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

Things we are going to want to talk about in the introduction: Morphology-metalicity relation? There is certainly a mass-metallicity relation. Basically, we want to say why we think we can do this at all? What is the primary science driver behind why we think this will work? Is it really just the fundamental plane? We aren't telling the thing about the mass or the distance to the objects.

Spectral super-resolution?

Large-area sky surveys, both on-going and planned, are revolutionizing our understanding of galaxy evolution. The on going Dark Energy Survey (DES; **?**) and planned Large Synoptic Survey Telescope (LSST; **?**) will survey vast swaths of the sky and create samples of galaxies much larger than any previously know. Spectroscopic follow-up will be key to a deep understanding of the properties of these galaxies, and constrain galaxy evolution (e.g., the mass-metallicity relation, **?**; or the fundamental metallicity relation, **??**). But as the dataset continues to grow individual follow-up becomes increasingly impractical. Therefore, large spectroscopic surveys are needed to more fully understand the observable-mass relation of clusters. In this work, we propose to use supervised ML, specifically convolution neural networks, to analyze pseudo-three color images to predict galaxy metallicity. This paper is organized as follows: Section **??** we briefly introduce convolutional neural networks, discuss selection of the network's hyperparameters and outline training the network. In Section 3, we describe the acquisition and cleaning of the SDSS data sample. We present the main results in Section 4 and discuss the results in the context of previous works in Section **??**. In Section 7, we summarize the key results and conclude. 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 (**?**), and use AB magnitudes (**?**).

2 METHODOLOGY

2.1 Convolutional neural networks

In recent years, neural networks have been able to accomplish a large number of tasks in the field of machine learning (LeCun et al. 1989). Image classification and regression problems are most readily solved by use of convolutions in mutliple 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; 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 2013).

The input layer is simply an image of 128 × 128 pixels with three channels (RGB). Dieleman et al. (Galaxy Zoo Kaggle competition) have already shown that CNNs are capable of classifying the morphologies of such images of galaxies from the Sloan Digital Sky Survey (SDSS) with the same accuracy as citizen scientists. Other people have been done other cools things (like use simulated images to select real galaxies, etc). blah blah blah. Define overfitting: a specific and usually not general set of features are learned and misapplied to the test set data.

2.2 Training in practice

We split our training sample of $\sim 130,000$ images into sets of $90,000$ for training, $20,000$ for validation, and $20,000$ for testing. Training images are seen once every 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 – another technique used to prevent overfitting. Each mini-batch is fed forward through the input layers, where a random fraction of connections $p = 0.25, 0.50$ are removed between each linear layer (dropout, Hinton et al. 2012; see subsection below). When the feed-forward network reports a prediction, whether a single or set of quantities, 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. We use gradient descent for each minibatch to adjust each weight parameter, and each fractional contribution of loss is determined by the backpropagation algorithm (cite original, LeCun et al.). 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*; batch-normalization is also applied in addition to a momentum term which ensures that the gradient is itself only changing slowly with each minibatch.

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 further layers simply noisily propagate the same signal and degrade the loss. We select an architecture called a residual neural network, or a *resnet* (He et al. 2015), which contains enhanced "shortcut connections" but are otherwise similar to other CNNs. Resnets have been shown to continue learning with increasing depth without the added cost of extra parameters.

We use a 34-layer resnet (He et al.), 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 (and "freeze" 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 the Appendix.

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 Appendix) 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:

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

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

Glossary of terminology:

- $Z \equiv 12 + \log(\mathrm{O/H})$ is the nebular phase metallicity
- Z_{true} is the spectroscopically derived metallicity
- Z_{pred} is the CNN predicted metallicity
- $\Delta Z \equiv Z_{\mathrm{pred}} - Z_{\mathrm{true}}$ is the residual, or error
- RMSE $\equiv \sqrt{\langle|\hat{y}-y|^2\rangle}$ is the root-mean squared error. The RMSE loss function is minimized during CNN training.
- NMAD $\approx 1.4826 \times \mathrm{median}\big(|x - \mathrm{median}(x)|\big)$ is the normal median absolute deviation

