the real data, the generated spectrum and the original spectrum use binary cross entropy as the loss function to measure the similarity between original and reconstructed data, as shown in Equation (1), and they also use the gradient descent algorithm to minimize the loss and back-propagation to update the parameters.

$$loss = \frac{1}{n} \sum_{i=1}^n \|x - \hat{x}\|^2 \quad (1)$$

2.2. Regularization Stage

In the regularization stage, we conduct confrontation training similar to GAN, where the decoder of the AutoEncoder is used as the generator and the GAN discriminant network is used as the discriminator.

First, a standard normal distribution code vector with a size of 32 is randomly generated as the positive sample, and the code vector generated by the encoder is used as the negative sample. Then, the discriminator extracts the features of the input encoding vector and uses the Sigmoid activation function in the last layer of the discriminator to normalize the final output within the range of [0, 1]. The output of the discriminator represents the probability of the input being true, which determines whether the input's distribution is closer to the real spectral data coding or the standard normal vector distribution. Therefore, the parameter optimization of the discriminator is carried out according to the idea

of zero-sum game, as shown in Equation (2), which represents the probability that the discriminator successfully recognizes the real data as true and the generated data as fake.

$$\max D = E_{x \sim p(x)}[\log(D(x))] + E_{z \sim p(z)}[\log(1 - D(G(z)))] \quad (2)$$

When training the generator, we improve the generator's ability to confuse the discriminator by optimizing the parameters of the encoder so as to minimize the probability of the discriminator successfully discriminating the generated data. As such, the encoding vector z generated by the encoder conforms to the prior distribution $p(z)$ as much as possible, as shown in Equation (3), which represents the probability that the generated data are recognized as fake by the discriminator.

$$\min G = E_{z \sim p(z)}[\log(1 - D(G(z)))] \quad (3)$$

In the process of model training, the ability of decoder to reconstruct the code vector z into spectrum continuously improves, and the code vector distribution obtained by the encoder gradually approaches the standard normal distribution simultaneously. Through this model, we obtain a decoder as the spectrum generator, which can reconstruct any vector conforming to the standard normal distribution into a high-quality target spectrum. When generating the spectrum, we first randomly generate a set of vector groups conforming to the standard normal distribution as the encoding vector, and then directly use the trained decoder as the generator to generate the spectrum. The detailed optimization procedures are summarized in Algorithm 1.

Algorithm 1 Adversarial AutoEncoder Training Strategy.

Input: Target spectral data X_t ;

Output: Spectrum \hat{X}_t generated by AAE ;

```

1: for  $i \in [1, epochs]$  do
2:   for each mini-batch  $x_i \in X_t$  do
3:     Reconstruction phase:
4:     Encoding  $x_i$  to  $z_i$  by Q;
5:     Decoding  $z_i$  to  $\hat{x}_i$  by P;
6:     compute Reconstruction loss between  $x_i$  and  $\hat{x}_i$  as Equation (1) and update the
       encoder and decode;
7:     Regularization phase:
8:     Randomly choose vectors  $\hat{z}_i$  from a Gaussian distribution as true data
9:     Generate vectors  $z_i$  from  $x_i$  as false data by Generator G (same as Encoder P)
10:    Combine  $z_i$  and  $\hat{z}_i$  as training data Z;
11:    Discriminating Z and compute Regularization loss as Equations (2) and (3), then
       update Discriminator D and Generator G;
12:   end for
13: end for
14: Generate data:
15: Randomly choose vectors  $\hat{z}_i$  from a Gaussian distribution
16: Use Encoder Q as Generator transforms  $\hat{z}_i$  to  $X_t$  as generate data
17: return  $X_t$ 

```

2.3. Visualization of Dimensionality Reduction

PCA and t-SNE convert high-dimensional data to low-dimensional data under the premise of small information loss, which is conducive to visualization. The goal of t-SNE is to map the data distribution of the high-dimensional space into the low-dimensional space and take the difference of the probability distribution of the data in the high-dimensional and low-dimensional space as a constraint condition. We define the KL divergence to represent the difference between the two distributions; the smaller the KL divergence is, the smaller the difference is:

$$C = \sum KL(P_i|Q_i) = \sum_i \sum_j p_{j|i} \log \left(\frac{p_{j|i}}{q_{j|i}} \right) \tag{4}$$

