Statistical techniques for predicting galaxy properties: SED fitting and Machine Learning

Cameron Morgan, University of Waterloo
PHYS788 - Statistical Tools for Astronomers

April 22, 2024

1 Introduction

As a test of statistical tools, we use spectral energy distribution (SED) fitting to predict properties of galaxies such as stellar mass and redshift. Using SED fits from PROSPECTOR (Johnson et al., 2021), we explore the meaning of χ^2 statistics, covariance and correlation matrices, and jackknife sampling. We then explore the Bayesian sampling techniques employed by SED fitting codes outside of PROSPECTOR itself. Finally, we test machine learning (ML) tools that can predict galaxy parameters with bypassing SED fitting entirely. These techniques are compared where applicable, with their pros and cons discussed. Finally, we provide some context of using these techniques in real research.

2 Initial SED fit and covariance of constrained parameters

While it would be ideal to test SED fitting techniques on a sample of galaxies, we choose one galaxy ("gal44") to carry out all tests on since SED fits can be computationally expensive. We initially set up a PROSPECTOR run using a delayed-τ star formation history model ($SFR \sim t_{\rm age} e^{t_{\rm age}/\tau}$ where $t_{\rm age}$ is the age of the galaxy and τ is the exponential decay factor). The simple model involves five free parameters: stellar mass, metallicity, dust opacity, t_{age} and τ. Additionally, we fit another model where we include redshift has a free parameter. To get a sense of the goodness-of-fit of our model to our data, we calculate the reduced chi-squared (χ^2_ν) statistic on each model. The (unreduced) χ^2 is calculated as follows:

$$\chi^2 = \sum_i \frac{(y_i - f(x_i))^2}{\sigma_i^2} \tag{1}$$

where y_i are the observed values (in this case, specific flux), $f(x_i)$ are the model-predicted values, and σ_i are the observed uncertainties. To turn a χ^2 into χ^2_ν, we divide by the number of degrees of freedom ($df = N - k$, where N is the number of data points and k is the number of free parameters). A good fit expects $\chi_\nu \sim 1$, but we find $\chi_\nu \gg 1$ for both our models, indicating that the model is a poor fit to the data. Tweaking the model parameters could produce a better fit for this given galaxy. This poor fit is unsurprising just by looking at the SED in Fig. 1. Additionally, the corner plot in Fig. 2 shows that the posterior distribution of many of the parameters is not well described by a Gaussian or any similar function.

To explore the covariance and correlation between the model parameters, we produce a covariance and correlation matrix. While both these provide an indication of the relationship between parameters, the correlation matrix is more informative in this case. This is because the dynamic range of each of the parameters are quite different from one another, skewing the covariance values. Instead, the correlation matrix will always provide correlation values between -1 and 1. We see from Fig. 3 that the diagonal values are always 1. This indicates perfect positive correlation between a parameter and itself, as expected. We note high positive correlation values between mass and redshift, as well as between $t_{\rm age}$ and τ. This can also be seen in the corner plot of Fig. 2.

We also explore jackknife resampling as a means to determine covariance and compare to analytic uncertainties. Jackknifing involves splitting a sample into n bins, and iterating through the sample n times, each time removing one bin, and calculating a statistic. In this case, we take a sample of $\sim 800,000$ galaxies, binned by stellar mass. The analytic error bars on each mass bin are represented by Poisson statistics, such that $\sigma_i = \sqrt{N_i}$. We perform 200 jackknife iterations, re-binning each sample and calculating the covariance and correlation between the mass bins. Fig. 4 shows that