3 DATA

3.1 Sample selection

Our sample of galaxies are selected from the NYU Value Added Catalog (VAC) cite, to have gas phase metallicities. The 50th percentile metallicity estimates (**?**) are derived from spectroscopic measurements and for the purposes of this work are assumed to be "the truth." In addition, we take the derived stellar mass for each galaxy. We supplement the data from the VAC with data from SDSS Data Release 13 (DR13; **?**) for each galaxy, the right ascension and declination and the magnitude in each of the five SDSS photometric bands (u, g, r, i, z), along with associated errors. We require that galaxies magnitudes are $10 < ugriz < 25$ mag. Galaxies should have colors $0 < u - r < 6$ why?, high confidence ($z_{err} < 0.01$) spectroscopic redshifts greater than 0.2, check the petro mags and why we are using those.

3.2 SDSS Images

From this input catalog we create RGB images 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 $0''.396$ SDSS pixel size are rescaled to $0''.296$ per pixel. We obtain 142,176 three color images to use as the full data set. We split this data set into random 60%, 20%, and 20% subsets, which comprise the training, validation, and test data sets respectively.

3.3 Metallicity

4 RESULTS

4.1 Example predictions

Examples of the 128×128 pixel *gri* SDSS images fed into the the CNN are shown in Figure 1. Rows (a) and (b) depict the galaxies with lowest predicted and lowest true metallicities, respectively. The CNN has associated blue, disky galaxies with low metallicities, and is generally accurate for those