$$p_{j|i} = \frac{\exp \left(-\|x_i - x_j\|^2 / (2\sigma_i^2) \right)}{\sum_{k \neq i} \exp \left(-\|x_i - x_k\|^2 / (2\sigma_i^2) \right)} \tag{5}$$

$$q_{j|i} = \frac{\exp \left(-\|y_i - y_j\|^2 \right)}{\sum_{k \neq i} \exp \left(-\|y_i - y_k\|^2 \right)} \tag{6}$$

where x_i and x_j represent two different spectral data, y_i and y_j correspond to x_i and x_j in low-dimensional space, σ_i represents the variance of the Gaussian distribution constructed with x_i as the center, $p_{j|i}$ represents the probability that x_j is the neighborhood of x_i , and $q_{j|i}$ represents the conditional probability that y_j is the domain of y_i . The gradient descent method is used to solve the optimization problem and find the minimum value of KL divergence. During iterative optimization, the data distribution in the low-dimensional space continuously gets closer to the data distribution in the high-dimensional space. Finally, the resulting low-dimensional space data can be considered the mapping of high-dimensional data in low dimensions.

3. Experiment

3.1. Spectra Acquisition and Preprocessing

A total of 25,892 M-type dwarf spectra from SDSS DR15 were obtained through Casjob¹. The subtype ranges from M0 to M6. The total number of M6 spectra we choose to generate is 127, accounting for 0.49% of the total, as shown in Table 2.

Table 2. Number of the experimental spectra.

Type	5–10 (SNR)	10–15 (SNR)	Up15 (SNR)
M0	2850	1503	2023
M1	1919	1025	1134
M2	3300	1745	1343
M3	3105	1055	1170
M4	1658	779	603
M5	334	137	82
M6	86	21	20

Then, we normalize the spectra using the following steps:

1. Discard all the spectral data with a signal-to-noise ratio < 5.
2. The uniform wavelength range is 3800–9000 Å, and the sampling points for each sample data is 3522.
3. Normalize each sampling point of the sample data, as shown in Equation (7). The range is [0, 1].

$$X_i = \frac{X_i - X_{min}}{X_{max} - X_{min}} \tag{7}$$

After normalization of each spectrum, the spectra is used as the input to the AutoEncoder network.

3.2. Spectra Generation

The spectra generation uses AAE to accept the original spectra as input. In 500 iterations of training, as shown in Figure 3, the loss of the autoencoder first experienced a rapid decrease and then became stable, indicating that the autoencoder part of AAE can successfully output reconstructed data that are similar to that of the input. In addition,

although there are some fluctuations in the loss of the generator and the discriminator, it is basically in a state of balance, indicating that the discriminator and the generator reach equilibrium during the game process, and problems such as gradient disappearance did not occur.

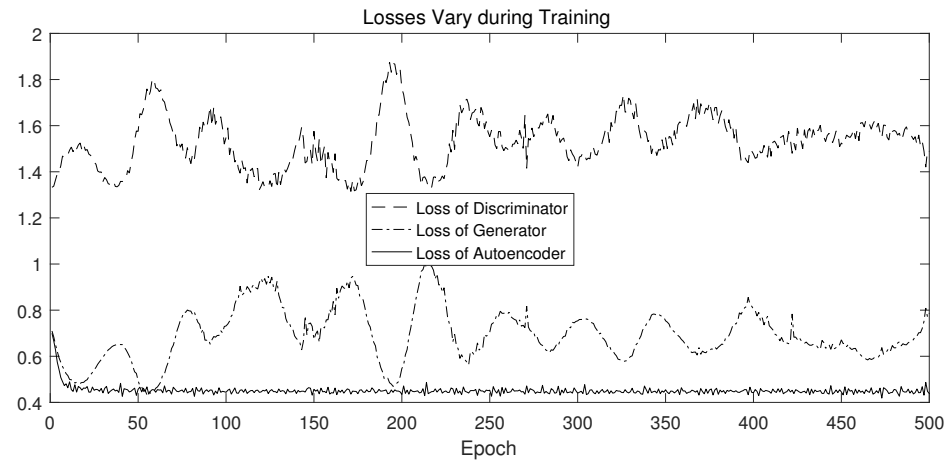


Figure 3. Loss of generator, discriminator, and autoencoder during training: -- represents loss of discriminator fluctuates in range of [1.2, 2.0], -.-.- represents the loss of generator fluctuates in range of [0.4, 1.0], — represents loss of autoencoder fluctuates in range of [0.4, 0.8].

