Domain Adaptation for Decoding Dark Matter with Strong Gravitational Lensing

Prospective student: Marcos Tidball

Mentors: Pranath Reddy, Michael Toomey, Sergei Gleyzer, Anna Parul and Sourav Raha

Introduction

Dark matter is one of the biggest questions in current cosmology, and many different theories were created to try to explain it. One of the challenges of studying dark matter is actually finding it, as it is only noticeable due to gravitational interactions. Fortunately, recent results (as discussed by Alexander et al. (2020)) have shown that strong gravitational lensing can be used as a probe for studying dark matter's substructure.

Unsupervised learning algorithms are particularly interesting in this field because they allow for the identification of dark matter substructure without a prior theoretical model assumption. Alexander et al. (2020) have already shown how promising this technique can be, labeling strong lens systems as having dark matter substructure for them to then be analysed more thoroughly with techniques that identify the type of substructure in the system. The pipeline that is currently being developed by this group that combines both strong lens simulations and deep learning models for strong gravitational lens modeling is called DeepLense.

While the performance of these algorithms is very promising, there is still a large gap when compared to supervised learning algorithms. One promising possibility is to use domain adaptation techniques to fine tune the models trained on simulation data with real data. Thus, this project will focus on using domain adaptation to account for the differences in the modelling and available real observation data, while also improving the interface with PyAutoLens, the software used for creating the strong lens simulations. This will enable the expansion of the DeepLense pipeline while also helping researchers have a more precise way of detecting dark matter substructure in strong gravitational lens images.

Evaluation Test

Each DeepLense related project had its specific evaluation test as well as a common test related to the usage of PyAutoLens. My submission with both tests is available here. The specific test was related to the creation of an unsupervised anomaly detector based on an Adversarial Autoencoder (AAE) architecture that uses transfer learning to differentiate between strong lensing images without dark matter substructure and images with dark matter substructure.

When I first contacted the mentors I had one week to complete the tests and unfortunately I had a lot of university work during that period. This, coupled with my lack of experience with AAEs, made it so that my first submission got poor results and had lots of mistakes. However, since I am **very** interested in this project, I continued researching about AAEs, unsupervised learning and anomaly detection in order to make a better submission that fixed the mistakes I had made. I contacted one of the mentors with some of my questions and received some suggestions that really helped me out.

My original submission is still available in the repository containing my evaluation test, but so is my new one. I just wanted to make this statement so that my submission, if already evaluated, could be evaluated again considering the rework that I've done!

Project Goals

During Google Summer of Code:

- The creation of a dataset with real observation data and simulated data;
- A systematic analysis of different unsupervised models and architectures;
- A systematic analysis of different domain adaptation techniques;
- A user-friendly implementation of the above-mentioned learning algorithms with proper documentation that make them easy to import and use;
- Improvements to PyAutoLens' user interface.

Future development:

- Exploration and development of graph-based models (as suggested by Alexander et al. (2020));
- Exploration and development of unsupervised few-shot learning models trained on real observation data;
- Creation of a source extraction pipeline that enables automatic extraction of candidates from large astronomical images to be classified by the unsupervised models.

Implementation Plan

During Google Summer of Code:

The creation of a dataset with real observation data and simulated data:
Having a good dataset is crucial to having a good model. With this in mind I will do research about surveys and data releases that have observation data of strong lensing systems with substructure. I will also research more realistic and complex simulation techniques that could be used in order to create a simulation dataset of strong lenses. The Bologna Lens Factory is an interesting starting point since their simulations have already been used in challenges related to classifying strong gravitational lenses.

When testing the models that we will use it may be interesting to have metrics both for simulated and real images. This will allow us to see how well they generalize to real images after being trained on real data by starting from models that have been trained on simulated data, for example.

A systematic analysis of different unsupervised models and architectures:
In order to develop better models it is necessary to study the ones that already exist. I will study current literature in this topic and experiment with different promising models and implementations. I will be especially interested in architectures related to AAEs. After implementing the most promising ones, I shall train them and compare their performances with previous literature in this topic (i.e. Alexander et al. (2020)) and also supervised learning architectures that will serve as the baseline.

A systematic analysis of different domain adaptation techniques:
Equally important as the models are the techniques that will be used for domain adaptation. Transfer learning, for being both easy to implement and widely used, can serve as the baseline. But more complex and model-specific techniques such as the ones discussed by Wang et al. (2018) are of large interest and will be implemented and tested. Now, for the systematic study, the baseline could be the models without domain adaptation techniques.

In both this and the previous task, techniques such as grid search, which allows us to search for the best hyperparameters are not only useful but fundamental in order to have a thorough analysis of the performance of the models and to find the optimal model.

A user-friendly implementation of the above-mentioned learning algorithms with proper documentation that make them easy to import and use:
In order for results to be reproducible and for the models to actually be used by other researchers it is necessary to have a well documented and easy-to-use implementation. I will maintain a similar level of code explanation as I did in my evaluation test. I will also strive to create a library with intuitive functions and classes in order to maintain the same level of explainability already present in the DeepLense pipeline (e.g. unsupervised-lensing).