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

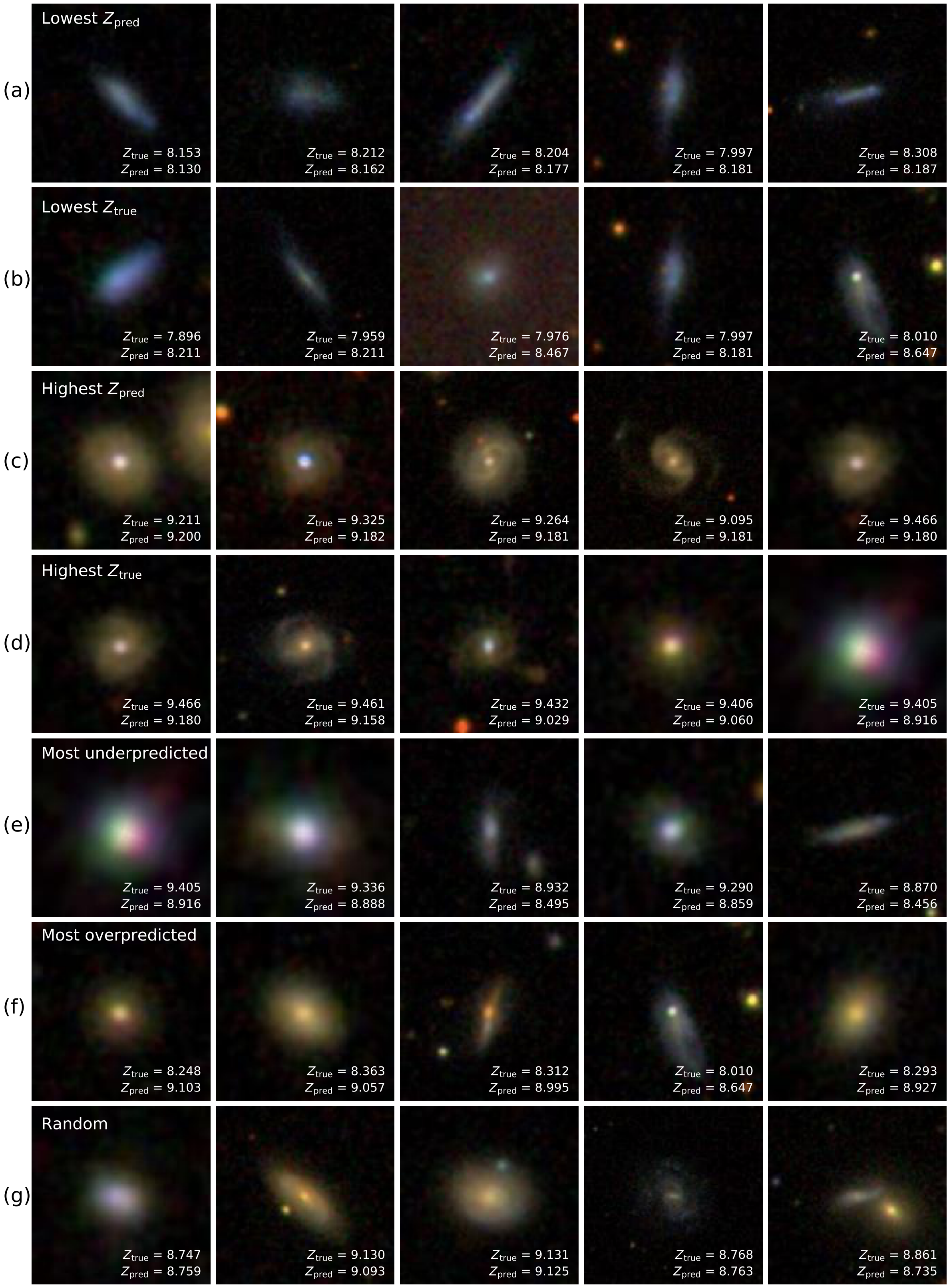
(a)
Lowest Z_{pred}
Z_{true} = 8.153
Z_{pred} = 8.130
Z_{true} = 8.212
Z_{pred} = 8.162
Z_{true} = 8.204
Z_{pred} = 8.177
Z_{true} = 7.997
Z_{pred} = 8.181
Z_{true} = 8.308
Z_{pred} = 8.187
(b)
Lowest Z_{true}
Z_{true} = 7.896
Z_{pred} = 8.211
Z_{true} = 7.959
Z_{pred} = 8.211
Z_{true} = 7.976
Z_{pred} = 8.467
Z_{true} = 7.997
Z_{pred} = 8.181
Z_{true} = 8.010
Z_{pred} = 8.647
(c)
Highest Z_{pred}
Z_{true} = 9.211
Z_{pred} = 9.200
Z_{true} = 9.325
Z_{pred} = 9.182
Z_{true} = 9.264
Z_{pred} = 9.181
Z_{true} = 9.095
Z_{pred} = 9.181
Z_{true} = 9.466
Z_{pred} = 9.180
(d)
Highest Z_{true}
Z_{true} = 9.466
Z_{pred} = 9.180
Z_{true} = 9.461
Z_{pred} = 9.158
Z_{true} = 9.432
Z_{pred} = 9.029
Z_{true} = 9.406
Z_{pred} = 9.060
Z_{true} = 9.405
Z_{pred} = 8.916
(e)
Most underpredicted
Z_{true} = 9.405
Z_{pred} = 8.916
Z_{true} = 9.336
Z_{pred} = 8.888
Z_{true} = 8.932
Z_{pred} = 8.495
Z_{true} = 9.290
Z_{pred} = 8.859
Z_{true} = 8.870
Z_{pred} = 8.456
(f)
Most overpredicted
Z_{true} = 8.248
Z_{pred} = 9.103
Z_{true} = 8.363
Z_{pred} = 9.057
Z_{true} = 8.312
Z_{pred} = 8.995
Z_{true} = 8.010
Z_{pred} = 8.647
Z_{true} = 8.293
Z_{pred} = 8.927
(g)
Random
Z_{true} = 8.747
Z_{pred} = 8.759
Z_{true} = 9.130
Z_{pred} = 9.093
Z_{true} = 9.131
Z_{pred} = 9.125
Z_{true} = 8.768
Z_{pred} = 8.763
Z_{true} = 8.861
Z_{pred} = 8.735

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 negative $\Delta Z \equiv Z_{\text{pred}} - Z_{\text{true}}$, (f) most positive ΔZ, and (g) randomly selected galaxies.

which it has identified to be 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 bulges or bright nuclei are predicted to be high in metallicity, and their predictions generally match $Z_{\rm true}$. The main result appears to be that galaxies predicted by our CNN to have high metallicity ($Z_{\rm pred} > 9.0$) tend to truly have high metallicity, and the equivalent for low-metallicity galaxies. Inversely, galaxies with the *highest* or *lowest* $Z_{\rm true}$ in the sample usually also yield high or low respective predicted metallicities. Note that inclined galaxies tend to be lower in metallicity whereas face-on galaxies appear to be higher in metallicity. 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 outer regions to depress the metallicity.