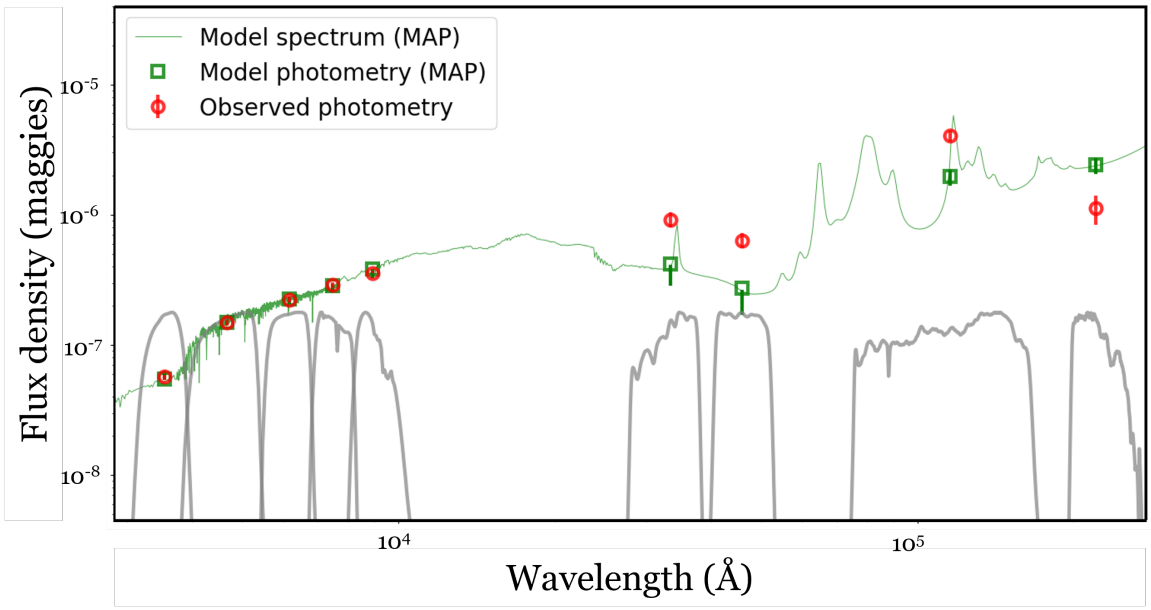

Figure 1: An example SED fit of a galaxy. The grey bands show the filter transmission curves for five optical and four infrared bands. The red points and errorbars show the observed photometry of the galaxy in those bands. The green points and errorbars show the best fit model photometry after a PROSPECTOR run, and the green line shows the inferred spectrum based on this model.

some mass bins are somewhat correlated. We find that the uncertainties determined using the jackknife covariance are slightly larger than the analytic errors.

3 Bayesian Sampling Techniques

PROSPECTOR can employ a number of Bayesian sampling techniques for SED fitting. Here we explore two of these outside of PROSPECTOR itself. Bayesian sampling techniques are based on conditional probabilities following Bayes theorem:

$$P(\Theta|\text{data}) = \frac{P(\text{data}|\Theta)P(\Theta)}{P(\text{data})} \tag{2}$$

where $P(\text{data}|\Theta)$ is the likelihood of the data given the model parameters, $P(\Theta)$ is the prior probability which captures previous known bounds on the model parameters, $P(\text{data})$ is the evidence or the probability of the data independent of the model, and $P(\Theta|\text{data})$ is the posterior probability of the model fitting the data.

3.1 Markov Chain Monte Carlo: emcee

The Markov Chain Monte Carlo (MCMC) sampling technique begins with some initial guess of the best-fit parameters for a model. The prior and likelihood probabilities are then computed at this initial step. Next, a second point is chosen based off some algorithm (the discussion of which is beyond the scope of this paper), with the new position being accepted or rejected with some probability. This process repeats some number of times, forming a Markov Chain. A number of Markov chains are employed simultaneously to explore the posterior probability space for each parameter. An example of 32 chains taking 5000 steps each to explore the redshift parameter

