Astronomy & Astrophysics manuscript no. output
January 10, 2020

Classifying SDSS spectra using Neural Networks

Soumya Shreeram⋆

École Polytechnique Fédérale Lausanne, Route Cantonale, 1015 Lausanne
e-mail: soumya.shreeram@epfl.ch

January 10, 2020

ABSTRACT

Context. Gravitationally lensed galaxies hold abundant knowledge in understanding the expansion of the universe and studying galaxy properties. The Sloan Digital Sky Survey (SDSS) has measured over a million spectra among which there is a sample characterizing gravitationally lensed spectra. This project aims to develop techniques using Neural Networks (NN) to make it possible to detect such lensed spectra.
Aims. A fully connect Feed-forward Neural Network (FNN) and Convolutional Neural Network (CNN) is implemented to classify spectra from SDSS III and IV, into galaxies, quasars and other astrophysical objects. The spectra that are indecisively classified into multiple categories by the NNs are speculated to be lensed spectra, because of the presence of double spectra along the same line of sight. Furthermore, three diagnostic tools are proposed to assist in the identification of such lensed spectra.
Methods. The project selects the desired spectra suitable for training the NN models and tunes the model parameters by a systematic procedure to produce a trained NN that maximizes the accuracy in classifying objects. The NN application programming interface used in this project is the Keras package that is written in Python.
Results. The accuracy achieved by the FNN was 94.5% with a loss error of 19.0%. CNN achieved a higher accuracy of 97.5% with a lower loss error of 9.5%. Thus, CNN outperformed the FNN. The region that is most probable for detection of lensed spectra is identified by plotting the confidence value distribution for the categorical classification. This method is not without shortcomings and further diagnostic tools that will be useful in the investigation of lensed spectra will be explored in more detail in the future.

Key words. Deep learning - Gravitationally lensed SDSS spectra - FNN and CNN

1. Introduction

The advent of solid-state imaging using charged-coupled devices (CCD), high computational power, and efficient instruments, stipulated the possibility of a wide-area digital sky survey (Gunn et al. 2006; Theuwissen 2006). The Sloan Digital Sky Survey (SDSS) uses a 2.5m wide telescope located at the Apache Point Observatory in New Mexico, United States. This telescope is used to conduct high precision spectroscopy and imaging of the night sky. It has covered 1/3 of the entire sky over the past 19 years of operation. The operation of the telescope is divided into four phases:

- **SDSS I** (2000-2005) commenced operations with the *Legacy* survey that imaged about $11,600$ deg^2 of the celestial sphere and measured over 1.8 million spectra of objects that were chosen from the imaging database. These spectra corresponded to a limited sample of galaxies, called Luminous Red Galaxies (LRGs), and quasars (York et al. 2000).

- **SDSS II** (2005-2008) carried out 2 additional surveys following SDSS I; the *Supernova* survey and the Sloan Extension for Galactic Understanding and Exploration (*SEGUE*) survey (Yanny et al. 2009). These two surveys aimed to study intermediate redshift supernovae to apply constraints on the nature of dark energy (Frieman et al. 2007) and to perform a detailed study the Milky Way's galaxy structure and formation. Overall, the primary goals of SDSS I and II were to create a well-calibrated map and a spectroscopic survey or the northern-Galactic cap.

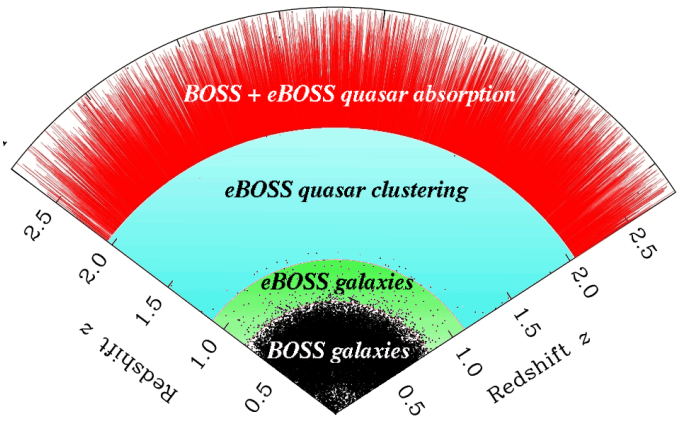

Fig. 1. The spectra of quasars and galaxies that are collected by the BOSS and eBOSS surveys are highlighted as a function of the redshift (eBOSS sdss.org 2018).

- **SDSS III** (2008-2014) further utilized the spectroscopic capabilities of the SDSS telescope by conducting four interconnecting surveys: SEGUE-2, Baryon Oscillation Spectroscopic Survey (*BOSS*), MARVEL and APOGEE (Ahn et al. 2014). Amongst all these surveys, BOSS is of particular relevance to this project (Dawson et al. 2012). The BOSS survey imaged 1.5 million LRGs to a redshift,

⋆ Supervised by: Anand S. Raichoor and Schäfer Christoph E. Rerné.

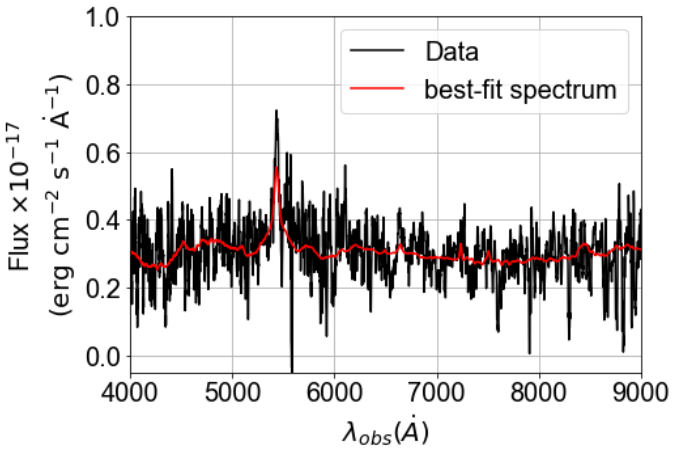

Fig. 2. A typical spectrum of a quasar at $z \sim 0.94$ with the best-fitting model, shown in red. The model has a reduced chi-squared, $\chi_{\rm red} = 0.84$.

