MNRAS **000**, 1–11 (2018) Preprint 24 September 2018 Compiled using MNRAS LATEX style file v3.0

Predicting Galaxy Metallicity from Three-Color Images using Convolutional Neural Networks

John Wu[1⋆] and Steven Boada[1]

[1] *Physics and Astronomy Department, Rutgers University, Piscataway, NJ 08854-8019, USA*

Accepted XXX. Received YYY; in original form ZZZ

ABSTRACT

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

1 INTRODUCTION

Large-area sky surveys, both on-going and planned, are revolutionizing our understanding of galaxy evolution. The on going Dark Energy Survey (DES; The Dark Energy Survey Collaboration 2005) and planned Large Synoptic Survey Telescope (LSST; LSST Dark Energy Science Collaboration 2012) will survey vast swaths of the sky and create samples of galaxies much larger than any previously known. Spectroscopic follow-up will be key to a deep understanding of the properties of these galaxies, and constrain galaxy evolution through relations such as the mass-metallicity relation (hereafter MZR; Tremonti et al. 2004); or the fundamental metallicity relation, (hereafter FMR; e.g., Mannucci et al. 2010). But as the data sets continue to grow, individual galaxy spectroscopic follow-up becomes increasingly impractical.

Fortunately, the large data sets produced are ripe for the application of machine learing (ML) methods. ML is already contributing heavily to wide ranging studies investigating galaxy morphology (e.g., Dieleman et al. 2015; Huertas-Company et al. 2015; Beck et al. 2018; Dai & Tong 2018; Hocking et al. 2018), gravitational lensing (e.g., Lanusse et al. 2018; Petrillo et al. 2017, 2018), galaxy clusters (e.g., Ntampaka et al. 2015, 2016), star-galaxy separation (e.g., Kim & Brunner 2017), creating mock galaxy catalogs (e.g., Xu et al. 2013), and asteroid identification (e.g., Smirnov & Markov 2017) among many others.

In recent years, ML methods utilizing learning networks or "neural" networks have grown to prominence. While neural networks are a relatively old technique (e.g., LeCun et al. 1989), their recent increase in popularity is driven by the wide spread availability of cheap graphics processing units (GPUs) which can be used to do general purpose, highly parallel computing. Also, unlike more "traditional" ML methods, neural networks excel at image classification and regression problems.

Inferring spectroscopic properties from the imaging taken as part of a large-area photometric survey is, at a basic level, an image regression problem. These problems are most readily solved by use of convolutions in multiple layers of the network (see e.g., Krizhevsky et al. 2012). Convolution neural networks (CNNs, or convnets) efficiently learn spatial relations in images whose features are about the same sizes as the convolution filters (or kernels) which are to be learned through training. CNNs are considered *deep* when the number of convolutional layers is large, leading to the term "deep learning." Visualizing their filters reveals that increased depth permits the network to learn more and more abstract features (e.g., from Gabor filters, to geometric shapes, to faces; Zeiler & Fergus 2014).

In this work, we propose to use supervised ML, specifically CNNs, to analyze pseudo-three color images to predict the gas phase metallicity of a sample of galaxies taken as part of a large-area sky survey. We then used the predicted metallicities to recover the well established MZR.

This paper is organized as follows: In Section 2, we describe the acquisition and cleaning of the SDSS data sample. In Section 3 we discuss selection of the network's hyperparameters and outline training the network. We present the main results in Section 4 and discuss the results in the context of previous works. In Section 5, we discuss how our recovered MZR compares to other previously known relations. Finally, we summarize our key results and discuss possible future work in Section 6.

Unless otherwise noted, throughout this paper, we use a concordance cosmological model ($\Omega_\Lambda = 0.7$, $\Omega_m = 0.3$, and $H_0 = 70$ km s^{-1}Mpc^{-1}), assume a Chabrier initial mass function (Chabrier 2003), and use AB magnitudes (Oke 1974).

We don't quote uncertainties at all, and label RMSE or NMAD when we discuss scatter, so I am removing the mention of 1 σ uncertainties.

2 DATA

To create a large training sample, we optically select galaxies from the *Sloan Digital Sky Survey* (SDSS; York et al.

2000) DR7 MPA/JHU spectroscopic catalog (Kauffmann et al. 2003; Brinchmann et al. 2004; Salim et al. 2007). This catalog provides us with derived spectroscopic properties, stellar mass ($M_\star$), and gas phase metallicity (Z) estimates (Tremonti et al. 2004). We supplement the data from the spectroscopic catalog with photometry in each of the five SDSS photometric bands (u, g, r, i, z), along with associated errors from SDSS DR14 (Abolfathi et al. 2018).

We require that galaxies magnitudes are $10 < ugriz < 25$ mag, to avoid saturated, and low signal-to-noise detections. Galaxies should have colors $0 < u - r < 6$, to avoid extremely blue or extremely red objects and high confidence ($z_{err} < 0.01$) spectroscopic redshifts greater than $z = 0.02$. We also require that the r-band magnitude measured inside the petrosian radius (`petroMag_r`) be less than 18 mag, corresponding to the spectroscopic flux limit.

With these conditions we construct an initial sample of 142,186 galaxies.

2.1 SDSS Images

From this initial sample we create RGB (irg) image cutouts of each galaxy with the SDSS cutout service[1]. Images are scaled to be 128×128 pixels in size, corresponding to $15'' \times 15''$ on the sky. The native $0\farcs396$ SDSS pixel size are rescaled to $0\farcs296$ per pixel.

These RGB JPEG images form the inputs into our network. No further cleaning or filtering of the images is conducted.

3 TRAINING THE NETWORK

Before the CNN can be asked to make predictions, it must be trained to learn the relationships between the input data (the images described above) and the desired out put (the metallicity). Once the network is trained we use the test data set to assess the correctness of the predictive relationships.

We split our initial sample of $\sim 140,000$ images into random 60%, 20%, and 20% subsets, which comprise the training, validation, and test data sets respectively. All training images are seen by the network once every training epoch what is epoch?, although usually each epoch is split into a number of mini-batches which are learned in parallel. Mini-batches are usually small (~ 256) and potentially not representative of the full training sample – a technique used to prevent overfitting. Overfitting is when the network learns a specific and usually not general set of features which are then misapplied to the test data set.

Each mini-batch is fed forward through the input layers, where a random fraction of connections $p = 0.25, 0.50$ what? are removed between each linear layer (a "dropout"; Hinton et al. 2012; see Section A5). When the feed-forward network reports a prediction, a single value or a set of values, then a loss/cost function is used to compute how incorrect the prediction ($\hat{y}$) is from the true value y. We use the root mean squared error (RMSE $\equiv \sqrt{\langle|\hat{y} - y|^2\rangle}$) loss function, and seek to minimize it.

[1] `http://skyserver.sdss.org/dr14/SkyserverWS/ImgCutout/getjpeg`

We use gradient descent for each mini-batch to adjust each weight parameter, and each fractional contribution of loss is determined by the backpropagation algorithm (LeCun et al. 1989). The backpropagation algorithm is simply the chain rule applied to finite derivatives, and skipped? for any non-linear layers. The gradient is multiplied by the *learning rate* (see Section A3); batch-normalization is also applied in addition to a momentum term which ensures that the gradient is itself only changing slowly with each mini-batch. what is momentum

CNNs become difficult to train after many layers are added, likely because the network has already found the best representation possible at a shallower layer, and deeper layers simply noisily propagate the same signal and degrade the loss. To avoid this, we select a network architecture called a residual neural network, or *resnet* (He et al. 2016), which contains enhanced "shortcut connections" but is otherwise similar to other CNNs. Resnets have been shown to continue learning with increasing depth without the added cost of extra parameters. cite?