Simultaneously, the AAE gradually and steadily generates higher-quality spectral data, as shown in Figures 4 and 5. With training of the model, the noise of the generated spectral data gradually decreases, the characteristic peaks are more obvious, and the signal-to-noise ratio is significantly improved. The generated spectrum and the measured spectrum are very similar in multiple characteristic peaks, and the generator learned the feature of the original spectral data.

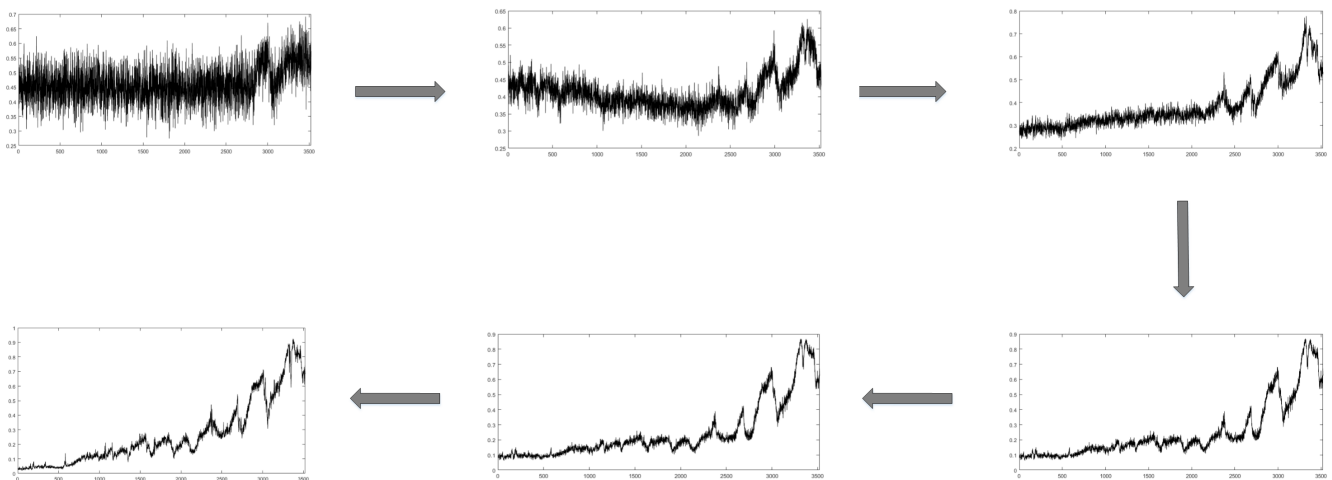


Figure 4. Generated spectral data during training.

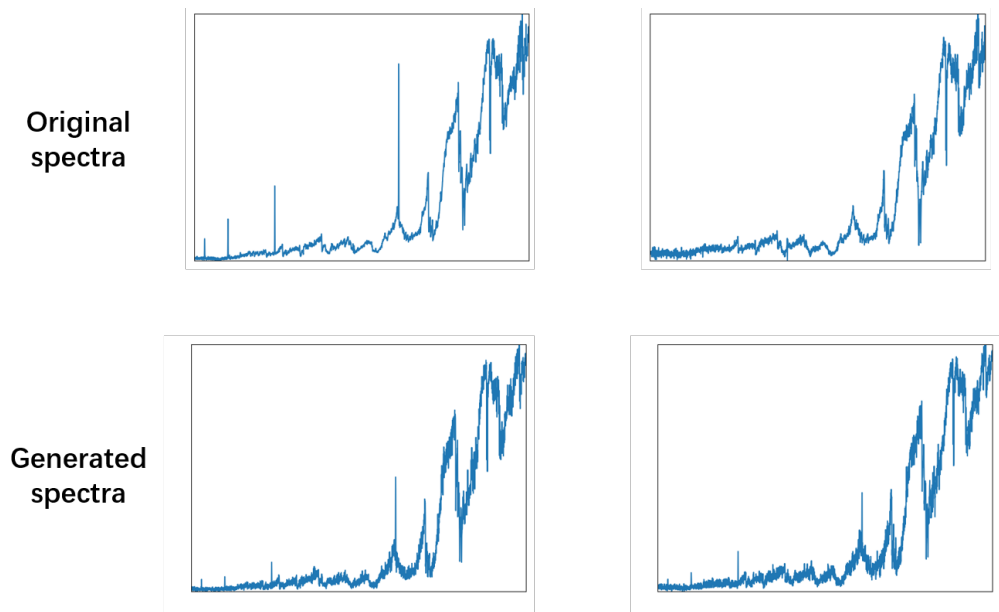


Figure 5. Compare original spectral data with generated spectral data.