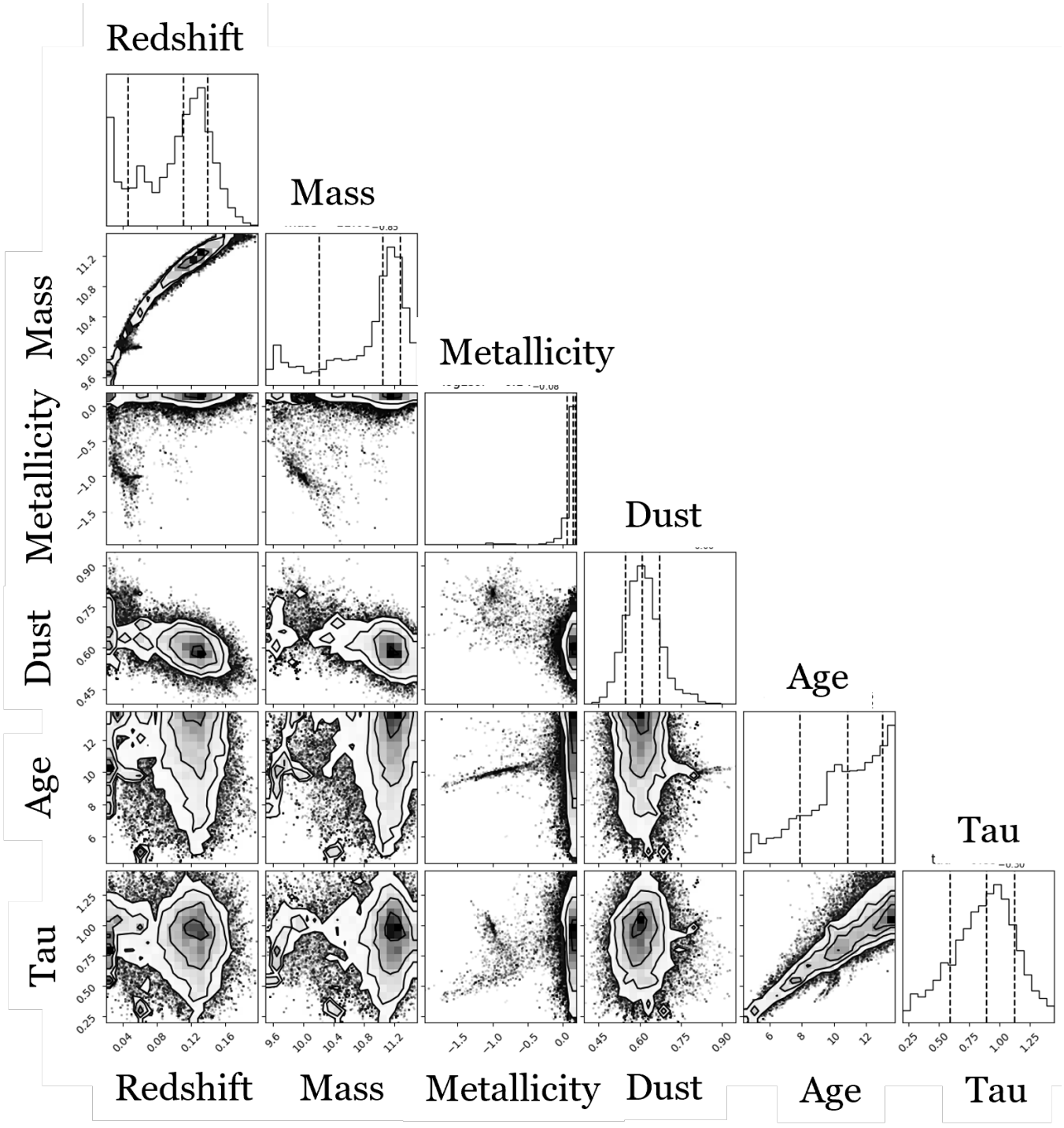

Figure 2: Corner plot showing the combined posterior distributions for the six model parameters and the galaxy fit in Figure 1.