We use a 34-layer resnet (He et al. 2016), whose architecture consists of three layer groups. Our resnet is initialized to pre-trained weights from the ImageNet data set, which consists of 1.7 million images belonging to 1000 categories of objects found on Earth (e.g., cats, horses, cars, or books). The earlier layer groups generally have already trained filters that represent low-level abstractions, such as edges or Gabor filters, so we first train only the last layer group of the network by "freezing" the weights in the first two groups. By reducing the number of trainable parameters, we can rapidly approach the global loss minimum in a few number of epochs.

We train the final layers for two epochs using a learning rate of 0.1, and then "unfreeze" the earlier layers and train for another eight epochs using learning rates of [0.001, 0.01, 0.1] for the first, second, and third layer group respectively. For more details about our training methodology, such as the use of learning rate annealing, data augmentation, dropout, batch-normalization, and other hyperparameters, see Section A3.

Our training methods near convergence after 10 epochs, and additional training only marginally improves the loss. In total, our training steps requires 25-30 minutes on our GPU and uses under 2 GB of memory (depending on batch size).

Prediction using data augmentation (see Section A2) takes two minutes for our full test set of 20,466 images, or approximately 6 milliseconds per image (or a little over 1 millisecond per image without augmentation). In comparison, Huertas-Company et al. (2015) train their network for 10 days on a GPU following the Galaxy Zoo architecture.

We evaluate predictions using not only the RMSE, which approaches the standard deviation for Gaussian-distributed data, but also the NMAD, or the normal median absolute deviation (e.g., Ilbert et al. 2009; Dahlen et al. 2013; Molino et al. 2017)).

$$\mathrm{NMAD}(x) \approx 1.4826 \times \mathrm{median}\big(|x - \mathrm{median}(x)|\big), \quad (1)$$

where for a Gaussian-distributed x, the NMAD will also approximate the standard deviation, σ. NMAD has the distinct advantage in that it is insensitive to outliers for non-Gaussian distributions and is useful for comparing scatter.

4 RESULTS

Here we present the galaxy metallicities predicted by our trained network.

4.1 Example predictions

In Figure 1, we show examples of 128×128 pixel *irg* SDSS images that are evaluated by the CNN. Rows (a) and (b) depict the galaxies with lowest predicted and lowest true metallicities, respectively. The CNN has associated blue, edge-on disk galaxies with low metallicities, and is generally accurate in its predictions of low $Z_{\rm pred}$. In rows (c) and (d), we show the galaxies with highest predicted and highest true metallicities, respectively. Here we find that red galaxies containing prominent nuclei are predicted to be high in metallicity, and their predictions generally match $Z_{\rm true}$.

Galaxies predicted by our CNN to have high metallicities ($Z_{\rm pred} > 9.0$) tend to be characterized by high $Z_{\rm true}$, and the equivalent is true for low-metallicity galaxies. Inversely, galaxies with the highest (lowest) *true* metallicities in the sample are also predicted to have high (low) metallicities. Note that inclined galaxies tend to be lower in metallicity whereas face-on galaxies appear to be higher in metallicity. I can get the axis ratio if we want to do that test Tremonti et al. (2004) explain this correlation by suggesting that the SDSS fiber aperture captures more column of a projected edge-on disk, allowing the metal-poor, gas-rich, and less-extincted outer regions to more easily be detected and depress the integrated $Z_{\rm true}$.

We will now consider examples of the most incorrectly predicted galaxies. In rows (e) and (f), we show instances in which the CNN predicted too low metallicity and too high metallicity, respectively. The two galaxies with lowest residuals $\Delta Z \equiv Z_{\rm pred} - Z_{\rm true}$ (i.e., most under-predicted metallicities) suffer from artifacts which cause nonphysical color gradients.[2] Some of the other mistakes made by the CNN are similar to ones that would go against human intuition: blue, disky sources are generally thought of as lower in metallicity, and redder, more spheroidal objects tend to be higher in metallicity.

In the bottom row (g) of Figure 1, we show five randomly selected galaxies. The random SDSS assortment consists of elliptical, spiral, and possibly even an interacting pair of galaxies. Residuals are low (below 0.15 dex), and we again find that the CNN predictions follow human visual intuition.

4.2 Comparing Predicted and True Metallicities

In Figure 2, we show the distributions of true and predicted metallicities as a black and red histogram respectively. The histogram bin sizes are chosen according to the Freedman & Diaconis (1981) rule for each distribution. The discreet striping of the Tremonti et al. (2004) and Brinchmann et al. (2004) metallicity estimator appears in the $Z_{\rm true}$ distribution but does not manifest in our CNN predictions. This should increase the scatter in our distribution of residuals.

As we have described previously, the range of $Z_{\rm pred}$ is more limited than the range of $Z_{\rm true}$. Too narrow a domain in $Z_{\rm pred}$ will lead to systematic errors, as the CNN will end up never predicting very high or very low metallicities. Although the two distributions are qualitatively consistent with each other at low metallicities (e.g., $Z < 8.5$). check numbers However, the fraction of galaxies with high $Z_{\rm true} > 9.1$ ($2573/20466 = 12.6\%$) is more abundant than the fraction with high $Z_{\rm pred} > 9.1$ ($1174/20466 = 5.7\%$).

We find that the mode of the predicted metallicity distribution is higher than the mode of $Z_{\rm true}$. This result may be a consequence of the CNN overcompensating for its systematic under-prediction of metallicity for galaxies with $Z_{\rm true} > 9.1$. However, its effect on the entire distribution is small, and may be remedied simply by increasing the relative fraction of very high-$Z_{\rm true}$ objects. We find overall good agreement between the $Z_{\rm pred}$ and $Z_{\rm true}$ distributions. t-test? quantify

4.3 Scatter in $Z_{\rm pred}$ and $Z_{\rm true}$

In Figure 3, we compare the distributions of $Z_{\rm true}$ and $Z_{\rm pred}$ using a two-dimensional histogram (shown in grayscale in the main, larger panel). We also show the median predictions varying with binned $Z_{\rm true}$ (solid red line), along with the scatter in RMSE (dashed blue) and NMAD (dashed orange), and also the one-to-one line (solid black). The running median agrees well with the one-to-one line, although at low metallicity we find that the CNN makes makes overpredictions.

A histogram of metallicity residuals is shown in the inset plot of the Figure 3 main panel. The ΔZ distribution is characterized by an approximately normal distribution with a heavy tail at large positive residuals; this heavy tail is likely due to the systematic over-prediction of low-$Z_{\rm true}$ galaxies. There is also an overabundance of large negative ΔZ corresponding to under-predictions at high $Z_{\rm true}$, although this effect is smaller (despite appearing to be more significant in Figure 2).

We now turn our attention to the upper panel of Figure 3, which shows how the scatter varies with spectroscopically derived metallicity. The RMSE scatter and outlier-insensitive NMAD are both shown. Marker sizes are proportional in area to the number of samples in each $Z_{\rm true}$ bin, and the horizontal lines are located at the average loss (RMSE or NMAD) for the full test data set.