3.3. Qualitative Experiment

After training, a model trained against the AutoEncoder was obtained, and 400 M6 dwarf spectra (black) were generated for verification. Simultaneously, 127 measured M6 spectra (red) and 400 data of M0~M5 dwarf were selected.

All the spectral data are reduced to two dimensions and projected by PCA, as shown in Figure 6. The black area basically surrounds the red area and is clearly distinguished from M0–M5 type dwarfs. Furthermore, a few outliers exist due to the influence of the signal-to-noise ratio.

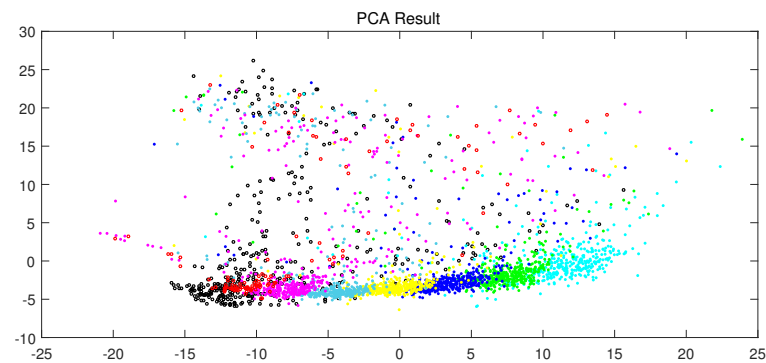


Figure 6. Result of dimension reduction by PCA.

We use t-SNE to visualize the above data by dimensionality reduction, as shown in Figure 7. Most of the black areas also coincide with the red areas. Although a small part of the black area does not coincide with the red area, there are black areas near each red area, which once again proves AAE's ability to generate the simulation data.

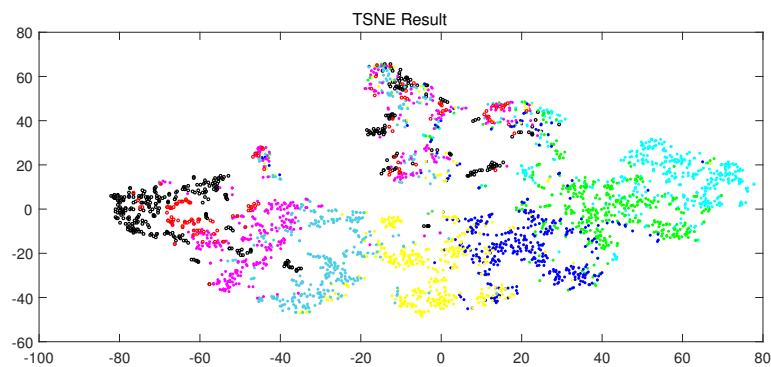


Figure 7. Visualization result of t-SNE dimension reduction.

3.4. Quantitative Experiment

3.4.1. Evaluation Protocol

To better evaluate the results of quantitative experiments, we propose an evaluation protocol consisting of Precision (P), Recall (R), and F1-score (F). Precision quantifies the number of positive class predictions that actually belong to the positive class; Recall quantifies the number of positive class predictions made out of all positive examples in the dataset, and F1-score provides a single score that balances both the concerns of Precision and Recall in one number, as defined in (8).

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN} \quad F = \frac{2TP}{2TP + FP + FN} \quad (8)$$

True Positive (TP) represents the number of correctly predicted positive samples; True Negative (TN) represents the number of correctly predicted negative samples; False Positive (FP) represents the number of incorrectly predicted positive samples; False Negative (FN) represents the number of incorrectly predicted negative samples, as shown below in the Confusion Matrix in Figure 8.

		Actual	
		Positive	Negative
Predicted	Positive	True Positive(TP)	False Positive(FP)
	Negative	False Negative(FN)	True Negative(TN)

Figure 8. Confusion Matrix.