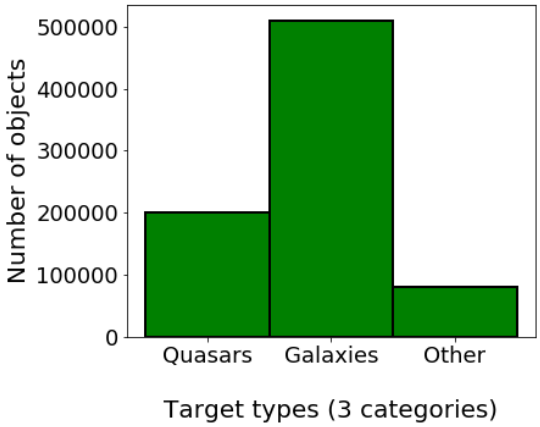

Fig. 3. Classification of 7.84×10^5 spectra into three categories. The data shows the dominance of galaxy-type spectra compared to spectra from quasars or other astrophysical objects. The category *other* contains spectra of astrophysical objects, mostly stars, that are unimportant for the purposes of this project.

$z \sim 0.7$, and collected 160,000 Lyα quasar spectra with $2.2 < z < 3.0$. This survey aimed to measure the correlation function of galaxies and the quasar Lyα forest to detect baryon oscillations (Aihara et al. 2011).

- **SDSS IV** (2014-2020) continues the legacy of SDSS III by performing three distinct surveys: eBOSS, APOGEE-2, and MaNGA (Dawson et al. 2016). For this project, eBOSS is of particular relevance and hence, it is briefly introduced here. eBOSS is the Extended Baryon Oscillation Spectroscopic Survey that is the successor of BOSS; it measures spectra from galaxies, called as Emission Line Galaxies (ELGs) along with LRGs in the redshift range $0.6 < z < 1.1$. Additionally, it also observes quasars at $0.9 < z < 2.1$ and Lyα quasars at $2.1 < z < 3.5$ (Blanton et al. 2017). Fig. 1 summarises the redshift ranges for the galaxy and quasar measurements by BOSS and eBOSS surveys.

The spectroscopic mode for the BOSS and eBOSS are enabled by aluminium plug plates that each observe a 3 deg^2 patch of the night sky. Each unique plate is plugged with 1000 optical fibres, where each fibre is positioned on a particular galaxy, quasar or star and carries this light to the spectrograph for analysis. Thus, this enables observations of multiple, distinct objects that results in the generation of 1000 spectra simultaneously (York et al. 2000). Fig. 2 shows an example of a quasar spectrum collected by eBOSS plate number 9003 with fibre ID 120.

Gravitationally lensed galaxies allow us to study the mass distribution of matter in the lensing galaxies, galaxy properties, and the rate of expansion of the universe (Morningstar et al. 2018). A remarkable feature of spectroscopic measurements is that it opened the pathway to detect strong gravitationally lensed candidates along a single line of sight. It must be noted that lensed spectra are accidental measurements, however, in large surveys there exists a finite probability to measure such spectra. Bolton et al. (2008) were the pioneers to develop a systematic strategy for strong lens spectroscopic detection by identifying the presence of characteristic emission lines in the background spectrum. This method not only enabled them to obtain precise spectral measurements of the lens itself but also the redshift information of the lens-source candidate was acquired. Furthermore, since the exact location of these lens candidates were known, confirmations were accomplished by photometric tests using the Hubble-Space Telescope (HST) imaging or other high-resolution facilities (Brownstein et al. 2011). However, in this method, succeeding an automated search through 1×10^5 spectra, one had to manually inspect a sub-sample of $\sim$ 2,000 potential spectra to obtain the an accurate sample of lensed spectra. Additionally, previous searches were restricted to spectra that necessarily contained a galaxy. Although, this is a reasonable assumption to begin with, NNs provide the freedom to hunt for other potential lensing candidates. With the inevitable dawn of the age of big data where SDSS has measured over a million spectra and upcoming experiments like DESI (Dey et al. 2019) aim to collect even more data, a more efficient method for strong-lens detection needs to be developed. The project aims to implement deep learning algorithms to detect such strong-lens candidates.

In this paper, two deep learning algorithms are implemented: the fully connected Feed-forward Neural Network (FNN) and the Convolutional Neural Network (CNN). The accuracy of these algorithms is tested by their ability to classify galaxies, quasars, and other astrophysical objects from 0.8 million BOSS and eBOSS spectra. Sec. 2 introduces the essential concepts in deep learning and describes the classification models that are built using the two algorithms. Sec. 3 describes the preliminary analysis performed on the data to produce training and testing data sets for FNN and CNN and Sec. 4 discusses the main results of the project. Finally, Sec. 5 explains how this work will be carried forward to detect lensed spectra.