Predictions appear to be both accurate and low in scatter for galaxies with $Z_{\rm true} \approx 9.0$, which is representative of a typical metallicity in the SDSS sample. Where the predictions are systematically incorrect, we find that the RMSE increases dramatically. However, the same is not true for the NMAD; at $Z_{\rm true} < 8.5$, it asymptotes to ~ 0.10, even though the running median is incorrect by approximately the same amount! This is because the NMAD determines the scatter about the *median* and not $\Delta Z = 0$, and thus, this metric becomes somewhat unreliable when the binned samples do not have a median value close to zero. Fortunately, the global median of ΔZ is -0.006, or less than 10% of the RMSE,

[2] Note that both are labeled as quasars according to their SDSS DR14 spectra. A. Baker mentioned that it is possible some of the incorrect predictions in Figure 1 may be due to the fact that the R_{23} estimator is a double-valued function, and the particular branch chosen may cause incorrect estimation of $Z_{\rm true}$. Such an effect may be possible if, e.g., one of the oxygen lines is at low SNR and the uncertainties are large.

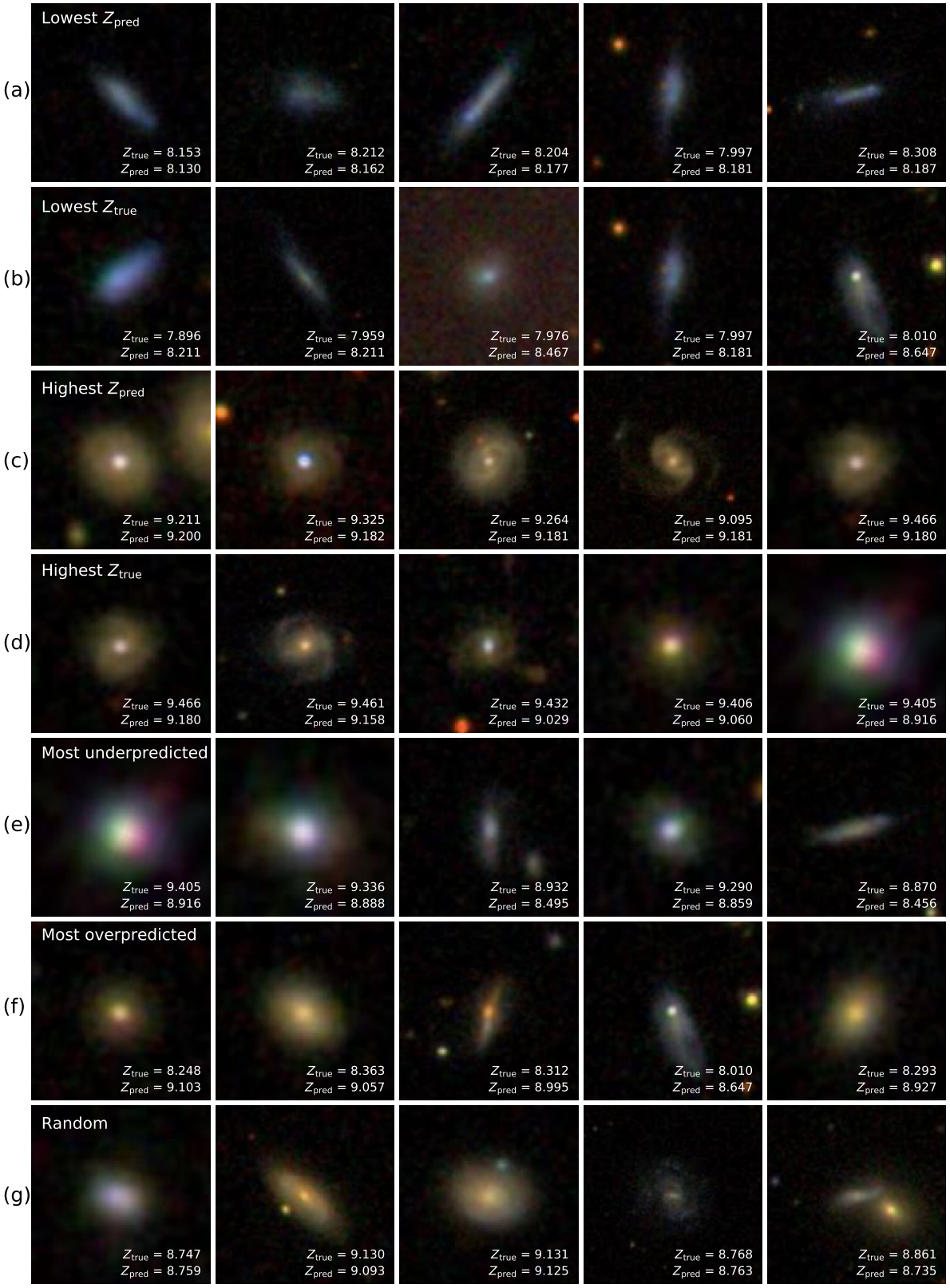

Figure 1. SDSS imaging with predicted and true metallicities from the test data set. Five examples are shown from each of the following categories: (a) lowest predicted metallicity, (b) lowest true metallicity, (c) highest predicted metallicity, (d) highest true metallicity, (e) most under-predicted metallicity, (f) most over-predicted metallcity, and (g) a set of randomly selected galaxies.

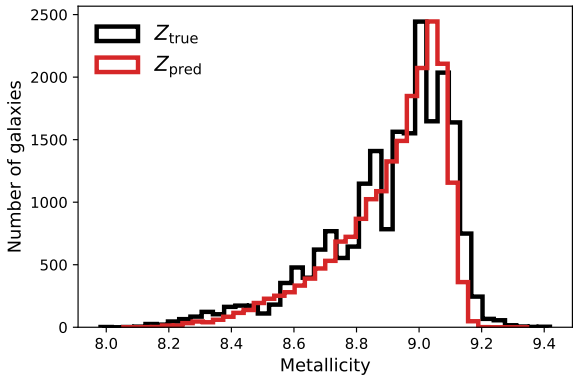

Figure 2. Distributions of the true (black) and predicted (red) galaxy metallicities. Note that the bin widths are different for each distribution. See text for details.

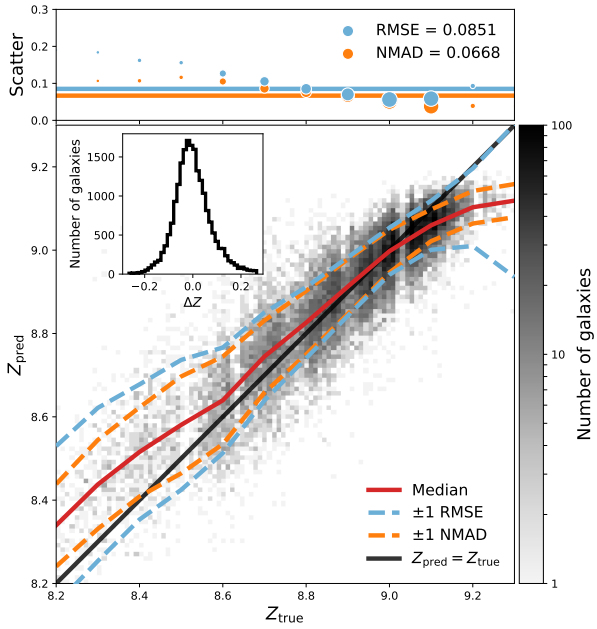

Figure 3. Bivariate distribution of true galaxy metallicity ($Z_{\rm true}$) and CNN predictions ($Z_{\rm pred}$) are shown in the main panel. Overlaid are the median predicted metallicity, solid red line, RMSE scatter, dashed blue line, NMAD scatter, dashed orange line, in bins of $Z_{\rm true}$. The solid black line shows the one-to-one relation. A distribution of residuals ($Z_{\rm pred} - Z_{\rm true}$) is shown in the inset plot. In the upper panel, we again show the binned scatter, where the size of each marker is proportional to the number of galaxies in that bin. Each horizontal line corresponds to the average scatter over the entire test data set (and global value indicated in the upper panel legend).

and thus the global NMAD = 0.0668 is representative of the outlier-insensitive scatter for the entire test data set.