We will now consider examples of the most incorrectly predicted galaxies. In rows (e) and (f) respectively, we show instances in which the CNN predicted too low metallicity and too high metallicity. The two galaxies with lowest residuals $\Delta Z \equiv Z_{\rm pred} - Z_{\rm true}$ (i.e., most underpredicted metallicities) suffer from artifacts which cause unphysical color gradients.[2] Otherwise, the mistakes made by the CNN are similar to ones that human intuition are prone to make: 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. The histogram bin sizes were chosen according to the Freedman & Diaconis (1981) rule for each distribution. The discreet striping of the Tremonti et al. and Brinchmann et al. metallicity estimator appears in the $Z_{\rm true}$ distribution (shown in black) but does not manifest in our CNN predictions (shown in red). Because the true metallicities are discreetly valued, we expect that our distribution of ΔZ will be heavy-tailed, since predicted values still be distributed smoothly over the peaks and troughs in the $Z_{\rm true}$ distribution. We also expect some outlier predictions at very negative or positive ΔZ for other reasons.

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 never predict very high or very low metallicities. Although the two distributions are qualitatively consistent with each other at low metallicities (e.g., $Z < 8.5$). However, the fraction of galaxies with high $Z_{\rm true} > 9.1$ ($2573/20466 = 12.6\%$) is more abundant than the fraction with high $Z_{\rm pred} > 9.1$ ($1174/20466 = 5.7\%$).

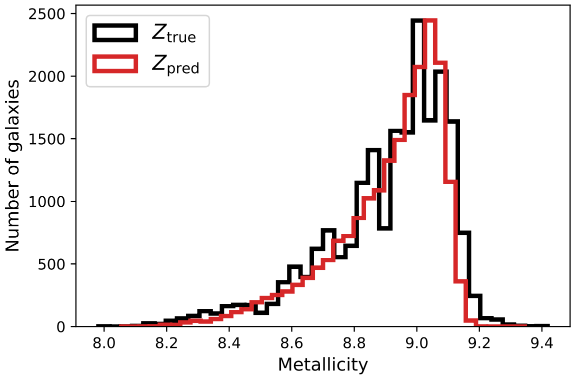

Figure 2. Distributions of the true (black) and predicted (red) galaxy metallicities. A distribution of residuals is shown in the inset panel. Note that the bin widths are different for each distribution.

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). Overall, the running median agrees well with the one-to-one line, although at low metallicity we find that the CNN makes makes overpredictions. Thus, even though $Z_{\rm pred}$ and $Z_{\rm true}$ are in agreement at $Z < 8.5$ in Figure 2, we now find that the low-metallicity predictions are systematically too high.

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 overprediction of low-$Z_{\rm true}$ galaxies.

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 an average 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 MAD determines the scatter about the *median* and not $\Delta Z = 0$, and thus, this metric be-

[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.

