UNet Segmentation Update

20191115

Status as of Aug. 2019

- Pre-processing
  - Obtained aligned DAPI and H&E cores
    - Currently using the cores from 20190116_TMA-2_IF_40x
  - Scaled DAPI and H&E images to 20x
  - Percentile thresholded the DAPI images at 90%
  - Cropped 324 256x256 tiles from the center of each core
  - Split dataset into train (90%) and test (10%)
- Created Jupyter notebook for training a UNet, including:
  - Splitting up the train set for k-fold cross validation (k=5)
  - A UNet instance with default parameters
  - Matplotlib plots for assessing model performance
- Goal: be able to hand this notebook over to someone else to be able to tune parameters and optimize the model

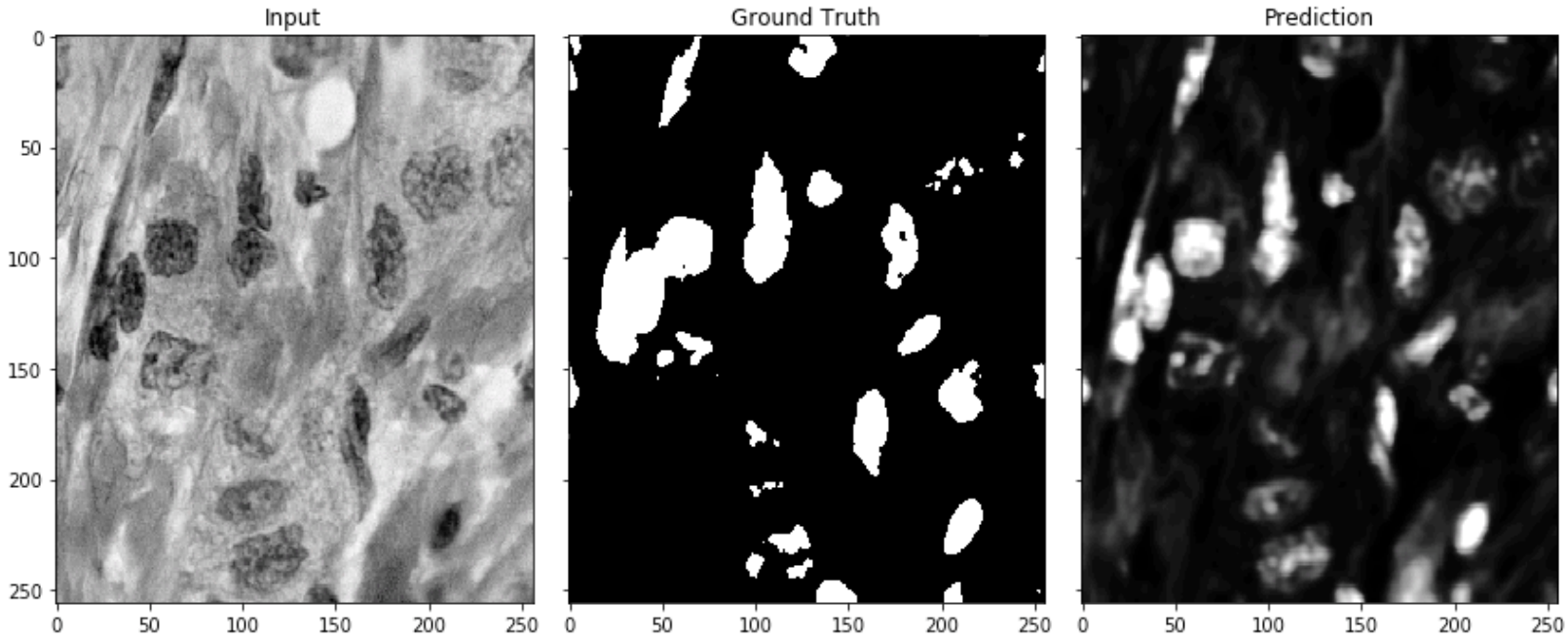
Known Issues as of Aug. 2019

- Max value of prediction is 0.5, but should be [0, 1)
- Final output should be binary, but it isn't
- Measures of training performance (accuracy, loss) not optimal
- Potential pre-processing problems
  - Focus issues
  - Thresholding issues