This effect partly explains why the global NMAD (0.0668) is higher than the weighted average of the binned NMAD (~ 0.05).Also, each binned NMAD is computed using its local scatter, such that the outlier rejection criterion

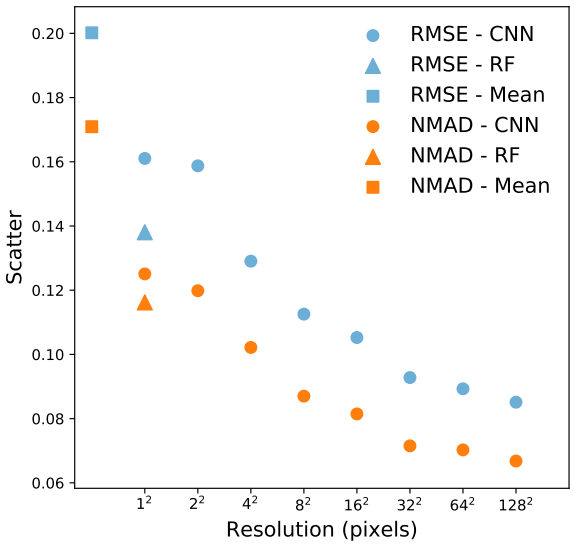

Figure 4. The effects of image resolution on CNN performance. Blue and orange circular markers indicate scatter in the residual distribution (ΔZ) measured using RMSE and NMAD, respectively. (Each point is analogous to the horizontal lines shown in Figure 3.) We also show predictions from a random forest algorithm as stars-shaped markers, and constant $\langle Z_{\rm true} \rangle$ predictions as square markers.

varies with $Z_{\rm true}$. To illustrate this effect with an example: $\Delta Z \approx 0.2$ would be treated as an $3\,\sigma$ outlier at $Z_{\rm true} = 9.0$, where the CNN is generally accurate, but the same residual would not be rejected as an outlier using NMAD for $Z_{\rm true} = 8.5$. Since the binned average NMAD depends on choice of bin size, we do not include those results in our analysis and only focus on the global NMAD.

4.4 Resolution effects

Because our methodology is so computationally light, we can run the same CNN training and test procedure on images scaled to different sizes in order to consider the effects of image resolution. Our initial results use SDSS $15'' \times 15''$ cutouts resized to 128×128 pixels, and we now downsample the same images to 64×64, 32×32, $\cdots$, 2×2, and even 1×1 pixels via re-binning. All images retain their three channels, so the smallest 1×1 image is effectively the pixels in each of the *irg* bands averaged together with the background and possible neighboring sources.

In Figure 4, we show the effects of image resolution by measuring the global scatter in ΔZ using the RMSE and NMAD metrics (shown in blue and orange circular markers, respectively). Also shown is the scatter in ΔZ if we always predict the mean value of $Z_{\rm true}$ over the data set (shown using a square marker). This constant prediction is effectively the worst-possible scatter, and the Tremonti et al. (2004) systematic uncertainty in $Z_{\rm true}$ of ~ 0.03 dex is the best-possible scatter. We find that both RMSE and NMAD de-

crease with increasing resolution, as expected if morphology or color gradients are instrumental to predicting metallicity.

There appears to be little improvement in scatter going from 1×1 to 2×2 pixel images. 1×1 three-color images contain similar information to three photometric data points (although because of the included background and neighboring pixels, it is less useful than photometry), which can be used to perform a crude spectral energy distribution (SED) fit. Therefore it unsurprising that the 1×1 CNN predictions perform so much better than the baseline mean prediction. A 2×2 three-color image contains four times as many pixels as a 1×1 image, but because the object is centered between all four pixels, this information is still averaged among all available pixels. Therefore, the scatter does not improve appreciably between 1×1 and 2×2 resolutions.[3]

The scatter is a strong function of resolution as the images are resolved from 2×2 to about 32×32. With further increasing resolution, improvement is still evident, although the scaling with scatter is noticeably weaker. Because the angular size of each image cutout stays the same, the pixel scale changes from $0\farcs469$/pix for 32×32 images, to $0\farcs234$/pix for 64×64 images, to $0\farcs117$/pix for 128×128 images. The native SDSS pixel resolution is $0\farcs396$/pix, such that the 64×64 and 128×128 resolutions result in the oversampling of each image! Thus, the plateauing of scatter with resolution is expected for images larger than 32×32. It is worth noting, however, that the CNN attempts to learn filters which depend on the size of the input image, and that smaller images may result in the CNN training filters that are too low in resolution to be completely effective for prediction. Therefore, it is also not surprising that the CNN makes incremental gains for images with increasing resolution beyond 32×32 pixels.

4.5 Random Forest Predictions for Metallicity

We also construct a random forest (RF) of decision trees in order to predict metallicity using the implementation from `scikit-learn` (Pedregosa et al. 2012). Hyperparameters are selected according to the optimal RF trained by Acquaviva (2016). We use exactly the same data labels (i.e., galaxies) to train/validate or test the RF as we have used for training and testing the CNN, so that our measurements of scatter can be directly compared. However, we have used the *gri* three-band photometry data (given in magnitudes) to train and predict metallicity. Since each galaxy only has three pieces of photometric information, it can be compared to the 1×1 three-band "images" processed by our CNN.

We note that the RF outperforms the CNN results using 1×1 and 2×2 images. This result is unsurprising because the RF is supplied aperture-corrected photometry, whereas the CNN is provided 1×1 *irg*"images" whose features have been averaged with their backgrounds. 2×2 images are only marginally more informative. When the resolution is further increased to 4×4 images, then the CNN can begin to learn rough morphological features and color gradients, which is already enough to surpass the performance (measured by both RMSE and NMAD) of the RF. This result suggests that the CNN is able to learn a nontrivial representation of gas-phase metallicity based on three-band brightness distributions, even with extremely low-quality data.

[3] There is extra information in the 2×2 pixel images in non-circularly symmetric cases. For an inclined disk, it is possible to roughly determine the orientation in the sky plane, but this information is not very useful. In the case of a major merger or interacting companion, the 2×2 images may be more powerful than 1×1 images.

4.6 Comparisons to Previous Works

CNNs have been used for a wide variety of classification tasks in extragalactic astronomy, including morphological classification (e.g., Dieleman et al. 2015; Huertas-Company et al. 2015; Simmons et al. 2017), distinguishing between compact and extended objects (Kim & Brunner 2017), selecting observational samples of rare objects based on simulations (Huertas-Company et al. 2018; Lanusse et al. 2018), and visualizing high-level morphological galaxy features (Dai & Tong 2018). These works seek to improve classification of objects into a discreet number of classes, i.e., visual morphologies. Our paper uses CNNs to tackle the different problem of regression, i.e., predict values from a continuous distribution.