3.4.2. Quantitative Experiment

To further quantitatively evaluate the quality of the spectrum generated by the AAE model, we used fully connected neural networks as spectral classifiers for classification, as shown in Table 3. We also chose all data from the M-type data set of M0–M6 with a signal-to-noise ratio of >5 as train data (0.75) and test data (0.25).

Table 3. Parameters of Classifier.

Classifier			
Layer	Input Shape	Output Shape	Activation Function
Input	(1, 3522)	(20, 1000)	ReLU
Linear	(1, 1000)	(1, 200)	ReLU
Linear	(1, 200)	(1, 50)	ReLU
Output	(1, 50)	(1, 7)	Softmax

Firstly, we use both the train and test data to train a classifier to verify the accuracy of generated spectral data so as to verify whether generated data could represent the desired class, as shown in Table 4. Then, we use the AAE model for data enhancement to complete all the train spectral data to 3000 to verify the enhancement of generated data for the real spectral data, as shown in Table 5.

Table 4. Accuracy of M0–M6 data generated by AAE.

Data Type	AAE		
	P	R	F
m0	1.000	1.000	1.000
m1	99.95	99.92	99.94
m2	99.91	99.95	99.93
m3	99.99	99.85	99.92
m4	97.63	99.85	98.73
m5	74.65	95.46	83.78
m6	96.73	67.69	79.64
macro avg	95.55	94.67	94.56
weighted avg	95.55	94.67	94.56

Table 5. Classification of M0–M6 subdwarfs with data augmentation.

Data Type	Ori			AAE		
	P	R	F	P	R	F
m0	97.91	98.23	98.07	98.07	98.01	98.04
m1	95.55	94.89	95.22	95.34	95.12	95.23
m2	96.64	96.16	96.40	96.85	96.16	96.50
m3	95.41	95.91	95.66	95.35	96.04	95.70
m4	95.06	94.81	94.94	94.80	94.55	94.68
m5	84.42	92.86	88.44	85.91	91.43	88.58
m6	87.50	65.62	75.00	77.42	75.00	76.19
macro avg	93.21	91.21	91.96	91.96	92.33	92.13
weighted avg	96.07	96.06	96.05	96.07	96.06	96.06

The quantitative experimental results of the AAE model can be analyzed. The AAE model can generate target spectral data stably and with high quality, representing the target class exactly. Data enhancement of the spectral data can not only improve the classification effect of the model, but also remit the problem of imbalanced data distribution. In the future, we will further explore the data-influencing factors on the spectral quality of the AAE generation model.

4. Discussion

In this section, we will discuss the astronomical application and the potential for further study of AAE. In case of a high cost of actual observation, AAE can generate simulation spectra to improve the M Dwarf dataset to better understand these stars in a simple way. It provides a reference method for the automatic searching of other rare and special celestial objectives in the massive spectra. This method can simulate any type of

spectrum and is an innovative research method, and we will extend AAE to other types of spectra for further studies. Moreover, AAE is an extension of GAN, so it can refer to other improvements related to GAN to improve the performance of AAE in spectral data generation, such as Wasserstein GAN [18], Cycle GAN [19], and so on.

5. Conclusions

In this paper, we proposed using AAE to generate high-quality spectral data. To verify the effectiveness of AAE, we designed quantitative and qualitative experiments on M-type stars. The experimental results showed that the generated spectra can expand the original spectra, and the expanded range does not exceed one subtype, which has strong practical value and significance.

Author Contributions: Conceptualization, B.J. and J.W.; methodology, J.W.; software, J.W. and X.W.; validation, X.W. and B.L.; formal analysis, Y.C. and B.L.; investigation, X.W.; resources, B.J.; writing—original draft preparation, J.W.; writing—review and editing, J.W.; visualization, B.L. and Y.C.; supervision, B.J.; project administration, B.J.; funding acquisition, B.J. All authors have read and agreed to the published version of the manuscript.

Funding: This paper was supported by the Shandong Provincial Natural Science Foundation, China (ZR2020MA064).

Data Availability Statement: Not applicable.

Acknowledgments: This paper was supported by the Shandong Provincial Natural Science Foundation (ZR2020MA064). Funding for the Sloan Digital Sky Survey IV was provided by the Alfred P. Sloan Foundation, the U.S. Department of Energy Office of Science, and the Participating Institutions. SDSS acknowledges support and resources from the Center for High-Performance Computing at the University of Utah. The SDSS web site is www.sdss.org. We acknowledge the use of spectra from SDSS.