2. Implementation of Deep learning algorithms

Gravitationally lensed spectra are identified as a composition of *double spectra* along a single line of sight. In this project, a supervised approach with a three-category classification is implemented to detect such lensed spectra. Fig. 3 demonstrates the three categories into which the original spectra are classified. However, noise and other emissions from astrophysical objects that are not identified by the deep learning algorithm pose a potential challenge for obtaining high accuracy in classification. We proceed to introduce some essential terminologies in Sec. 2.1 followed by the description of the two main neural networks trained in the project: the fully connected Feed-forward Neural

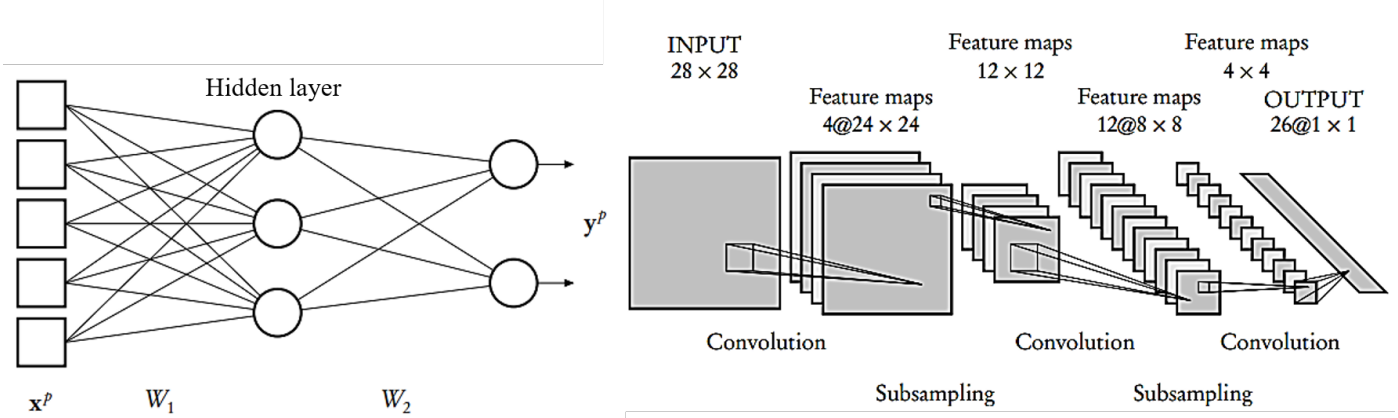

Fig. 4. *Left:* A simple example of a three-layer FNN containing one input layer, one hidden layer, and one output layer. *Right*: An example of a CNN that inputs a 24×24 pixel image. The NN contains 1 input layer, two hidden layers, each containing a convolutional layer and a sub-sampling layer, and a fully connected output layer. The two figures are adapted from Bouchain (2006).

Network (FNN) and Convolutional Neural Network (CNN), in Sec. 2.2 and 2.3 respectively.

2.1. Terminologies in Deep Learning

A neural network (NN) consists of multiple, connected neurons that each are activated to produce an output. The activation of the neuron is due to the input data or due to a previously activated neuron that is connected to it. Every neuron output is a scalar u that depends on the input vector $\mathbf{x}$, weight vector $\mathbf{w}$, and bias θ:

$$u = \sum_{k=1}^{n} \mathbf{x}_k \mathbf{w}_k - \theta_k \tag{1}$$

where the summation index k runs over all the n input vectors (Bouchain 2006). The output from every neuron is fed into an *activation function*, $f(u)$, such that the desired non-linearity in the system can be produced. Additionally, this output is a continuous number between 0 and 1. Some examples of activation functions are sigmoid, softmax, tanh, Rectified Linear Unit (ReLU), etc. Amongst these, ReLu is the most widely used activation function (Lau & Lim 2017). The *learning* process constitutes the goal of finding the weights $\mathbf{w}$ for every layer of the network such that the NN displays the desired output behaviour. The optimal $\mathbf{w}$ vector for every layer is calculated through a process called *back-propagation* and is subsequently adjusted by the *loss function*. A particular class of learning, called *Supervised Learning* (SL) is associated with adjusting $\mathbf{w}$ by comparing the output results with the previously given *test* labels. Thus for SL, we would need to provide the NN with a *training* and *testing* data sets (Schmidhuber 2015, p. 4).

2.2. Fully Connected Feed-forward Neural Network (FNN)

FNN is among the most popularly used NN called the multi-layer Feed-forward neural network in which the neurons are arranged in multiple ordered layers to form a unidirectional network. Fig. 4 (*left*) is a schematic of a simple three-layer FNN. Each neuron in a layer is connected to every neuron from the previous layer and every such connection has a w and θ associated with it. The learning process is executed by minimizing the cost function E that is expressed as,

$$E = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{2}$$

where i is the summation index that iterates over all N output neurons, y_i and $\hat{y}_i$ are the desired and computed outputs of the neurons respectively (Svozil et al. 1997, p. 45). The procedure for minimization of E is accomplished by calculating the gradient of the multi-variable parameter space. After every run through the entire data set, so-called as an *epoch*, the weights and biases of the entire network are updated. This part of the training process is called *back-propagation*. This process is repeated for multiple epochs until the highest accuracy and minimum loss is achieved when the model is applied to the *test* data set.

The aim is to generate a model that can be generalized to other data sets. Generalization means that the model performs well on new and unseen data. A measure to check if the model generalizes well in SL is by applying the model on the test data set. An over-trained or *overfitted* model will memorize the training data set and will not generalize well for the test data set (Svozil et al. 1997). Overfitting is a common problem that results in a model that can not be generalized and this can be avoided by optimizing model parameters, as will be discussed in Sec. 3.

In this project, the application programming interface used to generate neural networks is *Keras* (Chollet et al. 2015). The FNN model built contains 3 layers: input layer containing 90 neurons, a hidden layer containing 30 neurons, and an output layer containing 3 neurons. The activation function used for the input and hidden layer is 'ReLu' while for the output layer is 'softmax'. The gradient calculation and optimization of the cost function are executed by the Stochastic Gradient Descent optimizer (SGD) and the loss function used is categorical cross-entropy (Chollet et al. 2015).