Examples of regressing stellar properties in the astronomical ML literature (e.g., Bailer-Jones 2000; Fabbro et al. 2018), they train on synthetic stellar spectra and test on real data. Their predicted measurements of stellar properties, e.g., stellar effective temperature, surface gravity, or elemental abundance, are able to be derived from the available training data set. Our work is novel because we predict metallicity, a spectroscopically determined galaxy property, using only three-color images. Said another way, it is not necessarily the case that Z can be predicted from our training data. However, we find that galaxy morphology supplements color information that is useful for predicting metallicity.

A similar study to this work is that of Acquaviva (2016). The authors use a variety of machine learning methods including RFs, extremely random trees (ERTs), boosted decision trees (AdaBoost), and support vector machines (SVMs) in order to estimate galaxy metallicity. The Acquaviva (2016) data set consisted of a $z \sim 0.1$ sample (with $\sim 25,000$ objects) and a $z \sim 0.2$ sample (with $\sim 3,000$ objects), each of which had five-band SDSS photometry (*ugriz*) available as training data. These samples are sparsely populated at low metallicities, and they contain a smaller fraction of objects with $Z_{\rm true} < 8.5$ than our sample, but are otherwise similarly distributed in $Z_{\rm true}$ to ours. The estimates of Z come from the same place, so we are really using a similar catalog. Why are there so much fewer objects in her catalog?

We will first compare RF results, since this technique is common to both of our analyses, and reveals important differences in our training data. Because outliers are defined differently in both works, we will use the RMSE metric to compare scatter between the two. Acquaviva (2016) obtained RMSE of 0.081 and 0.093 dex when using RFs on the five-band photometry. Using the same RF approach on a larger sample, while working with only *three* bands of photometric information, we find RMSE = 0.130 dex. Our scatter is larger than the value reported by Acquaviva (2016) by a factor of $\sim 150\%$. This result may partly be explained by the fact that their $Z_{\rm true}$ distribution is narrower than for our training data set, or the fact that our data set spans

a broader range in galaxy redshift; however, some of this advantage is offset by our larger sample size.

Ultimately, it appears the extra u and z bands supply machine learning algorithms with valuable information for predicting metallicity. Indeed, the u and z-bands convey information about the galaxy's star formation rate (SFR) and stellar mass ($M_\star$) can we find a citation for this?. For this reason, it is possible that the RF trained on five-band photometry can estimate $Z_{\rm true}$ down to the limit of the FMR, which has very small scatter (~ 0.05 dex) at fixed $M_\star$ *and* SFR. The g, r, and i bands are rather insensitive to the SFR, but can still provide some information about the stellar mass, and so its results are more linked to the MZR rather than the FMR.

Regardless of these limitations, our CNN is able to estimate metallicity with $\Delta Z = 0.085$ dex, which is comparable to the scatter in residuals using the best algorithms from Acquaviva (2016). There is evidence that the morphological information provided by using images rather than photometric data is helping the CNN perform so well: (1) the RMSE scatter decreases with increasing image resolution, and (2) it identifies edge-on galaxies as lower-$Z_{\rm true}$ and face-on galaxies as higher-$Z_{\rm pred}$ (consistent with observational bias). Gradients in color, or identification of mergers (e.g., Ackermann et al. 2018) may also be helpful for predicting metallicity.

really like this section

5 THE MASS-METALLICITY RELATION

The MZR describes the tight correlation between galaxy stellar mass and nebular metallicity. Scatter in this correlation is approximately $\sigma \approx 0.10$ dex in $Z_{\rm true}$ over the stellar mass range $8.5 < \log(M_\star/M_\odot) < 11.5$ (Tremonti et al. 2004), where σ is the standard deviation of the metallicity and is equivalent to the RMSE for a normal distribution. The MZR can be characterized empirically using a polynomial fit:

$$Z = -1.492 + 1.847\log(M_\star/M_\odot) - 0.08026\left[\log(M_\star/M_\odot)\right]^2 . \quad (2)$$

The physical interpretation of the MZR is that a galaxy's stellar mass strongly correlations with its chemical enrichment. Proposed explanations of this relationship's origin include metal loss through blowout (e.g., Garnett 2002; Tremonti et al. 2004) inflow of pristine gas, or a combination of the two (Lilly et al. 2013); however, see also Sánchez et al. (2013). Although the exact physical process responsible for the low (0.10 dex) scatter in the MZR is not known, its link to SFR via the FMR is clear, as star formation leads to both metal enrichment of the interstellar medium and stellar mass assembly.

The FMR connects the instantaneous (~ 10 Myr) SFR with the gas-phase metallicity (~ 1 Gyr timescales; see, e.g., Leitner & Kravtsov 2011) and $M_\star$ (i.e., the ~ 13 Gyr integrated SFR). Our CNN is better suited for predicting $M_\star$ rather than SFR, using the *gri* bands, which can only weakly probe the blue light from young, massive stars. Therefore, we expect the scatter in CNN predictions to be limited by the MZR (with scatter $\sigma \sim 0.10$ dex) rather than the FMR ($\sigma \sim 0.05$ dex). It is possible that galaxy color and morphology, in tandem with CNN-predicted stellar mass, can be used to roughly estimate the SFR, but in this paper we will focus on only the MZR.

5.1 Predicting Stellar Mass

Since galaxy stellar mass is known to correlate so strongly with metallicity, and is easier to predict (than, e.g., SFR) from *irg* imaging, we consider the possibility that the CNN is simply predicting $M_{\star,\rm pred}$ accurately and then learning the simple polynomial transformation in order to predict metallicity. We can simulate this method by training the CNN on $M_{\star,\rm true}$ and then converting the stellar mass predictions to metallicities using Equation 2.

We re-run the CNN methodology to train and predict $M_\star$ using the 142,145 galaxies out of the original 142,186 that have available stellar mass measurements. These results are shown in the left panel of Figure 5. From the same subsample as before (minus three which do not have $M_\star$), we verify that $M_{\star,\rm true}$ median agrees with the median of $M_{\star,\rm true}$ for values between $9.0 \lesssim \log M_\star/M_\odot \lesssim 10.5$. The RMSE scatter in the $M_\star$residuals is ~ 0.22 dex, and the NMAD is ~ 0.20 dex. The slope of the empirical MZR at $\log(M_\star/M_\odot) \sim 10$ is (0.4 dex in Z)/(1.0 dex in $M_\star$), implying that the CNN might be able to leverage the MZR and predict metallicity to ~ 0.08 dex (plus any intrinsic scatter in the MZR, in quadrature).

We use Equation 2 and $M_{\star,\rm pred}$ to predict metallicity, which we call $Z_{\rm MZR}$. In the right panel of Figure 5, we compare $Z_{\rm MZR}$ against $Z_{\rm true}$. The scatter in residuals $Z_{\rm MZR} - Z_{\rm true}$ is 0.12 dex, which is significantly higher than the 0.085 dex scatter reported in Section 4. This evidence suggests that the CNN has learned to determine metallicity in a more powerful way than by simply predicting $M_{\star,\rm pred}$ and then applying a polynomial conversion.