Improvements to PyAutoLens' user interface:
I will research and study PyAutoLens' API in-depth in order to understand how the simulations actually work and how to create them more easily. Both the PyLensing repository and PyAutoLens' documentation will be my starting point. I will also try to create functions that can simulate more complex dark matter models in a more intuitive way.

Future development:

Exploration and development of graph-based models:
Graph Neural Networks have recently gained a lot of popularity due to how they can be applied to physics and other natural sciences (Zhou et al. (2019)). I will need to do research on how they could be used for unsupervised learning and also for problems related to image classification. But it is definitely a promising architecture to try!

Exploration and development of unsupervised few-shot learning models trained on real observation data:

Few-shot learning is a very promising technique which enables models to be trained on just a small amount of data. An exploration of these techniques could be especially useful considering the small quantity of real observation data available. While a combination of simulated images + real images as originally proposed may have the best performance, few-shot learning algorithms like the one proposed by Khodadadeh et al. (2019) could provide interesting results.

Creation of a source extraction pipeline that enables automatic extraction of candidates from large astronomical images to be classified by the unsupervised models:
There is a huge variety of current and future surveys that can provide valuable data, with different bands and depths. An implementation of a pipeline that optimizes a tool like SExtractor to detect strong gravitational lenses, extract cutouts of these objects and feed them to the unsupervised models could be especially useful when examining images from telescopes.

Timeline

My classes will end officially on the 31st of May, but by May 20 I will have already finished most of my current courses. The following timeline is a rough guideline on how I plan to do the project:

May 17 - May 24

Familiarize myself with the mentors and community surrounding DeepLense, the main tasks for the current project, already existing documentation. related projects and the current available data. I will also learn about the problems with PyAutoLens' current interface and what can be done to improve it.

May 25 - June 7

I will do research related to domain adaptation techniques for unsupervised learning and different unsupervised learning algorithms. I plan to do this by asking the mentors for interesting papers, checking Papers With Code and reading surveys related to the topic. I will also research about (ways of simulating) more realistic data if necessary. I will discuss my findings with the mentors as I read about these techniques and create a selection of the most promising ones to be implemented afterwards.

June 8 - July 16

- A thorough exploration of the current available dataset and an analysis of data from other sources that could benefit the deep learning models and provide more realistic data. I will also research how to create better simulations with PyAutoLens and implement functions that can create them automatically.

- An initial implementation of domain adaptation techniques/unsupervised learning models and data loaders using Jupyter Notebooks. These models will be working, being able to import the training data, train and evaluate new data. I plan to keep

close contact with the mentors during this phase as neural networks can be very tricky to implement.

- If possible, I will start to train and test the models on the available dataset.

July 17 - August 23

- An implementation of the working models and data loaders with a structure similar to CMU DeepLens and DeepLense's unsupervised-lensing that enables users to easily download the models, instantiate them, train them and evaluate data with them. I will also provide proper documentation and create examples using Jupyter Notebooks.

- Start (or continuation) of model testing. These tests will be compiled in a CSV file that contains important metrics such as true positives, false positives, true negatives, false negatives, AUROC and the hyperparameters used (learning rate, optimizers, batch size). I will also provide plots of the losses, ROC, confusion matrix and other metrics that may be necessary. This will allow for a systematic analysis of each of the chosen unsupervised learning models and domain adaptation methods.

- Create functions that provide a more intuitive interface with PyAutoLens. The mentors' input will be especially important here. I will follow on the steps of PyLensing in order to create an interface that can simulate data for machine learning tasks more easily.

About Me

I have a good working knowledge of Python and PyTorch, having used both for different projects. I have a huge interest in exploring machine learning applied to physical sciences, in special astronomy. For the last year I have been working in a research project that does just that, using Convolutional Neural Networks (CNNs) to classify a special kind of galaxy: Low Surface Brightness Galaxies (LSBGs). These are very diffuse and hard to spot, but are also really important in order to discover more about the universe. I have experience dealing with astronomical data, using source extraction tools on DECam images, fine tuning the thresholds on these tools to extract only LSBG-like objects. In order to find these thresholds it is necessary to analyse the brightness distribution of candidates, size distribution, and other parameters. I also have experience processing FITS images to use them in CNNs. The pre-processing steps are especially important when using transfer learning models, which is exactly what I currently do.

I have experience with reading academic papers and code related to machine learning, astronomy **and** machine learning in astronomy. I am currently writing an article to describe the project I am doing, so I am also gaining experience on writing scientific papers.

I am passionate about this area! I plan to keep working on it during my graduate studies and I really believe that the next breakthroughs in both astronomy and physics will be made possible thanks to the usage of machine learning models. I want to work with this project in order to immerse myself more in this area and to be able to provide significant contributions to the astronomy community!