2.3. Convolutional Neural Network (CNN)

CNN's are built by convolutional layers where the output of the neuron is a fundamental unit called a *feature map*. Feature maps are results of the convolution of the input image with *filter matrices* or *filters*. As shown in Fig. 4 (*right*), each 2D convolutional layer contains multiple feature maps that are produced by convolutions of the input image with the smaller filters that are projected on the previous layer. CNN outperforms the simple case of FNN because they are built on three novel ideas: invariance under distortions and space translations in the input data sample, shared weights, and local receptive fields. Locally receptive fields mean that a unit in every layer receives

Parameters	Definitions
batch-size	The sample size over which the optimization of the the training set is performed over an iteration
epochs	Number of times the training set is passed through the entire NN
learning rate	Size of step to reach the local minimum
momentum	It helps to dampen the oscillations by aiding the optimizer to find a minimum in the right direction

Table 1. Summary of definitions of the optimization and training parameters that are utilized by FNN and CNN (Ruder 2016; Goodfellow et al. 2016).

the output from a set of units in a small neighbourhood of the previous layer (Bouchain 2006).

The input layer in the case of Fig. 4 is an image of 28×28 pixels. The subsequent layers extract a particular set of features from the input image and eventually recombine these features in the higher layers; this constitutes the learning process. Each feature map shares the same $\mathbf{w}$ vector and the outputs of all the feature maps in a layer can be generated parallelly. Overall, a hidden layer in the NN is composed of a convolutional layer and a *sub-sampling layer*, where the sub-sampling layer reduces the resolution of the feature maps and consequently increases the number of feature maps. The reason why this is made possible is that the important features of the input image are already captured by the CNN architecture due to the property of invariance under space translations which makes the location of the feature in the original image itself unimportant (LeCun et al. 1995).

The model build in this project contains two 1D convolution layers that each contain 16 and 32 nodes respectively and an average sub-sampling size per spectra was applied for every 5 wavelengths. The filter matrix is of size 3 and the activation function used for the two hidden layers is *ReLu*. Finally, the NN is fed into two fully connected layers, containing 64 and 3 neurons respectively, for outputting the final results. The model is compiled by using the SGD optimizer and categorical cross-entropy loss function (Chollet et al. 2015). Further details and the python code for the compiled and trained models: FNN and CNN, can be found on GitHub repository[1].

3. Preliminary Analysis

This section aims to give an overview of the methodology by which the data was prepared before being fed into the FNN and CNN. Sec. 3.1 explains the selection criteria imposed on the spectral data and Sec. 3.2 describes the methodology implemented for generating data such that it could be interpreted by NNs. The corresponding code can be found in Notebooks 02 and 03 of the GitHub repository.

[1] Link to notebook number 04 and 05 are available to view at the GitHub repository: `github.com/SoumyaShreeram/Analyzing_spectra_with_ML/`

3.1. Data Preparation

As mentioned briefly in Sec. 1, the eBOSS and BOSS surveys contain over three million observed spectra at a range of different redshifts. Since the interest of this project is to build a classification model that can distinguish between galaxies, quasars and other astrophysical objects, we would need a spectral data set that mainly includes these objects. Additionally, to train the model to high accuracy, the data must be coherent and the number of noisy spectra in the data set must be minimized. Several selection cuts were applied to choose spectra and plates that met the desired criteria and they are summarized below.

- **Plate Selection**: Of the thousands of plates available in the BOSS and eBOSS catalogue, a sample containing 1200 plates were chosen. This provided a sufficiently large sample of spectra for training the NN models. Among the 1200 plates: 300 eBOSS plates contained Emission-Line Galaxies (ELGs), 500 eBOSS plates contained Luminous Red Galaxies (LRGs) and Quasi-Stellar Objects (QSOs), and 400 BOSS plates contained LRGs and Lyα QSOs.

- **Wavelength adjustments**: The spectrographs used for BOSS and eBOSS (Smee et al. 2013) have an effective wavelength, λ, coverage of approximately $3,600 - 10,400$ Å (Blanton et al. 2017, p.29). Although all the spectra have the same wavelength coverage, there is a slight offset at the start of wavelengths for every spectrum because of the slight variation in the pixel sizes of the spectrograph. So the wavelengths for all the spectra are chosen from the point $\lambda_{\rm min} = 3,700$Å to $\lambda_{\rm max} = 10,000$Å, such that all the spectra had coinciding start and end values. The indices of these start points are also recorded to know the corresponding fluxes.

- **Removing noisy spectra**: This measure was to eliminate any sky and star spectra that were present in the selected plates. The sky spectra are taken for sky subtraction but they are not of importance for the spectral data set that is required for this project. Additionally, the spectra are fitted with a range of templates that are used to determine the redshift alongside some other fit parameters. Only the spectra that resulted in a secure fit i.e. with a reduced $\chi^2 > 0.4$ were selected. Finally, the reliable fibres that have no known issues were chosen by the imposing the command `ZWARN=0` (Dawson et al. 2016).

3.2. Generation of Training and Testing Data Sets

The following sub-sections present the loss and accuracy curves for FNN and CNN, describe the motivation for the given choice of model parameters, the confidence value distribution for the 3-category classification is displayed, and diagnostic methods for further investigation of double spectra are proposed.

The total data set contained 7.8×10^5 spectra and it was divided into two disjoint subsets such that 5.5×10^5 spectra were used for training and 2.4×10^5 spectra for testing the NN models. For every spectrum, the target type, i.e., a galaxy, quasar or other, was recorded by the y-array. The x-array stored the flux values for all the 7.8×10^5 spectra that ranged from wavelengths $3,700 - 10,000$ Å.