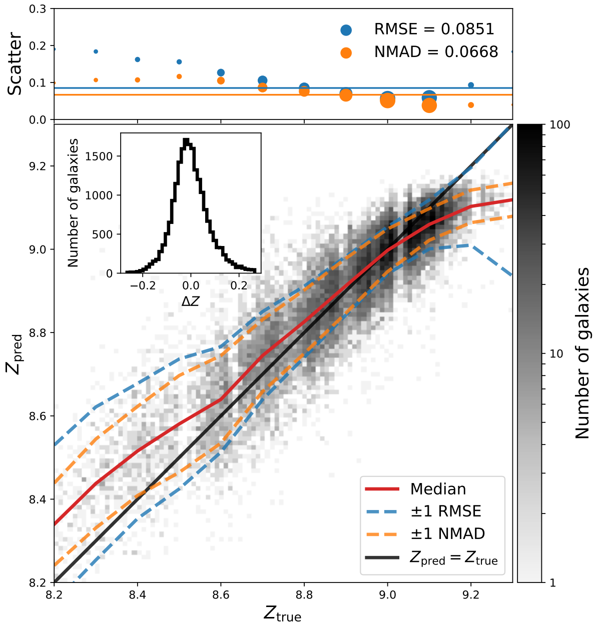

Figure 3. Bivariate distribution of metallicity truths ($Z_{\rm true}$) and CNN predictions ($Z_{\rm pred}$) are shown in the main panel. Overlaid are the predicted median metallicity (*solid red line*), RMSE scatter (*dashed blue line*), NMAD scatter (*dashed orange line*) in bins of $Z_{\rm true}$, and a one-to-one relation (*solid black line*). 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).

comes 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, 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). Another reason why the global NMAD exceeds the binned NMAD average is that each binned NMAD is computed using its local scatter, and this allows for outlier rejection using a standard which varies with $Z_{\rm true}$. We can demonstrate this result using an example: $\Delta Z \approx 0.2$ would be treated as an 3 σ 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 consider the effects of image resolution by running the exact same CNN training and test procedure on the same images scaled to different sizes. Our initial results use SDSS 15″ × ″ cutouts resized to 128 × 128 pixels, and we now downsample the images to 64 × 64, 32 × 32, $\cdots$, 2 × 2, and even 1 × 1 pixels. All images retain their three channels, so the smallest 1 × 1 image effectively contains three colors of the image (averaged together with its background and possible neighboring sources).

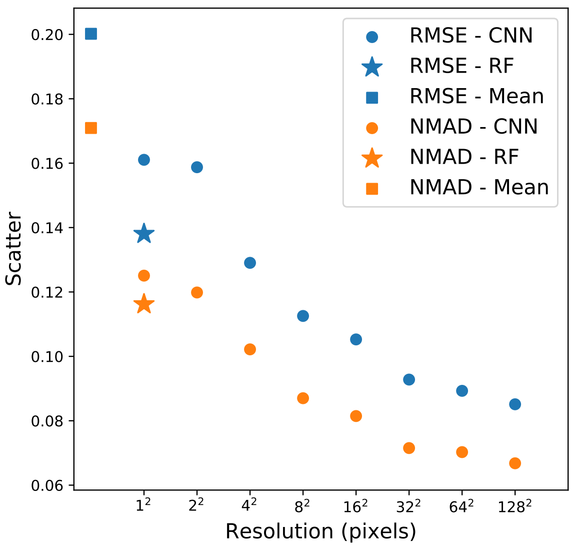

Figure 4. The effects of image resolution on CNN performance. Blue and orange circular markers indicate scatter in the residual distribution (ΔZ) measured using RMSE and NMAD, respectively. (Each point is analogous to the horizontal lines shown in Figure 3.) We also show predictions from a random forest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

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, and labeled “Mean”). This constant prediction is effectively the worst-possible scatter, and the SDSS systematic uncertainty in $Z_{\rm true}$ of ~ 0.03 dex is the best-possible scatter. We find that both RMSE and NMAD decrease with increasing resolution. 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.

There appears to be little improvement in scatter going from 1 × 1 to 2 × 2 pixel images, although in hindsight such a finding should not be a surprise. 1 × 1 three-color images contain similar information to three photometric data points (although because of the averaged background and neighboring pixels, it is less useful than photometry), and three photometric data points can be used to perform a crude spectral energy distribution (SED) fit. The color information is useful enough to create an approximate color-magnitude diagram, and each galaxy’s position in this parameter space correlates with $Z_{\rm true}$. The 2 × 2 three-color images contain four times as many pixels as the 1 × 1 images. However, because there is an even number of pixels, this information is still averaged between all available pixels, as so the incremental gain is small![3]

4.5 Random forest predictions

We also construct a random forest (RF) of decision trees in order to predict metallicity, using the implementation from `scikit-learn` (cite). 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 had done 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 expected, given that the RF is supplied aperture-corrected photometry, whereas the CNN is given 1×1 *gri* "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.

