

Convolutional Autoencoders for Gravitational Lenses Candidate Search

1. Introduction

The purpose of this research projects was to determine if simple auto-encoders can be used to find potential gravitational lens candidates. In particular, a convolutional auto encoder was used in this project.

Auto-encoder is a type of neural network which encodes and then decodes some data, usually images. Auto-encoder consists of two parts. First part is encoder which takes in image in a format of tensor and performs certain transformations to encode it into a smaller tensor, called latent space. Second part of the network is decoder, it is symmetrical to the encoder in its structure but performs an opposite task. It takes in the transformed tensor outputted by the encoder and performs certain transformations to expand this small tensor to the original size, trying to recreate original data as close as possible. To determine accuracy of the network loss value is calculated after each epoch by comparing original picture to the one passed through the auto encoder.

The idea behind using auto encoder latent space to find potential gravitational lens candidates lies in the fact that we have vast amounts of pictures of space structures that might be gravitational lenses, but from all those potential matches very few will actually end up being gravitational lenses. We use this fact to train our auto encoder. Since the dataset is heavily imbalanced, containing way more non lenses than lenses, it is hypothesized that in the process of training auto encoder will adapt to non-lens images and therefore do better job in terms of how close original image to the processed one.

2. Building Autoencoders

At first, to test if convolutional autoencoders can be trained to reconstruct images of gravitational lens candidates with high degree of accuracy, a simple autoencoder was developed that would work with grayscale representation of images. It took in reduced size images (256x256 reduced to 28x28) and outputted their latent space and reconstruction. Autoencoder

was able to shrink images to 8x8 latent space and then reconstruct them with pretty accurately, visually images looked very similar to the originals, but also had less noise than originals. Next, new autoencoder was developed to work with full size (256x256) grayscale images and results were the same. Autoencoders had 3 layers for encoder and 3 layers for decoder parts, each layer containing 1-3 filters, therefore training process was very fast.

After such approach was determined to be promising, I started working with full size RGB images. First, to simplify the process, I created 3 autoencoders: one for each color layer of image, that would independently process each color layer of image and in the end, I would combine reconstructions together to get an RGB image. I just copied code from the previous autoencoder 3 times and trained them independently. Each autoencoder was trained in the following way: a set of 3000 randomly chosen images from over 300,000 images to be used as a validation set across the whole training process, next 10000 images would be randomly chosen for a training set from a pool of 300,000, autoencoder would train for 5 epochs with batch size of 128 using those 10,000 images, after which next 10,000 would be selected, and the training process would repeat. Overall, each of 3 autoencoders went through 15 such cycles, while validation set would remain unchanged for each cycle and after training encoder parts of each autoencoder were compiled into separate models to later be used for obtaining the latent space.

3. Classifier training

To classify images based on their latent space, and XGBoost classifier was selected as it is very adaptive, precise, and fast in training.

A set of 255 confirmed gravitational lenses, and confirmed non-lenses was used for further training. Label arrays were created as well and then the sets were split into training subsets and validation subsets to be used for XGBoost classifier training.

As described earlier, it was hypothesized that reconstruction of images of gravitational lenses would be less precise, and therefore standard deviation of squared errors between original images and reconstructed would be higher for gravitational lenses. To use this information in

classifier training, images would be passed through autoencoders, reconstructed, and standard deviation would be calculated. The standard deviation value was appended to the end of a latent space of a respective image, reshaped into a 1-dimensional array. This gave us an 1 dimensional array of size 193x1, where 192 values were values from an image latent space and 1 value the standard deviation of squared errors between reconstructed and original image. Every image in the training and validation sets was converted to this format.

Next, an XGBoost classifier was trained using a converted training set.

To test the accuracy of obtained classifier, it was fed the validation set, which achieved 89% accuracy, but the false positive rate was rather high. To combat this, a new classifier was trained on a bigger set: 8000 non-lenses and 200 lenses for training. Resulting accuracy on a validation set with 2000 non-lenses and 25 lenses was close to 1, and false positive rate nearly zero. To improve these numbers even further, images were reclassified using higher threshold.

Probabilities of each class for each image were obtained from the classifier, and then manually classified, assigning 'lens' class to images with probability of lens class >67% (average 'lens' probability for non-lens image) and non-lens to the rest. As a result, all lenses were identified correctly, with no non-lenses classified as lenses. For further testing 30,000 random images were read from unclassified dataset, and mixed with 25 lens images that were not used for classifier training to simulate low density of lenses in an realistic dataset. Regular classification was 45% accurate, and an average 'lens' probability of non-lens image classified as lens was 0.6724633574485779. When that value was used as a classification threshold, resulting accuracy was 99.9%. Out of 25 lenses, 21 were classified as lenses, and only 4 as non-lenses, but what is more important is that none of the non-lens images were classified as lenses, which is more important, since it potentially reduces wasted telescope time when checking the candidates.