5.2 Lowering the scatter in the MZR

The RMSE = 0.085 dex difference between the true and CNN-predicted metallicities can be interpreted in one of two ways: (1) the CNN is inaccurate, and $Z_{\rm pred}$ deviates randomly from $Z_{\rm true}$, or (2) the CNN is labeling $Z_{\rm pred}$ according to some other hidden variable, and ΔZ represents a non-random shift in predictions based on this variable. If the first scenario is true, then we would expect the random residuals to increase the scatter of known correlations such as the MZR. If the second is true, then we would expect the tightness of such known correlations to remain unchanged.

In Figure 6, we show true stellar mass ($M_{\star,\rm true}$) against CNN-predicted metallicity. For comparison, we also overlay the Tremonti et al. (2004) MZR and its scatter ($\sigma = 0.10$ dex). The empirical median relation (solid black) matches our predicted MZR median (solid red), and the lines marking observed scatter (dashed black) appear to match observed scatter as well (dashed blue and orange). Over the range $9.5 \leq \log(M_{\star,\rm true}/M_\odot) \leq 10.5$, the RMSE scatter in $Z_{\rm pred}$ (dashed blue) appears to be even tighter than the observed $\pm 1\,\sigma$ (dashed black). The same is true for the NMAD, which is even lower over the same interval.

In the upper panel of Figure 6, we present the scatter in both predicted and Tremonti et al. (2004) MZR binned by mass. We confirm that the CNN predicts a MZR that

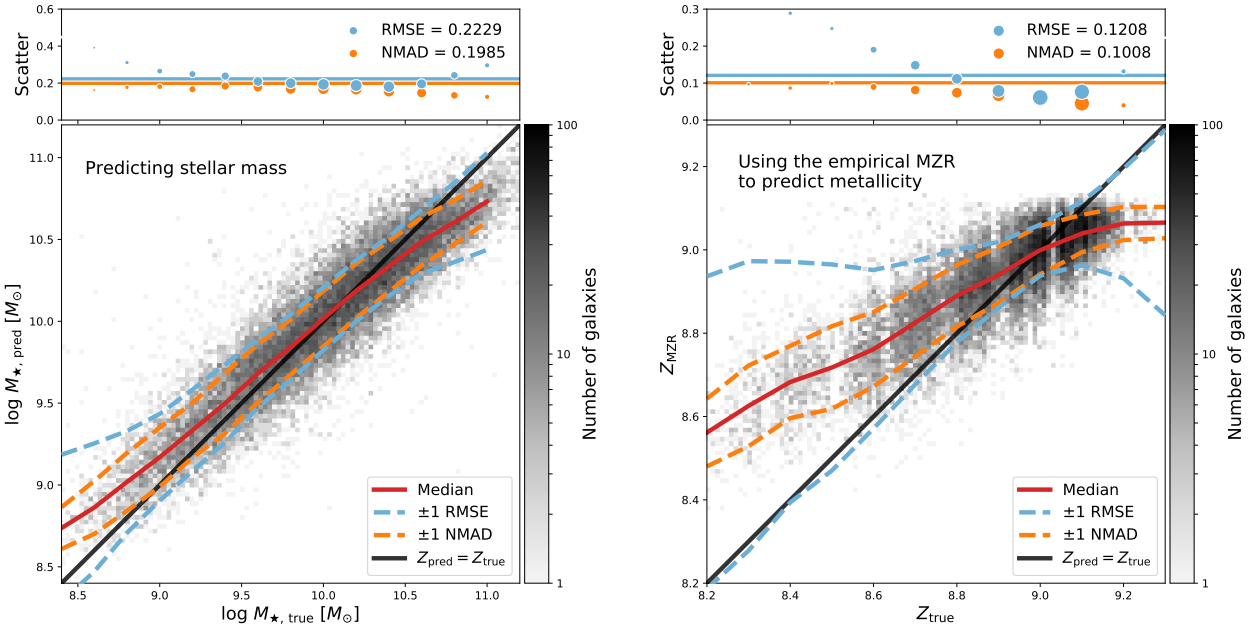

Figure 5. In the left panel, we show the CNN predicted galaxy stellar mass against true stellar mass. Colors and marker or line styles are the same as in Figure 3. In the right panel, we compare the predicted stellar mass converted to metallicity, assuming the Tremonti et al. 2004 MZR, with the true metallicity.

is at most equal (and possibly smaller) in scatter than one constructed using the true metallicity. The stellar mass bins for which the predicted RMSE is lower than measured σ are the ones which contain the most training examples. feel like this sentence is missing words Thus, it may be possible that if our data set was augmented to include additional low- and high-$M_{\star,\mathrm{true}}$ galaxies, then the predicted RMSE (and NMAD) may be even lower.

The fact that a CNN trained on only *irg* imaging is able to predict metallicity accurately enough to reproduce the MZR in terms of median and scatter is not trivial. The error budget is very small: $\sigma = 0.10$ dex affords only, e.g., 0.05 dex of scatter when SFR is a controlled parameter plus a 0.03 dex systematic scatter in Z_{true} measurements, leaving only ~ 0.08 dex remaining for CNN systematics.

This is somewhat compatible with our result of RMSE(ΔZ) = 0.085. However, this cannot be correct since it assumes that the CNN is recovering the FMR perfectly – and as we have discussed before, it is highly unlikely that the CNN is sensitive to the SFR and therefore cannot probe the MZR at individual values of the SFR. The error budget for the MZR is already exceeded, too, as we have found RMSE = 0.10 dex for both the $Z_{\mathrm{pred}} - M_{\star,\mathrm{true}}$ relation and the empirical MZR ($Z_{\mathrm{true}} - M_{\star,\mathrm{true}}$) without accounting for the fact that Z_{pred} and Z_{true} differ by RMSE = 0.085 dex!

We thus find more evidence that the CNN has learned something from the SDSS *irg* imaging that is different from, but at least as powerful as, the MZR. One way that this is possible is if the CNN can measure some version of metallicity that is more fundamentally linked to the stellar mass, rather than Z_{pred} as derived from oxygen spectral lines. Another possibility is that the MZR is a projection of a correlation between stellar mass, metallicity, and a third parameter, perhaps one that is morphological in nature. If this is the case, then the Tremonti et al. (2004) MZR represents a relationship that is randomly distributed in the yet unknown third parameter, while our CNN would be able to stratify the MZR according to this parameter (much like how the FMR does so with the SFR). We are unfortunately not able to identify any hidden parameter using the current CNN methodology, but we plan to explore this topic in a future work.

this is good stuff

6 SUMMARY

We have trained a deep, convolutional neural network (CNN) to predict galaxy gas-phase metallicity using only 128 × 128 pixel, three-band (*irg*), JPEG images take from the SDSS. Our conclusions are as follows:

(i) By training for a half-hour on a GPU, the CNN can achieve $Z_{\mathrm{pred}} - Z_{\mathrm{true}}$ residuals with RMSE = 0.085 dex (or NMAD = 0.067 dex if outliers are removed).

(ii) We find that the residual scatter decreases in an expected way as resolution is increased, suggesting that the CNN is leveraging the spatial information about the galaxy (morphology) to predict metallicity.

(iii) The CNN outperforms a random forest trained on *gri* photometry if provided images larger than 4×4 pixels, and is as accurate as a random forest trained on *ugriz* photometry when given 128 × 128 pixel *irg* images.