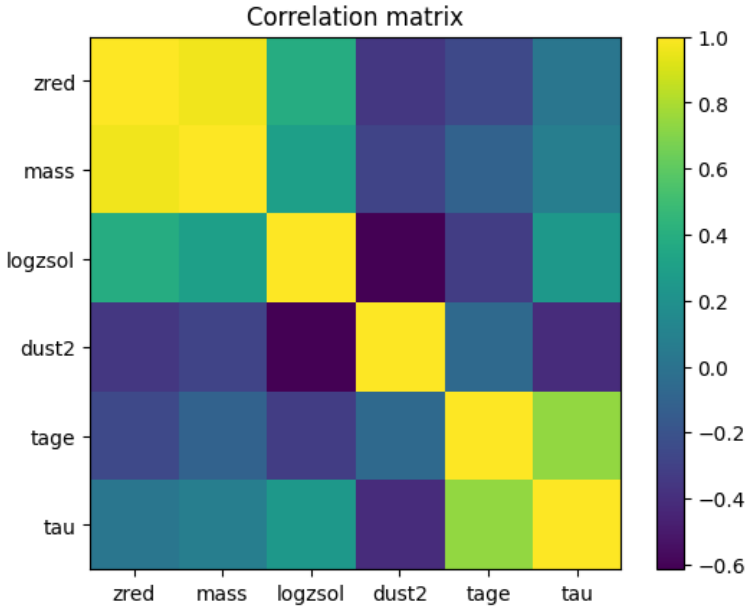

Figure 3: Visualization of the correlation matrix of the six model parameters for the galaxy SED fit in Figure 1. Green to yellow colours show increasing positive correlation, whereas dark blue to purple colours show increasing negative correlation. The median blue colours show no substantial correlation.

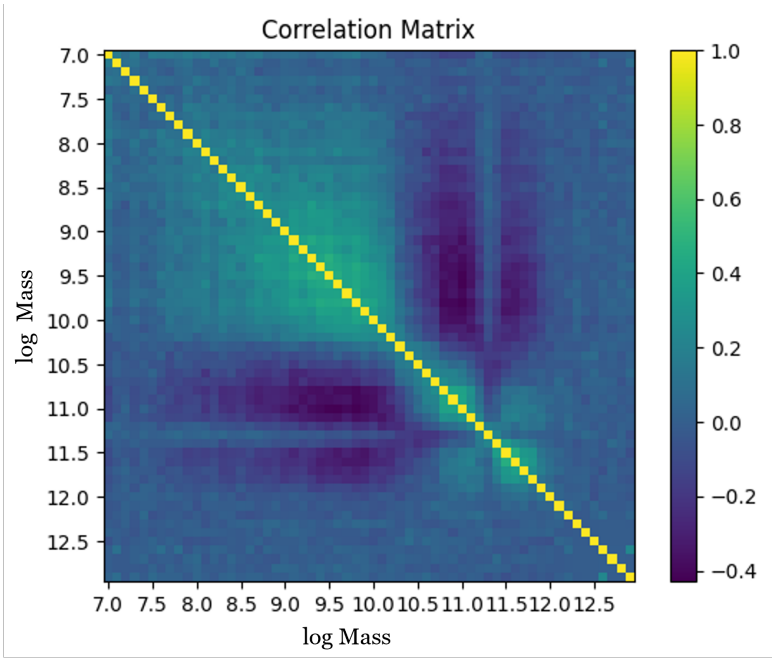

Figure 4: Jackknife correlation matrix of stellar mass bins for a sample of galaxies.

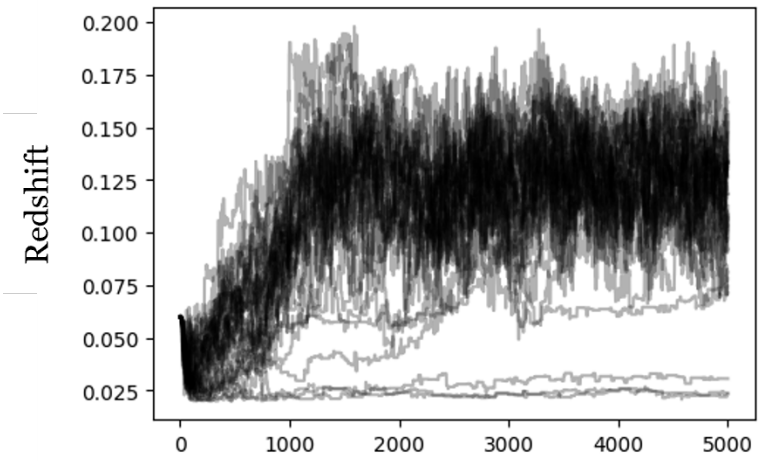

Figure 5: Example of 32 Markov chains taking 5000 steps to explore the parameter space of the redshift value for the galaxy being fit.