4.6 Comparisons to Previous Works

CNNs have been used for a wide variety of classification tasks in extragalactic astronomy, including morphological classification (see, 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, e.g., visual morphologies. Our paper uses CNNs to tackle the different problem of regression, i.e., predict values from a continuous distribution.

We have also found examples of regressing stellar properties in the astronomical machine learning literature (see, e.g., Bailer-Jones 2000; Fabbro et al. 2018); they train on synthetic spectra and test on real data. However, their true values of, e.g., stellar effective temperature, surface gravity, or elemental abundance, are known to be encapsulated in the synthetic stellar spectra. Our work is novel because we predict metallicity, a spectroscopically determined galaxy property, only using three-color images — and public-facing JPG images at that. We have found that galaxy morphologies and colors contain information useful for predicting metallicity.

Acquaviva (2016) have used 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 data set used in this work 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 features. 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.

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 when using RFs on the five-band photometry. Using the same approach on a larger, unified, and different data set, which — most importantly — only contains *three* bands of photometric information, we find RMSE = 0.1296. 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 the $Z_{\rm true}$ distribution is narrower than for our data set, or the fact that we do not control for galaxy redshift; however, some of this advantage is offset by our larger sample size. Ultimately, it appears the extra u and z bands provide valuable information, which allows for better estimation of metallicity. RFs using the added photometry data output $Z_{\rm pred}$ with significantly lower scatter than our using only *gri* bands.

Indeed, the u and z bands convey information about the star formation rate (SFR) and stellar mass ($M_\star$). 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 given $M_\star$ *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.

5 THE MASS-METALLICITY RELATION

The mass-metallicity relation (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 mass range $8.5 < \log(M_*/M_\odot) < 11.5$ (Tremonti et al. 2004), where σ is the standard deviation of the metallicity and is equal to the RMSE for a normal distribution. They characterize the MZR using a polynomial fit:

$$Z = -1.492 + 1.847 \log(M_*/M_\odot) - 0.08026\,[\log(M_*/M_\odot)]^2\,. \tag{2}$$

The physical interpretation of the MZR is that a galaxy's mass strongly influences its chemical enrichment. The origin of such a relationship is not clear, and proposed explanations 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

[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.

also Sánchez et al. (2013). Although the exact physical process responsible for tight, 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 ISM and stellar mass assembly. The FMR connects the instantaneous ($\sim$ 10 Myr) SFR with the gas-phase metallicity ($\sim$ 1 Gyr timescales; see, e.g., Leitner & Kravtsov 2011) and M_* (i.e., the $\sim$ 13 Gyr integrated SFR). Our CNN is better suited for predicting M_* rather than SFR, using the *gri* bands, which can only weakly probe the bluer light from young, massive stars. Therefore, we expect scatter from 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 the color and morphology, in tandem with CNN-predicted stellar mass, can be used to roughly estimate the SFR, although any such relationship is not within the scope of our paper.

5.1 Predicting stellar mass

We re-run the CNN methodology to train and predict M_* 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_*), we verify that $M_{*,\rm true}$ median agrees with the median of $M_{*,\rm true}$ for values between $9.0 \lesssim \log M_* \lesssim 10.5$. The RMSE scatter in the M_* residuals is $\sim$ 0.22 dex, and the NMAD is $\sim$ 0.20 dex. The slope of the empirical MZR at $\log(M_*/M_\odot) \sim 10$ is (0.4 dex in Z)/(1.0 dex in M_*), implying that the CNN might be able to leverage the MZR and predict metallicity to $\sim$ 0.08 dex (plus any intrinsic scatter in the MZR, in quadrature). Given our CNN's ability to accurately predict M_*, coupled with the inability of SDSS images to inform it about nebular regions or spectral lines, is it possible that the CNN's unexpectedly strong performance in predicting metallicity comes through the MZR?