(iv) We find that scatter in the mass-metallicity relation (MZR) constructed using CNN-predicted metallicities is as tight as the empirical MZR ($\sigma = 0.10$ dex). Because predicted metallcities differ from the "true" metallicities by RMSE = 0.085 dex, the only way that the predicted MZR can have such low scatter is if the CNN has learned a con-

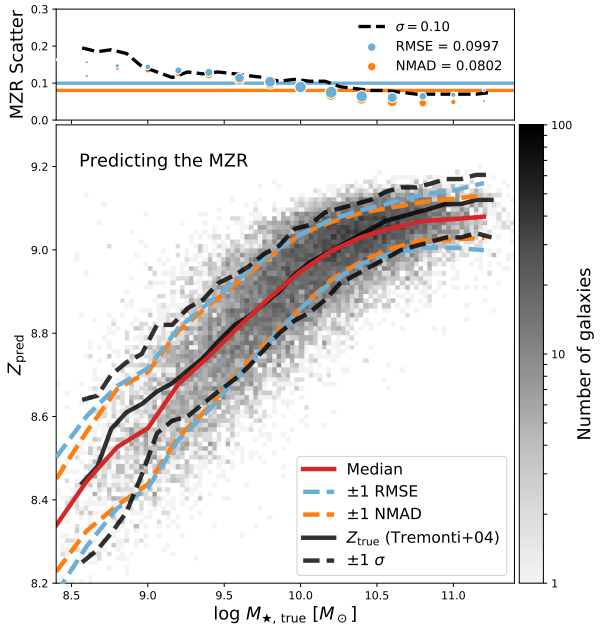

Figure 6. In the main panel, the predicted MZR comparing true $M_\star$ against CNN predicted $Z_{\rm pred}$ is shown in grayscale. The running median (solid red) and scatter (dashed blue and orange) are shown in 0.2 dex mass bins. For comparison, we also show the Tremonti et al. (2004) observed median and scatter binned by 0.1 dex in mass (solid and dashed black lines, respectively). In the top panel, we show the scatter in the predicted and empirical MZR. The standard deviation of the scatter in the MZR is shown as a dashed black line, while the blue and orange circles show the RMSE and NMAD, respectively, in bins of true $M_\star$. Marker sizes are proportional to the number of galaxies in each stellar mass bin. Global scatter in the CNN-predicted MZR appears to be comparable or even lower than scatter from the true MZR.

nection to metallicity that is more strongly linked to the stellar mass than the nebular lines.

Future work: not really thought about

An future extension to our work might be to repeat our analysis but to instead train on simulated data. Another is to feed SFR to the CNN and see if we can beat the FMR. We can also train CNNs individually on each mass bin and see if indeed we can predict metallicity as accurately.

ACKNOWLEDGEMENTS

JW is supported by XXXXXXXX grant. SB is supported by NASA Astrophysics Data Analysis grant number NNX14AF73G and NSF Astronomy and Astrophysics Research Program award number 1615657. The authors thank Eric Gawiser and Andrew Baker for helpful comments and discussions, and also thank David Shih and Matthew Buckley for use of their GPU cluster at Rutgers University High Energy Experimental Physics department. This research made use of the IPython package (Perez & Granger 2007) and matplotlib, a Python library for publication quality graphics (Hunter 2007). Funding for the SDSS and SDSS-II has been provided by the Alfred P. Sloan Foundation, the Participating Institutions, the National Science Foundation, the U.S. Department of Energy, the National Aeronautics and Space Administration, the Japanese Monbukagakusho, the Max Planck Society, and the Higher Education Funding Council for England. The SDSS Web Site is http://www.sdss.org/.

REFERENCES

Abolfathi B., et al., 2018, The Astrophysical Journal Supplement Series, 235, 42
Ackermann S., Schawinski K., Zhang C., Weigel A. K., Turp M. D., 2018, Monthly Notices of the Royal Astronomical Society, 479, 415
Acquaviva V., 2016, Monthly Notices of the Royal Astronomical Society, 456, 1618
Bailer-Jones C. A. L., 2000, A&A, 357, 197
Beck M. R., et al., 2018, Monthly Notices of the Royal Astronomical Society, 476, 5516
Brinchmann J., Charlot S., White S. D. M., Tremonti C., Kauffmann G., Heckman T., Brinkmann J., 2004, Monthly Notices of the Royal Astronomical Society, 351, 1151
Chabrier G., 2003, Publications of the Astronomical Society of the Pacific, 115, 763
Dahlen T., et al., 2013, The Astrophysical Journal, 775, 93
Dai J.-M., Tong J., 2018, preprint (arXiv:1807.05657)
Dieleman S., Willett K. W., Dambre J., 2015, Monthly Notices of the Royal Astronomical Society, 450, 1441
Fabbro S., Venn K. A., O'Briain T., Bialek S., Kielty C. L., Jahandar F., Monty S., 2018, Monthly Notices of the Royal Astronomical Society, 475, 2978
Freedman D., Diaconis P., 1981, Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete, 57, 453
Garnett D. R., 2002, ApJ, 581, 1019
He K., Zhang X., Ren S., Sun J., 2016, in Proceedings of the IEEE conference on computer vision and pattern recognition. pp 770–778
Hinton G. E., Srivastava N., Krizhevsky A., Sutskever I., Salakhutdinov R. R., 2012, eprint arXiv:1207.0580
Hocking A., Geach J. E., Sun Y., Davey N., 2018, Monthly Notices of the Royal Astronomical Society, 473, 1108
Huertas-Company M., et al., 2015, The Astrophysical Journal Supplement Series, 221, 8
Huertas-Company M., et al., 2018, The Astrophysical Journal, 858, 114
Hunter J. D., 2007, Computing in Science & Engineering, 9, 90
Ilbert O., et al., 2009, The Astrophysical Journal, 690, 1236
Kauffmann G., et al., 2003, Monthly Notices of the Royal Astronomical Society, 341, 33
Kim E. J., Brunner R. J., 2017, Monthly Notices of the Royal Astronomical Society, 464, 4463
Krizhevsky A., Sutskever I., Hinton G. E., 2012, Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, 60, 1097
LSST Dark Energy Science Collaboration 2012, arXiv preprint arXiv:1211.0310, p. 133
Lanusse F., Ma Q., Li N., Collett T. E., Li C.-L., Ravanbakhsh S., Mandelbaum R., Póczos B., 2018, Monthly Notices of the Royal Astronomical Society, 473, 3895
LeCun Y., Boser B., Denker J. S., Henderson D., Howard R. E., Hubbard W., Jackel L. D., 1989, Neural Computation, 1, 541
Leitner S. N., Kravtsov A. V., 2011, ApJ, 734, 48
Lilly S. J., Carollo C. M., Pipino A., Renzini A., Peng Y., 2013, ApJ, 772, 119
Mannucci F., Cresci G., Maiolino R., Marconi A., Gnerucci A.,