space for a galaxy is shown in Figure 5. We use the Python package EMCEE (Foreman-Mackey et al., 2013) to perform an MCMC run on our test galaxy. MCMC is a relatively easy and visual way to explore the parameter space of a model. However, it can be computationally expensive. Additionally, MCMC is highly dependent on the choice of priors. Informed priors are likely to lead to a much better fit. It is also possible for chains to get stuck in a local minimum when exploring the parameter space, leading to the appearance of a well converged fit but missing the global minimum.

3.2 Nested Sampling: dynesty

We use the package DYNESTY (Speagle, 2020) to explore nested sampling techniques. Nested sampling explores the best fit model parameters by computing the Bayesian evidence by integrating the likelihood over the prior range. The use of 'live points' allow nested sampling techniques to avoid wasting too much computing power sampling low-likelihood regions and focus on regions of higher likelihood. Nested sampling can be faster than MCMC techniques and show a clear convergence on a best-fit. We can see this based on the likelihood, importance weight PDF and evidence, shown on Figure 6. However, just like with MCMC, there is the possibility of finding a local best-fit that is not the global best-fit. While MCMC techniques may require a 'burn-in' (some number of points need to be removed from chains at the beginning where the run is exploring near the initial guess), nested sampling techniques do not require burn-in.

In the best case for this report, DYNESTY and MCMC both performed similarly. The prior distribution was chosen based on the initial PROSPECTOR run, and the DYNESTY results had narrower posterior distributions for some of the parameters. DYNESTY was tested with a fixed-redshift model as well, which removed the mass-redshift degeneracy and allowed the mass to converge. Nested sampling models can be compared with the Bayes factor, which is simply the ratio of the evidences. In this case, the Bayes factor suggested the free-redshift fit to be better, while a χ^2_ν comparison suggested the fixed-redshift was a better fit.

4 Machine Learning

Very generally, machine learning (ML) techniques give machines the ability to learn from data to make further predictions. We use SCIKIT-LEARN (Pedregosa et al., 2011) There are broadly two