For both the FNN and the CNN, the training process used the Keras command `model.fit()` (Chollet et al. 2015) that included input parameters: *batch size*, and epochs. Additionally, both models used SGD for the optimization process that used

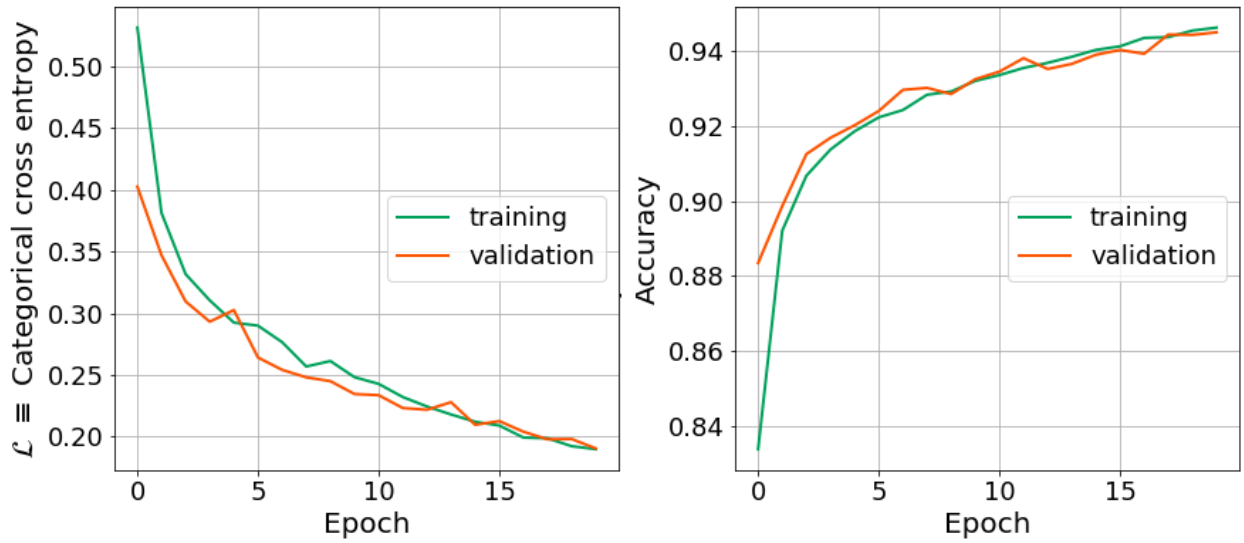

Fig. 5. The loss (*left*) and accuracy (*right*) curves for FNN for the training and validation data sets. A quick summary of the FNN parameters: batch-size of 40, learning rate 0.0001, momentum 0.0 were chosen for 20 epochs.

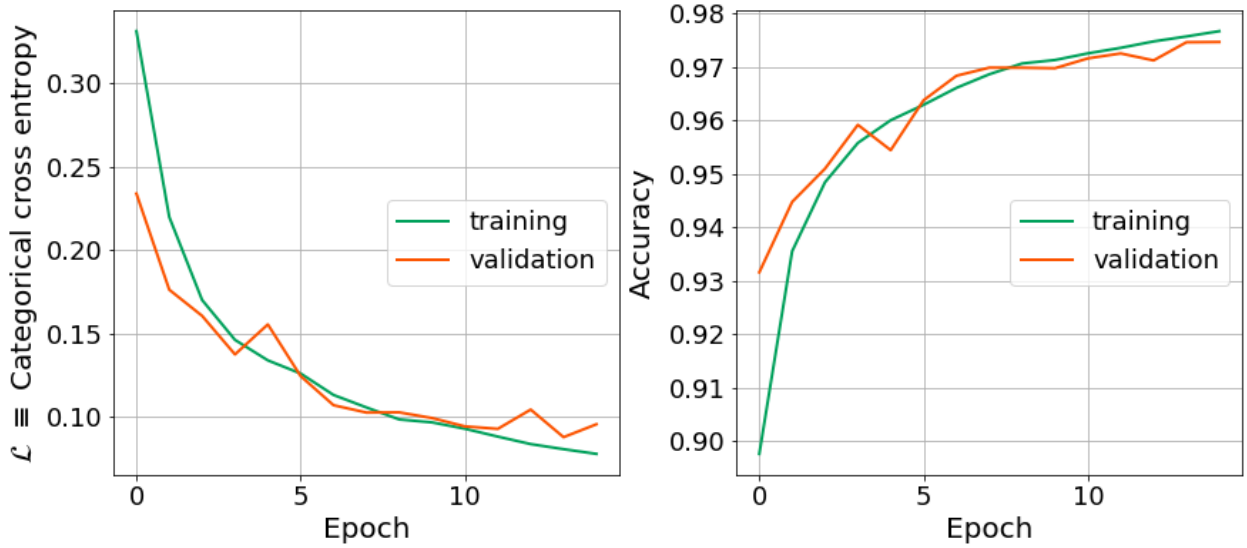

Fig. 6. The loss (*left*) and accuracy (*right*) curves for CNN for the training and validation data sets. A quick summary of the CNN parameters: batch-size of 30, learning-rate of 0.001, momentum of 0.0 were chosen for 15 epochs.

tunable parameters like *learning rate* and *momentum*. The definitions of all these parameters are briefly summarized in Tab. 1. These four parameters, described in Tab. 1, along with activation functions for every NN layers, the number of layers, and the number of neurons in every layer affect the model accuracy. This effect was systematically studied in the project; the results of testing these parameters and the final NN model generated are explained in Sec. 4.