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

Note

¹ <https://dr15.sdss.org/>, accessed on 13 March 2021

References

1. Welther, B.L. Discoveries, Achievements, and Personalities of the Women Who Evolved the Harvard Classification of Stellar Spectra: Williamina Fleming, Antonia Maury, and Annie Jump Cannon. Available online: <https://ui.adsabs.harvard.edu/abs/2010AAS...21520004W/abstract> (accessed on 31 August 2021)
2. Gray, R.O.; Corbally, C.J. *Stellar Spectral Classification*; Princeton University Press: Princeton, NJ, USA, 2009.
3. Fischer, D.A.; Marcy, G.W. Multiplicity among M Dwarfs. *Astrophys. J.* **1992**, *396*, 178. [[CrossRef](#)]
4. Bessell, M.S. The Late M Dwarfs. *Astron. J.* **1991**, *101*, 662. [[CrossRef](#)]
5. Yanny, B.; Rockosi, C.; Newberg, H.J.; Knapp, G.R.; Adelman-McCarthy, J.K.; Alcorn, B.; Allam, S.; Allende Prieto, C.; An, D.; Anderson, K.S.J.; et al. SEGUE: A Spectroscopic Survey of 240,000 Stars with $g = 14\text{--}20$. *Astron. J.* **2009**, *137*, 4377–4399. [[CrossRef](#)]
6. Blanton, M.R.; Bershady, M.A.; Abolfathi, B.; Albareti, F.D.; Allende Prieto, C.; Almeida, A.; Alonso-García, J.; Anders, F.; Anderson, S.F.; Andrews, B.; et al. Sloan Digital Sky Survey IV: Mapping the Milky Way, Nearby Galaxies, and the Distant Universe. *Astron. J.* **2017**, *154*, 28. [[CrossRef](#)]
7. Cui, X.Q.; Zhao, Y.H.; Chu, Y.Q.; Li, G.P.; Li, Q.; Zhang, L.P.; Su, H.J.; Yao, Z.Q.; Wang, Y.N.; Xing, X.Z.; et al. The large sky area multi-object fiber spectroscopic telescope (LAMOST). *Res. Astron. Astrophys.* **2012**, *12*, 1197. [[CrossRef](#)]
8. Zhao, G.; Zhao, Y.H.; Chu, Y.Q.; Jing, Y.P.; Deng, L.C. LAMOST spectral survey—An overview. *Res. Astron. Astrophys.* **2012**, *12*, 723–734. [[CrossRef](#)]
9. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes. *arXiv* **2014**, arXiv:1312.6114.
10. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *arXiv* **2014**, arXiv:1406.2661.
11. Radford, A.; Metz, L.; Chintala, S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv* **2015**, arXiv:1511.06434.
12. Lucic, M.; Kurach, K.; Michalski, M.; Gelly, S.; Bousquet, O. Are gans created equal? A large-scale study. *arXiv* **2017**, arXiv:1711.10337.

13. Ledig, C.; Theis, L.; Huszar, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. Available online: https://openaccess.thecvf.com/content_cvpr_2017/papers/Ledig_Photo-Realistic_Single_Image_CVPR_2017_paper.pdf (accessed on 31 August 2021)
14. Gui, J.; Sun, Z.; Wen, Y.; Tao, D.; Ye, J. A review on generative adversarial networks: Algorithms, theory, and applications. *arXiv* **2020**, arXiv:2001.06937.
15. Makhzani, A.; Shlens, J.; Jaitly, N.; Goodfellow, I.; Frey, B. Adversarial autoencoders. *arXiv* **2015**, arXiv:1511.05644.
16. Ke, N.Y.; Sukthankar, R. PCA-SIFT: A more distinctive representation for local image descriptors. In Proceedings of the IEEE Computer Society Conference on Computer Vision & Pattern Recognition, Washington, DC, USA, 27 June–2 July 2004.
17. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
18. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein generative adversarial networks. In Proceedings of the International Conference on Machine Learning (PMLR), Sydney, Australia, 6–11 August 2017; pp. 214–223.
19. Almahairi, A.; Rajeswar, S.; Sordoni, A.; Bachman, P.; Courville, A. Augmented CycleGAN: Learning Many-to-Many Mappings from Unpaired Data. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018.