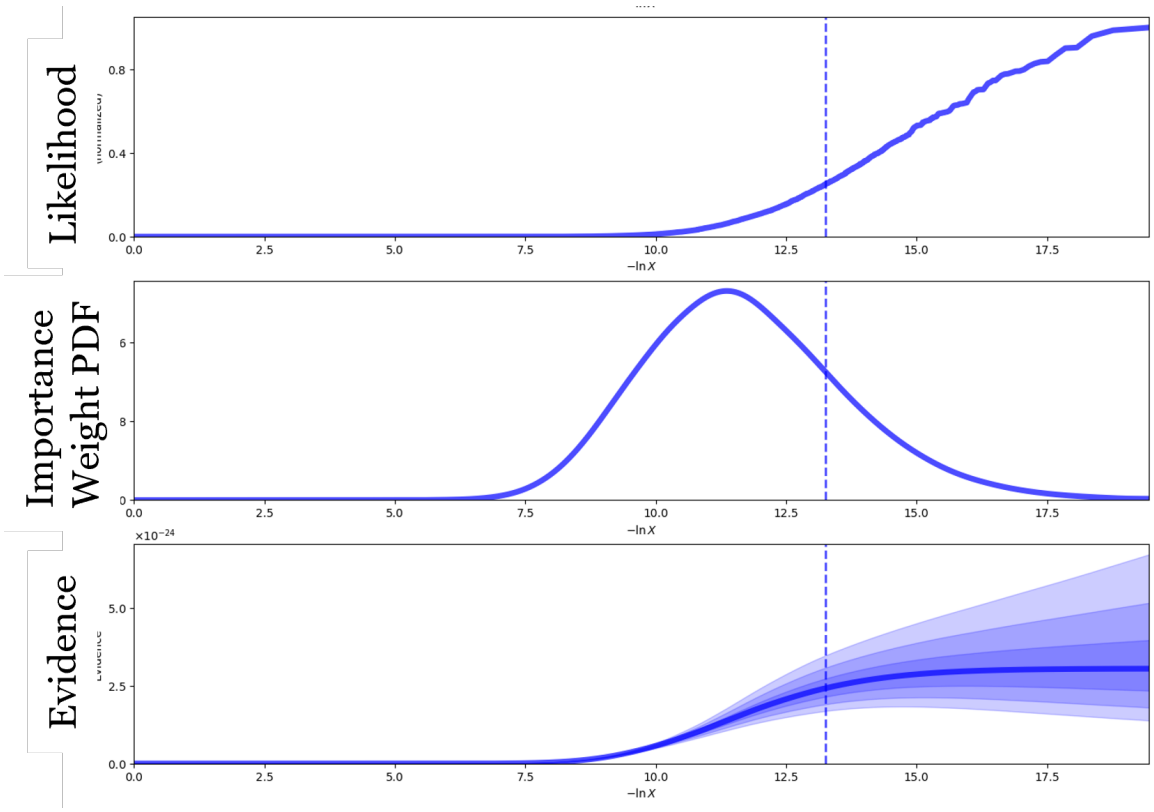

Figure 6: Nested sampling parameters that show convergence to a best-fit value. When the importance weight PDF peaks then decreases and the evidence plateaus, the nested sampling run has reached a good fit.

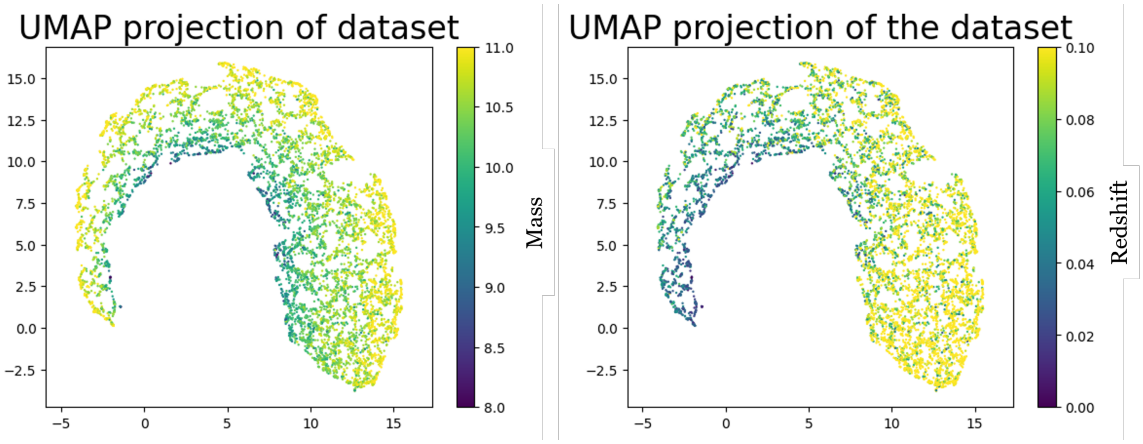

Figure 7: UMAP dimensionality reduction of optical photometry of a sample of galaxies. The left highlights the trends in the UMAP manifold withmass and the right with redshift.

types of ML:

- Unsupervised Learning: In this case, you do not know the 'right answer' about your data. Many of these techniques involve dimensionaltiy reduction and clustering, where the algorithm takes high-dimensional data and reduces it to two dimensions by combining variables so that they can be easily visualized and the data can be separated into clusters.
- Supervised learning: In this case, you use a majority of your data to 'train' a model. The training involves giving the algorithm input parameters and the outputs you wish it to predict. The model that is formed can then be validated on the remaining data and used to predict parameters for new data.