4. Results and Discussions

4.1. Loss and Accuracy curves for FNN and CNN

The accuracy of a model is a measure of the number of test spectra that are classified correctly by the NN. An equivalent measure can be obtained by calculating the error rate or loss of the trained model. However, in machine learning we are also interested to minimize the test loss, also called as the *validation loss*. The validation loss corresponds to the loss encountered by the model for a new input i.e. the loss represents the failure rate of the model to classify test data (validation data). An ideal case for a well-trained model is when the gap between the training loss and the validation loss is small. Sec. 2.2 briefly introduced the problem of overfitting that here, graphically translates to a large gap between the training and validation loss values. Similarly, the case of underfitting is described as the scenario where the training loss value is large (Goodfellow et al. 2016, p.101). The final loss and accuracy curves for the training and validation processes for the two NNs, FNN and CNN, are shown in Fig. 5 and Fig. 6 respectively. Both models show desired behaviour by avoiding overfitting, as the gap between the training and validation curves are negligible. The accuracy achieved by the FNN was 94.5% with a loss of 19.03%. CNN achieved a higher accuracy of 97.5% with a lower loss of 9.5%. Thus, the CNN outperformed the FNN, as would be expected. Additionally, the FNN displays the case of underfitting relative

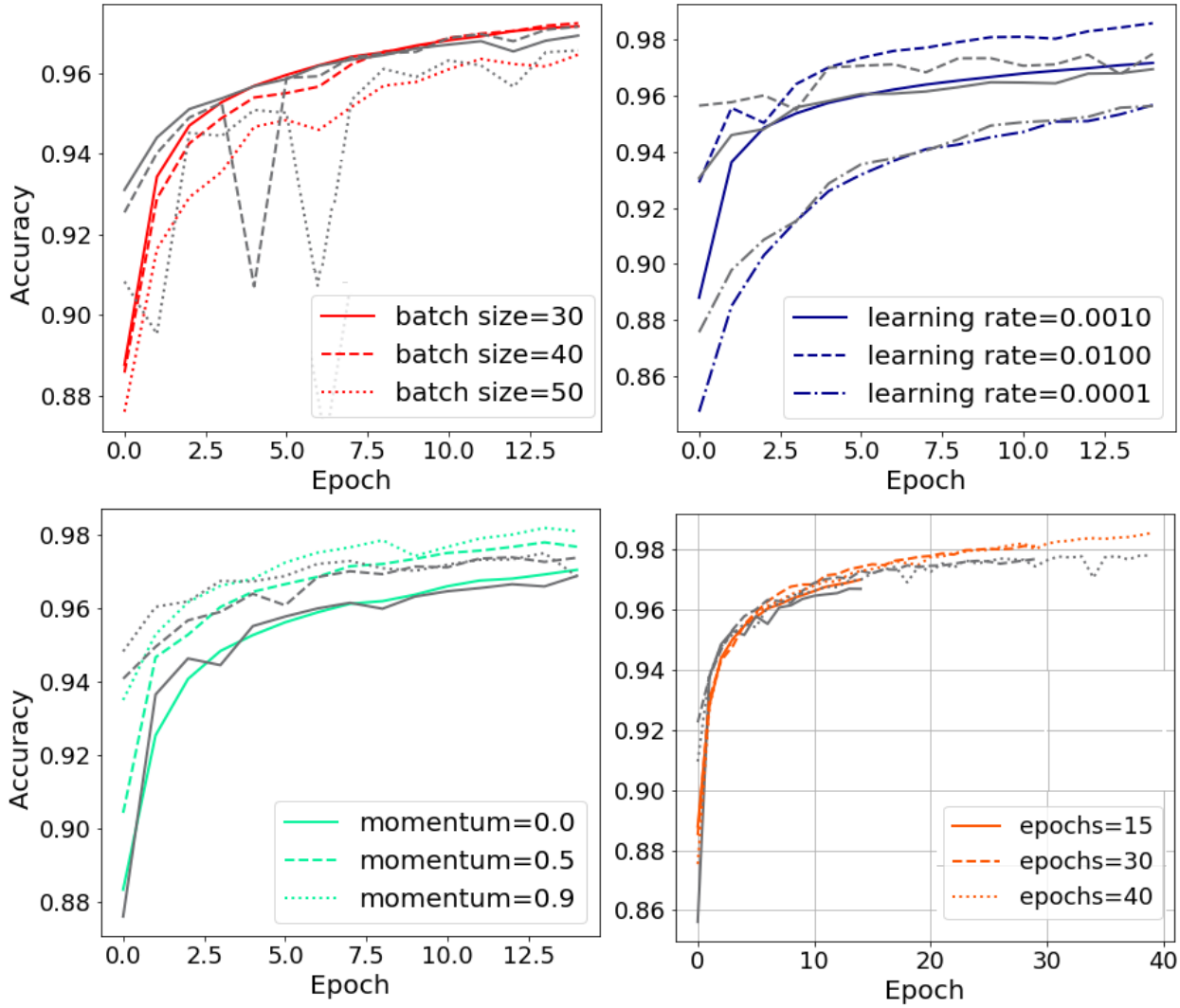

Fig. 7. Summary of the effect of changing the SGD optimization parameters: learning rate and momentum, and training parameters: batch-size and epochs on the accuracy of CNN model. In all the four plots, the coloured lines represent the training data set while the grey lines correspond to the validation or test data set.

to the CNN as the validation[2] loss of FNN (19.0%) is higher than that of CNN (9.5%).

4.2. Exploring the Parameters of the FNN and CNN

Making note of the resulting desirable features, the parameters were adjusted to achieve these features in a systematic way that is explained here. The effect of varying the parameters like the batch-size, learning-rate, momentum and epochs on the loss and accuracy curves of the NN were monitored. Fig. 7 depicts this study for the CNN; a similar study was conducted for the FNN. When a parameter was varied, like the batch-size, all the remaining parameters were kept fixed. Based on the best result for the loss and accuracy curves, the optimal batch-size was chosen. Subsequently, the learning rate was varied for fixed values of batch-size, epochs, and momentum and so forth. In the case of Fig. 7 (*top-left*), both batch-sizes of 30 and 40 generated resembling accuracy curves. However, the effect of varying learning rate (*top-right*) largely influenced the achieved accuracy. Larger learning rates allows achieving higher accuracy at the cost of overfitting the data. As for momentum (*bottom-left*), larger momentum also allowed for marginally higher accuracies. Finally, the epochs were stopped at 15 because running the learning process over more epochs only resulted in overfitting, as can be observed by the increase in the gap between the training and validation curves in Fig. 7 (*bottom-right*). So the two NNs were enhanced by this procedure albeit, further investigations of the parameter space and addition of more NN layers are possibilities that could result in better models. Posterior to the generated results, the question remains: *how confidently can these models be trusted to guide us towards detection of lensed spectra?*

4.3. Confidence Values for the 3-Category Classification

For every input spectrum, an output value is generated for all three categories. For example, if an arbitrary input spectrum generated output values, ranging from 0 to 1, for the category galaxy as 0.6, QSO as 0.2 and other objects as 0.2, we can conclude that input spectrum is most probable to be from a galaxy opposed to a spectrum from a QSO or another

[2] Here validation and training loss are approximately the same at the end of 15 epochs, hence, can be used interchangeably.