One option is that the CNN learned to predict M_* followed by the polynomial MZR transformation (Equation 2). We can simulate predicting metallicity this way by outputting $M_{*,\rm pred}$, and then passing those predictions through the empirical MZR (which we will call $Z_{\rm MZR}$). We show $Z_{\rm MZR}$ against $Z_{\rm true}$ in the right panel of Figure 5. 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. Intriguingly, however, the scatter is slightly lower than the 0.13 dex $\approx \sqrt{(0.10~\rm dex)^2 + (0.08~\rm dex)^2}$ expected from independently adding the MZR scatter and M_* prediction scatter in quadrature. **This evidence suggests that the CNN has learned to predict metallicity in a more powerful way than the MZR.**

5.2 A tighter MZR?

In Figure 6, we show true stellar mass ($M_{*,\rm ~true}$) against CNN-predicted metallicity. For comparison, we also overlay the Tremonti et al. (2004) MZR and its scatter (σ = 0.10 dex). The empirical median relations matches our predicted MZR median, and the lines marking different measures of scatter appear to match each other as well. For $9.5 \leq \log(M_{*,true}/M_\odot) \leq 10.5$, the RMSE scatter in $Z_{\rm pred}$, shown in dashed blue, appears to be even tighter than the measured $\pm 1~\sigma$ (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, which confirms that our CNN predicts a MZR that is at most equal (and possibly smaller) in scatter than one constructed using the true metallicity. The mass bins for which predicted RMSE is lower than measured σ are the ones which contain the most training examples. Thus, it may be possible that if our data set was augmented to include additional low- and high-$M_{*,\rm true}$ galaxies, then the predicted RMSE (and NMAD) would be even lower.

The fact that a CNN trained on only *gri* 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 0.03 dex systematic scatter in $Z_{\rm true}$ measurements leaves only $\sim$ 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 $Z_{\rm pred} - M_{*,\rm true}$ relation and the empirical MZR ($Z_{\rm true} - M_{*,\rm true}$) without accounting for the fact that $Z_{\rm pred}$ and $Z_{\rm true}$ differ by RMSE = 0.085 dex! We thus find more evidence that the CNN has learned something from the SDSS *gri* imaging that is different from, but at least as powerful as, the MZR.**

6 FUTURE APPLICATIONS

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.

7 SUMMARY

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.

Our main conclusions are the following:

(i) 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.

