Week 6

Progress during 3-9th Aug

Hao SUN

August 10, 2017

Contents

1. Unsupervised star-galaxy segmentation and classification
   - Tried to use more hidden variables
   - Tried different normalization method
   - Improved the classification accuracy to 88.8% (supervised 90.1%)
   - A sum-up diagram

Progress in this week

1. Tried to use more hidden variables, improved the segmentation result
2. Tried different normalization method
3. Improved the classification accuracy to 88.8% (My tantative supervised approach: 90.1%)
4. A model architecture diagram

Segmentation with more hidden variables

With more hidden variables, the VAE can learn more information about the input images.

By comparing the performance of networks under different hyper parameters. I choose to use 30 hidden variables for the hidden layer. And use 4 Convolution layers and 3 Maxpooling layers in encoder while 4 Convolution layers and 3 Upsampling layers in decoder.

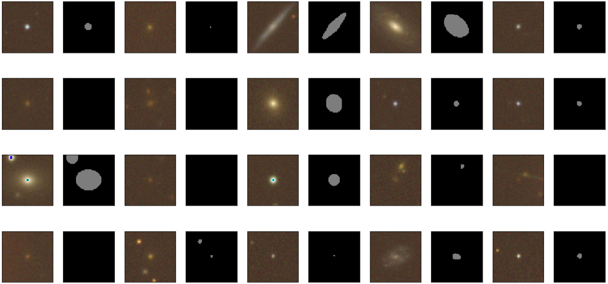


Figure: 1 hidden varialbe with the first 10 layers as input

Segmentation with more hidden variables

Using more hidden variables will lead to better result, especially when including the layers of decoder in Hypercolumns

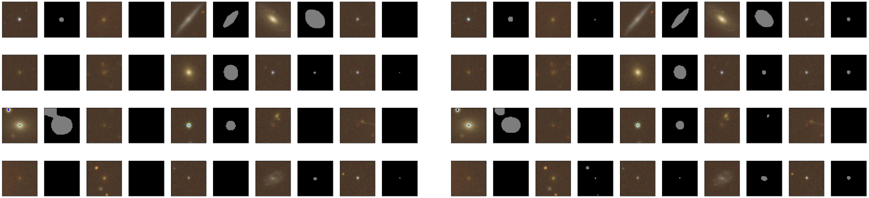


Figure: 50 hidden varialbe with the first 10(L) or 20(R) layers as input

Segmentation with more hidden variables

Actually the layers included in Hypercolumns is also a hyper parameter.
We use Hyperparameter instead of using the original images to run the clustering algorithms because the Conv layers can implement data denoising.
So, in order to get better denoised input data, I use only the i-band as the only input channel for VAE. (TODO: try to use both gri channels)

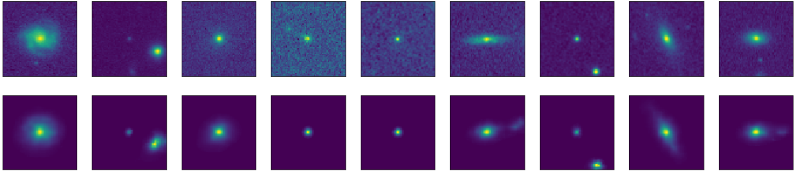
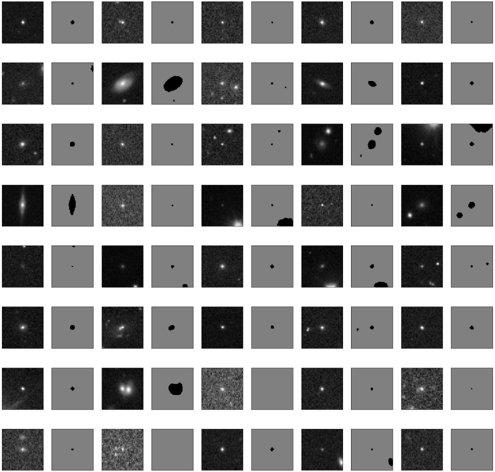


Figure: some of the inputs and outputs

Segmentation with more hidden variables

Segmentation result when use the first 3 layers and the output layer as inputs for Hypercolumns. This result is much better than before.



Different normalization method

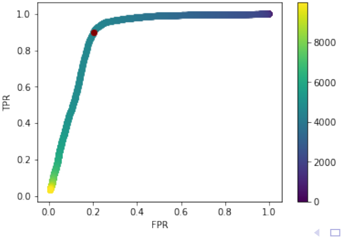
1. Zero-mean normalization
2. Zero-mean scaling
3. Min-Max normalization
4. Scaling to [-1,1]
5. Combination

In VAE, the output is either 64*64*5 images or 64*64*1 images, with each pixel value has the same range [-1,1] or [0,1] (depend on the activation function of the output layer)

Mapping the input data into [-10,1] or [-10,2] by Z-score normalization leads to the best result so far. But some of the information is lost. (TODO: go on to try other normalization method)

The classification task (ROC curve)

- In classification task, more hidden variables lead to worse result.
- Although more hidden variables can capture more useful information, those hidden variables are not orthogonal to each other. In other words, an useful feature may be mapped into several hidden variables, instead of contained by a certain hidden varible.
- Then when we use manifold learning, the geometry may be destroyed and then lead to an low accuracy. (learn more imformation can lead to better reproduction, but may not help for classification, the only goal for a VAE is to reproduce the input image but not separate them into different classes)
- I got the highest accuracy when I used 3 hidden variables.



The classification task (result)

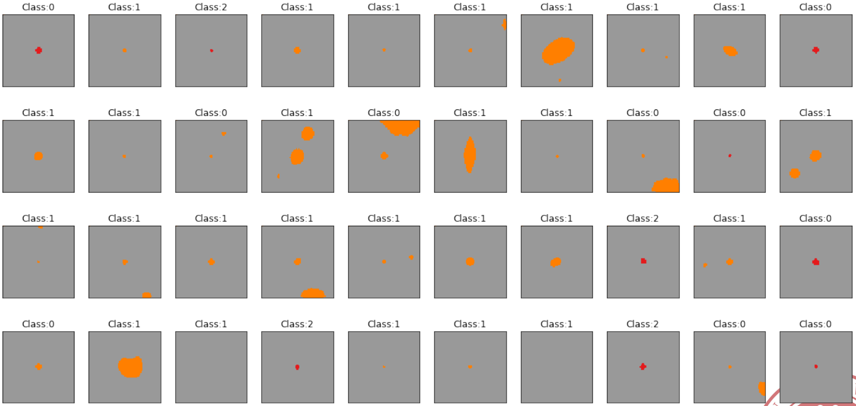


Figure: Red: stars/QSOs; Orange: galaxies

Summary and TODOs

- Summary
  1. Pixel level unsupervised learning. Reached a higher segmentation sensitivity about 95% and classification accuracy over 88.8%.
  2. A model architecture diagram on [Github]
- TODOs
  1. Revise the normalization method to make it more scientific
  2. Try to use 3 channels, or use u-g, g-r, r-i, i-z instead
  3. Use SDSS field data (2,000*1,500 large scale frame data)
  4. Try more hyper parameters (kernel numbers, strides)