We used UMAP dimensionality reduction, spectral clustering and linear regression to predict mass and redshift without SED fitting. By passing in optical fluxes and uncertainties, the resulting UMAP manifold shows trends in stellar mass across one axis, and trends in redshift across another axis. These results are shown in Figure 7.

We then ran spectral clustering on the compressed data. Choosing the number of clusters was non-trivial since the parameters we hope to cluster around are continuous. Choosing three clusters, the results showed similar distributions of mass in each cluster. These clusters are shown in Figure 8. However, the redshift distributions were different in each. As another test, the clustering was ran on the uncompressed data. The resulting clusters had distinct distributions of both mass and redshift. However they are much harder to visualize given the high dimensionality of the data.

Finally, we trained a linear regression model using the optical fluxes and uncertainties for 7000 galaxies. The resulting model was validated to predict mass and redshift on the remaining 1000 galaxies in the sample, and worked very well! There was negligible bias in the values when comparaed to the catalogue from (Chang et al., 2015), and the scatter was 0.025 for redshift and 0.3dex in stellar mass, as seen in Figure 9. However, when tested on a sample of starburst galaxies, the model did not perform well, as seen in Figure 10. This highlights one of the key problems with using ML techniques in place of machine learning. "Outlier" galaxy types, such as starbursts or very dusty galaxies, are likely to be underpresented in the training set, meaning that the model will likely not be able to predicts their properties well. However, ML techniques in these tests were orders of magnitude faster than SED fitting, taking less than an hour to train and test on thousands of galaxies, compared to several minutes per galaxy for SED fitting.

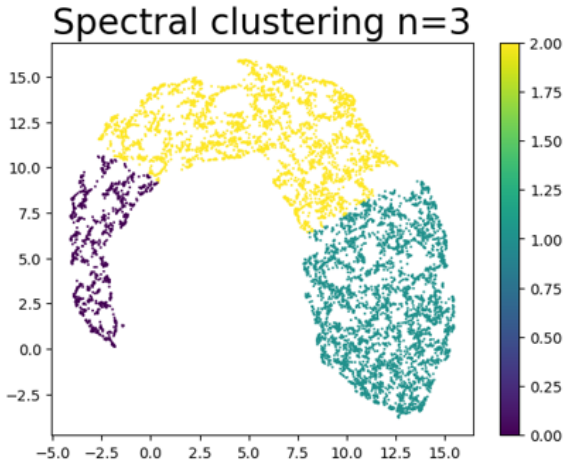

Figure 8: Spectral clustering with three clusters of the UMAP reduced data from Figure 7.

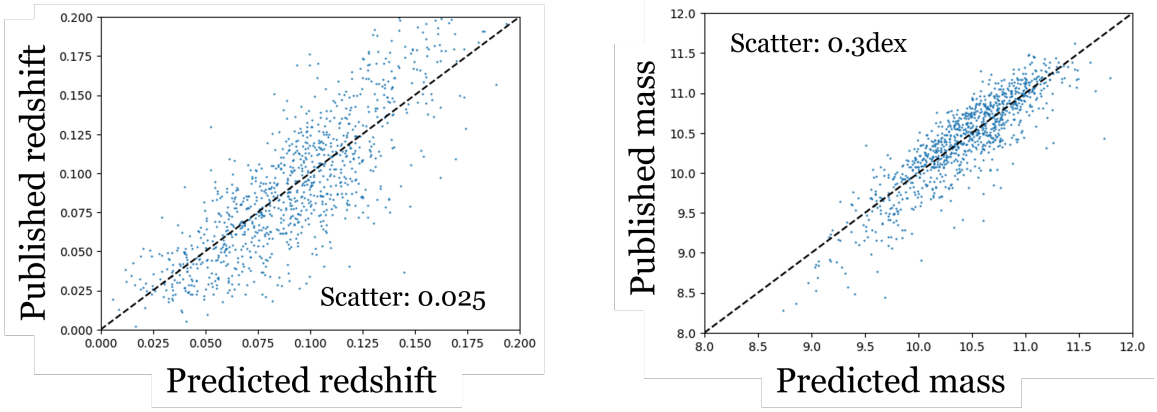

Figure 9: Linear regression predictions compared to those from (Chang et al., 2015) for redshift and mass for 1000 validation galaxies after training the model on a set of 7000 galaxies.