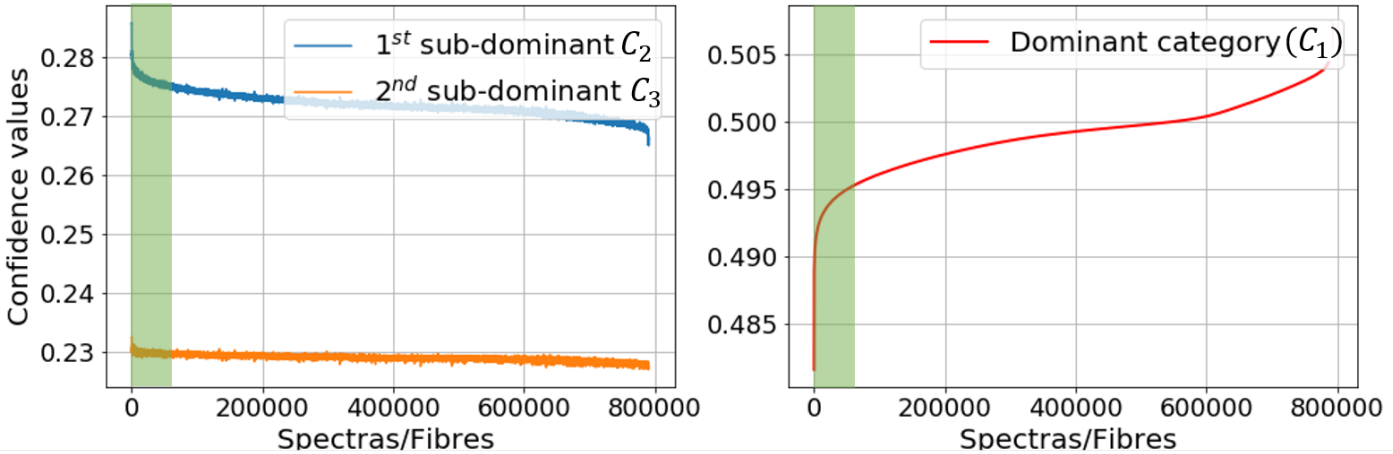

Fig. 8. *Left*: The confidence values for the two sub-dominant categories sorted as a function of the spectra. *Right*: The confidence values for the *dominant category* plotted as a function of spectra. These spectra are sorted in an ascending manner based on the magnitude of the confidence values. The highlighted box in both plots approximates the region where it is most probable to detect lensed spectra in the data.

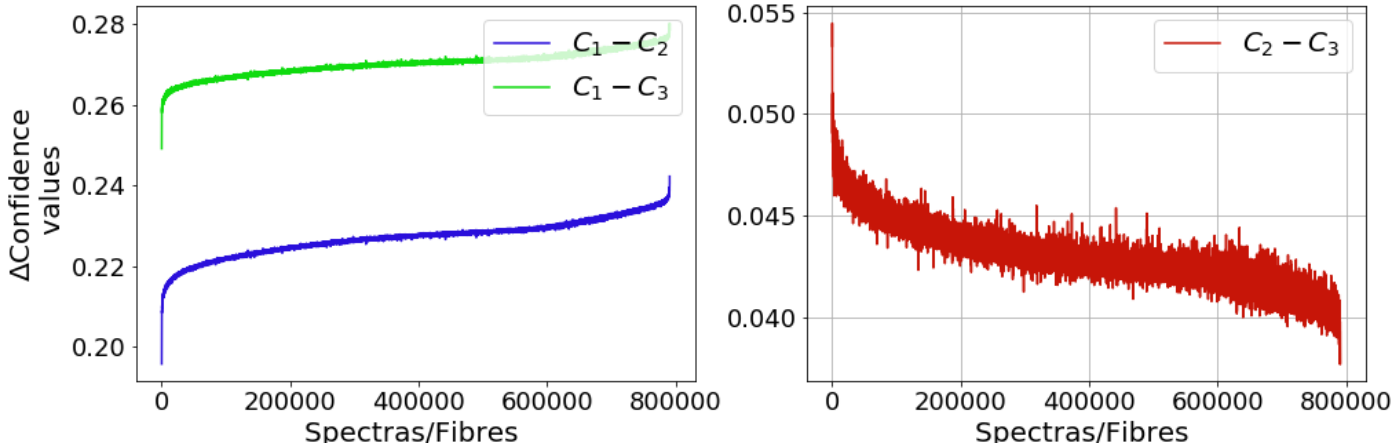

Fig. 9. *Left*: The difference in confidence values between the dominant and sub-dominant categories ($C_1 - C_{2/3}$) is plotted as a function of the sorted spectra. *Right*: The difference in confidence values between the two sub-dominant categories is plotted as a function of the sorted spectra.