2010, Monthly Notices of the Royal Astronomical Society, 408, 2115
Molino A., et al., 2017, Monthly Notices of the Royal Astronomical Society, 470, 95
Ntampaka M., Trac H., Sutherland D. J., Battaglia N., Póczos B., Schneider J., 2015, The Astrophysical Journal, 803, 50
Ntampaka M., Trac H., Sutherland D. J., Fromenteau S., Póczos B., Schneider J., 2016, The Astrophysical Journal, 831, 135
Oke J. B., 1974, The Astrophysical Journal Supplement Series, 27, 21
Pedregosa F., et al., 2012, Journal of Machine Learning Research, 12, 2825
Perez F., Granger B. E., 2007, Computing in Science & Engineering, 9, 21
Petrillo C. E., et al., 2017, Monthly Notices of the Royal Astronomical Society, 472, 1129
Petrillo C. E., et al., 2018, eprint arXiv:1807.04764
Salim S., et al., 2007, The Astrophysical Journal Supplement Series, 173, 267
Sánchez S. F., et al., 2013, A&A, 554, A58
Simmons B. D., et al., 2017, MNRAS, 464, 4420
Simonyan K., Zisserman A., 2014, preprint, (arXiv:1409.1556)
Smirnov E. A., Markov A. B., 2017, Monthly Notices of the Royal Astronomical Society, 469, 2024
The Dark Energy Survey Collaboration 2005, eprint arXiv:astro-ph/0510346, p. 42
Tremonti C. A., et al., 2004, The Astrophysical Journal, 613, 898
Xu X., Ho S., Trac H., Schneider J., Poczos B., Ntampaka M., 2013, The Astrophysical Journal, 772, 147
York D. G., et al., 2000, The Astronomical Journal, 120, 1579
Zeiler M. D., Fergus R., 2014, in Fleet D., Pajdla T., Schiele B., Tuytelaars T., eds, Computer Vision – ECCV 2014. Springer, Cham, Cham, pp 818–833, doi:10.1007/978-3-319-10590-1_53, http://link.springer.com/10.1007/978-3-319-10590-1{_}53

APPENDIX A: RESIDUAL CONVOLUTIONAL NEURAL NETWORKS

We find that a 34-layer resnet (Resnet-34) architecture can be trained efficiently on a Pascal P100 with 16 GB of memory. Using the hyperparameters described below, an epoch takes about 60 seconds to train. We initialize our resnet with pretrained weights from the ImageNet (Russakovsky et al. 2014; He et al. 2015) 1.7 million image data set trained to recognize 1000 classes of objects found on Earth (i.e., cats, dogs, or cars). In practice, the filters learned through the early layers of the pretrained CNN can be used for other image recognition tasks, aptly named "transfer learning" (cite).

A1 Hyperparameter selection

A2 Data augmentation

Nearly all neural networks benefit from larger training samples because they help prevent overfitting. Outside of the local Universe, galaxies are seen at nearly random orientation; such invariance permits synthetic data to be generated from rotations and flips of the training images (see, e.g., Simonyan & Zisserman 2014). Each image is fed into the network along with four augmented versions, thus increasing the total training sample by a factor of five. This technique is called data augmentation, and is particularly helpful for the network to learn uncommon or unrepresented truth values (e.g., in our case, very metal-poor or metal-rich galaxies). Each training-augmentation is fed-forward through the network and gradient contributions are computed together as part of the same mini-batch. A similar process is applied to the network during predictions, which is called test-time augmentation (TTA). Synthetic images are generated according to the same rules applied to the training data set. The CNN predicts an ensemble average over the augmented images. It has been found that data augmentation improves predictions by as much as $\sim 20\%$ (cite).

A3 Cyclical learning rate annealing

The learning rate determines how large of a step each network weight takes in the direction of the backpropagated error. A large learning rate thus forces the weights to make large updates, which generally prevents overfitting but also may cause the parameters to overshoot minima in the loss function landscape. A small learning rate may allow the weights to get stuck in a local minima near their initial positions, or also might cause the network to learn very slowly. The number of local minima increases exponentially with dimensionality (cite), so selecting a low learning rate does not ensure that the network will eventually make it to near the global minimum. Therefore, it is often useful to anneal, or reduce, the learning rate from a high value in the beginning – which allows the weights to find the right ballpark values – to a low value – which allows the weights to take finer steps near the global minimum – over the course of multiple training epochs.

In practice, annealing can be enforced manually, e.g., the learning rate might be reduced by a factor of 10 every time the loss function plateaus over a number of epochs. Eventually even reduced learning rates do not permit additional improvement of the loss, and so the training period is concluded. We instead use a method called cosine annealing, during which the learning rate is annealed continuously over one or more epochs for *each* mini-batch until it eventually reaches zero.

We employ stochastic gradient descent with restarts (SGDR), over progressively longer training cycles, and restart cosine annealing over each cycle (Leslie Smith 2015?). For example, the first cycle comprises one epoch during which the learning rate is cosine annealed. It then trains for another cycle with cosine annealing, which starts at the same learning rate but is annealed over twice the duration (two epochs). These "restarts" have been shown to kick the weights configuration out of saddle points in the loss of some high-dimensional parameter space. Training can then proceed past what appear to be local minima but are actually saddle points (where gradient descent generally performs poorly).

A4 Layer learning rates

ADD MORE HERE

Frozen training to get final activates in right "ballpark." We then unfreeze all layers and train using different rates in different layer groups.

A5 Batch normalization and Dropout

Batch normalization (BN; 1502.03167) is a technique developed to fix the vanishing gradients problem which make deep networks inefficiently slow to train. The issue arises when gradients are backpropagated through deep neural networks to update weights, and loss contributions become vanishingly small except only when the weights have small magnitudes. Normalizing the inputs to each mini-batch mean and standard deviation somewhat remedies the problem, and has been applied to (convolutional) neural networks since the 1990s. BN extends this normalization to all activations in hidden layers, thus adding two hyperparameters which modulate the mean and standard deviation of each layer to zero and unity, respectively. The BN hyperparameters are learned for each mini-batch and are updated in addition to weight parameters during the backward pass.

Without BN, activations in any given layer may span a large range and contributions from certain parts of the network may become dwarfed by others. After the normalization step, all pre-activations are on the same scale and thus gradient descent steps can more efficiently traverse the loss function space. Training proceeds more rapidly and converges toward a solution more quickly. Choice of mini-batch size also impacts the learning rate, as small batch size increases stochasticity in each gradient. Large batch size allows for better parallelization on the GPU and ensures smoother gradients. We note that smooth gradients across mini-batches can prevent the solution from hopping out of local basins, and negatively impact performance. Increasing the learning rate as batch size is increased can resolve this problem, at least in part, but limits the convergence ability of the network (until the rate is annealed). We find in practice that batch sizes between 128 and 512 work best at balancing noisy gradients and learning rate.

Dropout is a method of disabling a random subset of connections after linear layers for each mini-batch (Hinton et al. 2012). By removing random connections between fully-connected layers, dropout effectively treats each mini-batch as one in an ensemble of random training subsets . It is instructive to think of each mini-batch in the full training data as analogous to a decision tree in a random forest. The ensemble of learned gradients is less prone to overfit the training data set because the network is forced to discard random (and potentially valuable) information. The resulting network is better able to, for example, learn subtle differences in the data that would otherwise be ignored when more obvious features dominate the gradient descent process. Since our Resnet architecture is broadly separated into multiple groups of layers, we can apply lower dropout rate at earlier layers in order to ensure that those filters are learned more quickly. Using a differential dropout rate is sensible because filters learned in the first few layers of the network tend to be Gabor filters and edges in general, and there is little risk of overfitting when such low-level abstractions are always necessary for learning high-level features in the middle and final layers.

During validation and testing, dropout is not used because all of the data is useful for predictive power. We use dropout rates of 0.25 for the linear layer after the early group, and 0.50 at the later linear layer. We find that dropout combined with BN – although not recommended in the original paper – work in tandem to boost training speeds and avoid overfitting.

This paper has been typeset from a TEX/LATEX file prepared by the author.