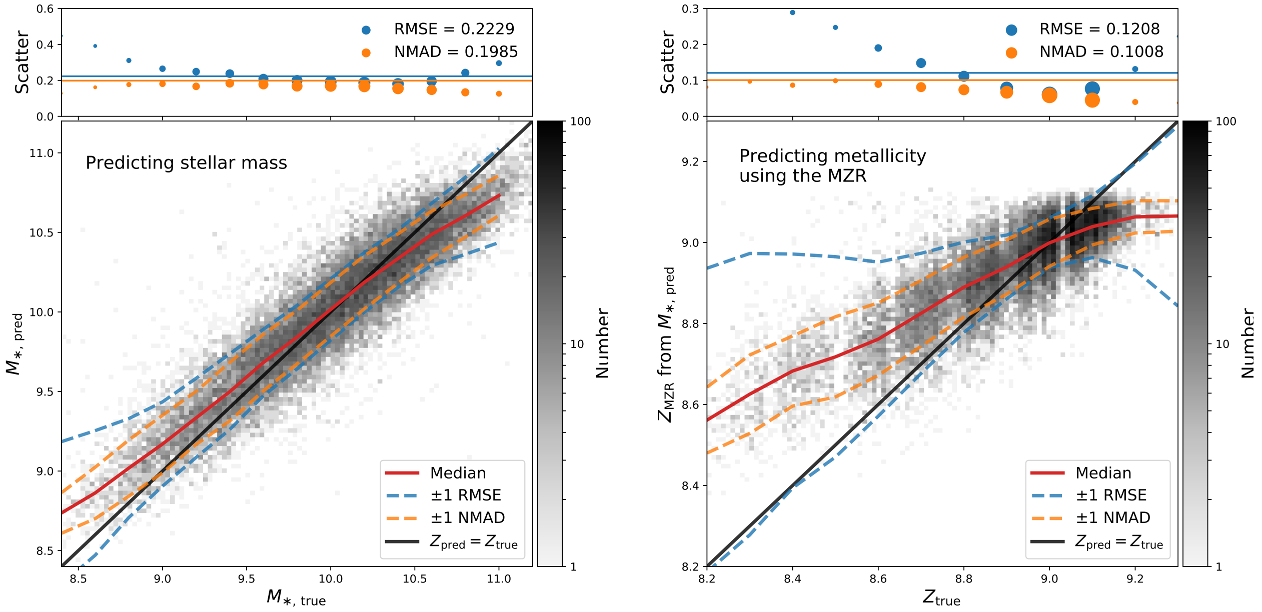

Figure 5. In the left subplot, we show predicted against true stellar mass. Colors and marker or line styles are the same as in Figure 3. If the right subplot, we compare the predicted stellar mass converted to metallicity, assuming the Tremonti et al. mass-metallicity relation (MZR), with the true metallicity.

(ii) 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.

ACKNOWLEDGEMENTS

The authors also wish to thank the anonymous referee whose comments and suggestions significantly improved both the quality and clarity of this work. The authors 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 (?) and matplotlib, a Python library for publication quality graphics (?). 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

Acquaviva V., 2016, MNRAS, 456, 1618
Bailer-Jones C. A. L., 2000, A&A, 357, 197
Dai J.-M., Tong J., 2018, preprint, (arXiv:1807.05657)
Dieleman S., Willett K. W., Dambre J., 2015, MNRAS, 450, 1441
Fabbro S., Venn K. A., O'Briain T., Bialek S., Kielty C. L., Jahandar F., Monty S., 2018, MNRAS, 475, 2978
Freedman D., Diaconis P., 1981, Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete, 57, 453
Garnett D. R., 2002, ApJ, 581, 1019
Huertas-Company M., et al., 2015, ApJS, 221, 8
Huertas-Company M., et al., 2018, ApJ, 858, 114
Kim E. J., Brunner R. J., 2017, MNRAS, 464, 4463
Lanusse F., Ma Q., Li N., Collett T. E., Li C.-L., Ravanbakhsh S., Mandelbaum R., Póczos B., 2018, MNRAS, 473, 3895
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
Sánchez S. F., et al., 2013, A&A, 554, A58
Simmons B. D., et al., 2017, MNRAS, 464, 4420
Tremonti C. A., et al., 2004, ApJ, 613, 898

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).

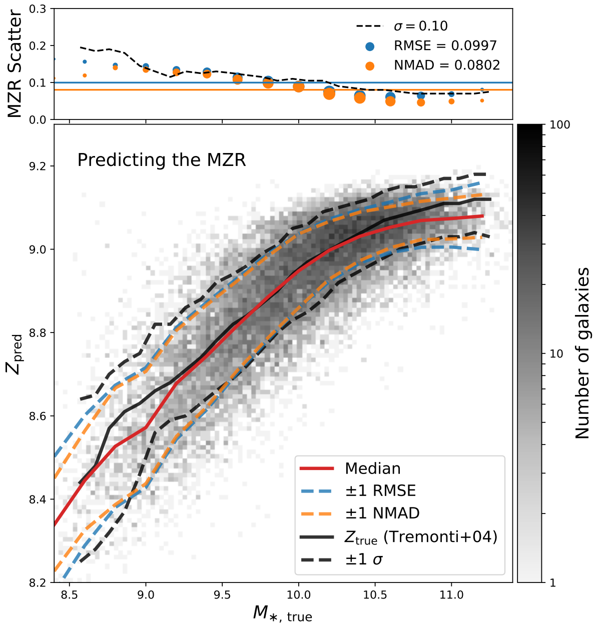

Figure 6. In the main panel, the predicted MZR comparing true M_* against CNN predicted $Z_{\rm pred}$ is shown in grayscale. The running median (solid red) and scatter (dashed blue and orange) are shown in 0.2 dex mass bins. For comparison, we also show the Tremonti et al. (2004) observed median and scatter binned by 0.1 dex in mass (solid and dashed black lines, respectively). In the panel above the main figure, 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_* (and whose marker sizes are proportional to the number of galaxies in the mass bin). Global scatter in the predicted MZR appears to be comparable or even lower than scatter from the true MZR.

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. 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.