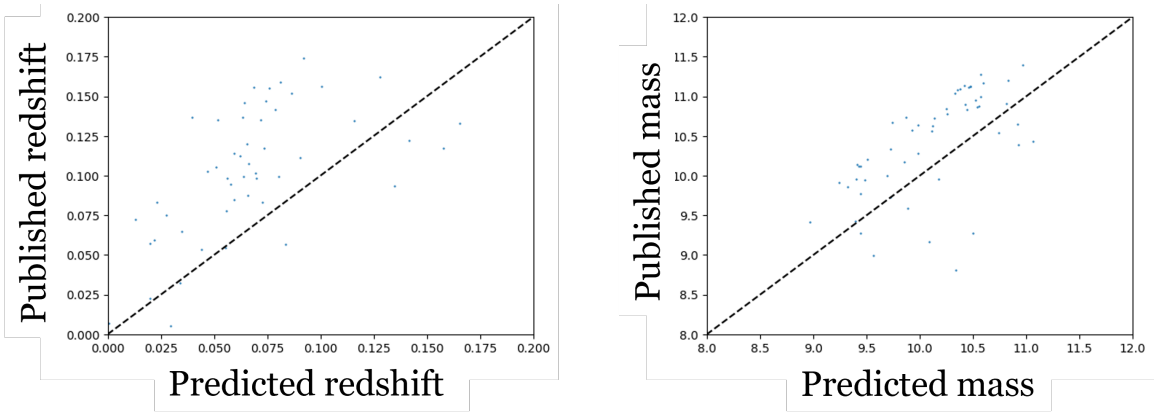

Figure 10: Same as Figure 10 but for a sample of starburst galaxies.

5 Use in research

SED fitting is a key tool in studying galaxy populations. Codes such as PROSPECTOR have been recently developed and can perform very well - however SED fitting remains a challenge. ML learning can be used in place of or to supplement SED fitting, but has its own challenges. While ML techniques tend to be fast, they are most effective at predicting the properties of "average" galaxies. SED fitting is much slower but is more able to predict the properties of individual galaxies, regardless of the average of the sample.

In my own research, I plan to undertake a project which involves a comparison of high-resolution optical, UV and Hα imaging of galaxies. Since UV is particularly sensitive to dust, spatially-resolved SED fitting may be an important part of this project such that the spatial distribution of dust can be mapped and corrected for. The techniques I have learned here will all be applicable. It is useful to understand the different sampling techniques that codes such as PROSPECTOR use for a better understanding of SED fitting.

Beyond SED fitting, I believe that ML techniques are very useful in trying to learn about large sets of data. In the case of galaxy research, this could include clustering techniques to classify galaxy morphology based on certain measurable properties, or even training models to predict galaxy morphology or different quenching mechanisms acting on galaxies.

Acknowledgements

Thank you to the postdocs (Liza, Ana, Jack, Pierre, Marco and Simone) for making this course happen and taking the time to convey your expertise on statistics and machine learning to us. This course is just one of countless examples of how you all help to make graduate life in the WCA exciting and accessible, and for that (I think I can speak for all graduate students) we are very grateful.

References

Chang Y.-Y., van der Wel A., da Cunha E., Rix H.-W., 2015, , 219, 8

Foreman-Mackey D., Hogg D. W., Lang D., Goodman J., 2013, , 125, 306

Johnson B. D., Leja J., Conroy C., Speagle J. S., 2021, , 254, 22

Pedregosa F., et al., 2011, Journal of Machine Learning Research, 12, 2825

Speagle J. S., 2020, , 493, 3132