astrophysical-object. In this case, the category galaxy is called the *dominant category* C_1 while QSO and other objects are the *sub-dominant categories*, C_2, and C_3 respectively. Fig. 8 shows these confidence values for the dominant (*left*) and sub-dominant (*right*) categories for about 0.8 million spectra that are classified using CNN. Furthermore, the difference between the dominant category and the sub-dominant categories ($C_1 - C_{2/3}$), and the difference between the two sub-dominant categories ($C_2 - C_3$) are also plotted in Fig. 9. It must be noted that these spectra are sorted in the ascending order of their confidence values and so, the plot does not convey the information about which category the spectra belongs. However, for current purposes, we are only interested in the spectra where C_1 is similar to $C_{2/3}$. In theory, spectra with this condition are more probable to be double spectra. The highlighted box in Fig. 8 shows the region of interest for detecting such lensed spectra. The boundary of the highlighted region is an approximation that roughly signifies the point after which the gradient approaches a constant value. However, the highlighted region could easily manifold itself as noisy spectra or it could direct us towards some of the shortcomings in the NN model used to classify the spectra. An analogous way of approaching the issue is by demanding that lensed spectra must be present in the region where the difference in confidence values between $C_1 - C_{2/3}$ is minimum (see Fig. 9). These claims demand the requirement of diagnostic tools to check if the spectra in the region of interest contains lensed spectra.

4.4. Proposed Methods for Further Diagnosis of Lensed Spectra

To summarize the previous discussion, we are interested in spectra were $C_1 - C_{2/3}$ is minimized. Once these potential spectra are collected there are three diagnostic methods proposed for detecting the lensed spectra:

1. **Using visual aids**: This method would use the space-coordinates of the observed astrophysical-object from the spectra and extract the visual images of the candidates from previous sky surveys. These visual images can then be sampled manually or undergo a further filtering process through deep learning algorithms like the one offered by Lanusse et al. (2017).
2. **Previous techniques**: As initially proposed by Bolton et al. (2008), this method would require fitting the potential sample of spectra with templates and observing the background spectrum to detect tracers for gravitational lensing.

3. **Using Active Learning**: This is an advanced technique in artificial intelligence where the learning algorithm can interactively learn and achieve higher accuracy while using fewer labelled data. This is accomplished by using human expertise to aid the learning process (Settles 2009).

5. Conclusions and Future Work

Two deep learning algorithms, FNN and CNN, have been proposed to classify BOSS and eBOSS spectra into three categories: galaxies, quasars, and other astrophysical objects. A systematic study was conducted to obtain the optimal values for the parameters that are crucial in building the NN. The confidence value distribution for the categorical classification using CNN was studied and this lead to the conclusion that the region where $C_1 - C_{2/3}$ is minimum is the most probable region for detection of lensed spectra. Albeit, this conclusion is not without shortcomings and defining the boundaries for this region of interest requires further research. Therefore, three different diagnostic methods are proposed to validate this conclusion which will be an interesting topic to explore for future work on this project. Additionally, we also aim to understand the ratio of lensed spectra present in the data set, define the characteristics of such spectra, and develop tools that can be used to automate the detection of lensed spectra.

All the code for this project is made available at the following GitHub repository: https://github.com/SoumyaShreeram/Analyzing_spectra_with_ML/.

References

Ahn, C. P., Alexandroff, R., Prieto, C. A., et al. 2014, The Astrophysical Journal Supplement Series, 211, 17

Aihara, H., Prieto, C. A., An, D., et al. 2011, The Astrophysical Journal Supplement Series, 193, 29

Blanton, M. R., Bershady, M. A., Abolfathi, B., et al. 2017, The Astronomical Journal, 154, 28

Bolton, A. S., Burles, S., Koopmans, L. V., et al. 2008, The Astrophysical Journal, 682, 964

Bouchain, D. 2006, Institute for Neural Information Processing, 2007

Brownstein, J. R., Bolton, A. S., Schlegel, D. J., et al. 2011, The Astrophysical Journal, 744, 41

Chollet, F. et al. 2015, Keras, https://keras.io

Dawson, K. S., Kneib, J.-P., Percival, W. J., et al. 2016, The Astronomical Journal, 151, 44

Dawson, K. S., Schlegel, D. J., Ahn, C. P., et al. 2012, The Astronomical Journal, 145, 10

Dey, A., Schlegel, D. J., Lang, D., et al. 2019, The Astronomical Journal, 157, 168

eBOSS sdss.org. 2018, eBOSS

Frieman, J. A., Bassett, B., Becker, A., et al. 2007, The Astronomical Journal, 135, 338

Goodfellow, I., Bengio, Y., & Courville, A. 2016, Deep Learning (MIT Press), http://www.deeplearningbook.org

Gunn, J. E., Siegmund, W. A., Mannery, E. J., et al. 2006, The Astronomical Journal, 131, 2332

Lanusse, F., Ma, Q., Li, N., et al. 2017, Monthly Notices of the Royal Astronomical Society, 473, 3895

Lau, M. M. & Lim, K. H. 2017, in 2017 2nd international conference on control and robotics engineering (ICCRE), IEEE, 201–206

LeCun, Y., Bengio, Y., et al. 1995, The handbook of brain theory and neural networks, 3361, 1995

Morningstar, W. R., Hezaveh, Y. D., Levasseur, L. P., et al. 2018, arXiv preprint arXiv:1808.00011

Ruder, S. 2016, An overview of gradient descent optimization algorithms

Schmidhuber, J. 2015, Neural networks, 61, 85

Settles, B. 2009, Active learning literature survey, Tech. rep., University of Wisconsin-Madison Department of Computer Sciences

Smee, S. A., Gunn, J. E., Uomoto, A., et al. 2013, The Astronomical Journal, 146, 32

Svozil, D., Kvasnicka, V., & Pospichal, J. 1997, Chemometrics and intelligent laboratory systems, 39, 43

Theuwissen, A. J. 2006, Solid-state imaging with charge-coupled devices, Vol. 1 (Springer Science & Business Media)

Yanny, B., Rockosi, C., Newberg, H. J., et al. 2009, The Astronomical Journal, 137, 4377

York, D. G., Adelman, J., Anderson Jr, J. E., et al. 2000, The Astronomical Journal, 120, 1579