Examples as of Aug. 2019

Problem: Non-binary output, max prediction value is 0.5

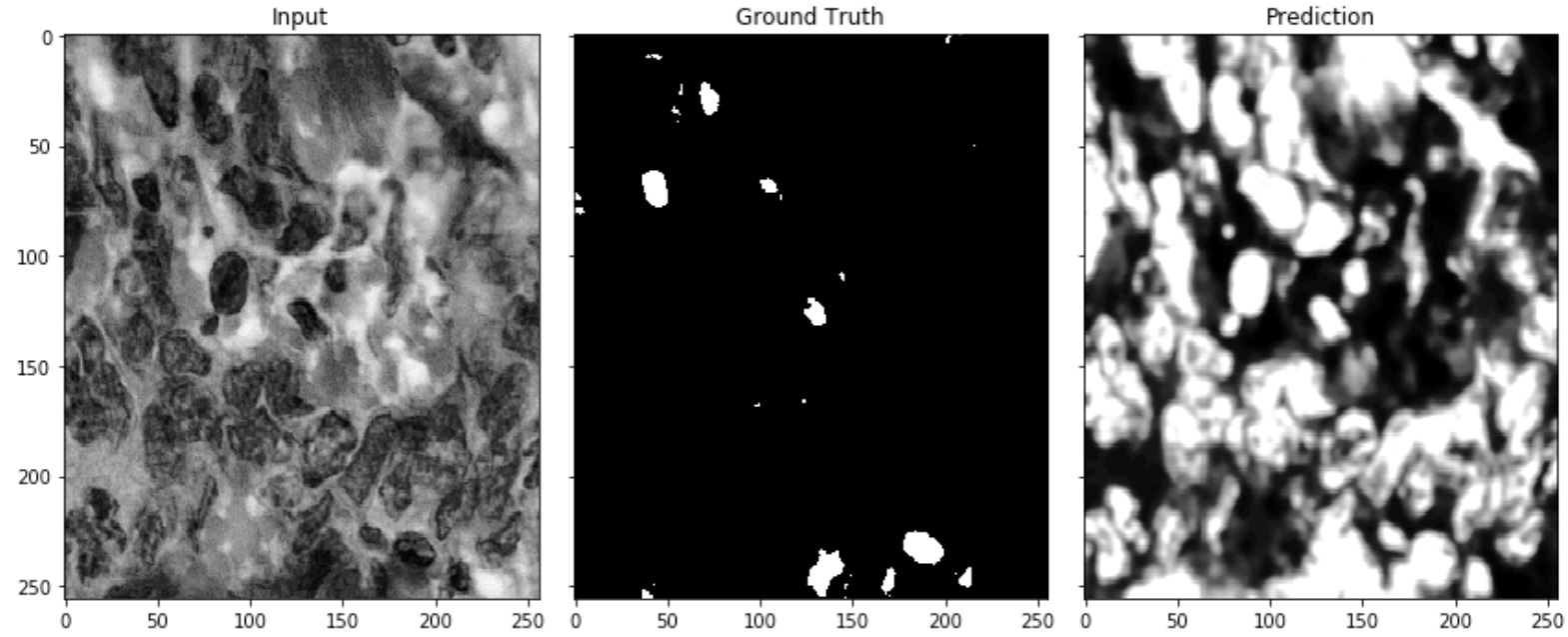
```
Error Rate: 11.21826171875

CPU times: user 14.6 s, sys: 5.4 s, total: 20 s
Wall time: 1.07 s
```



Examples as of Aug. 2019

Problem: Thresholding

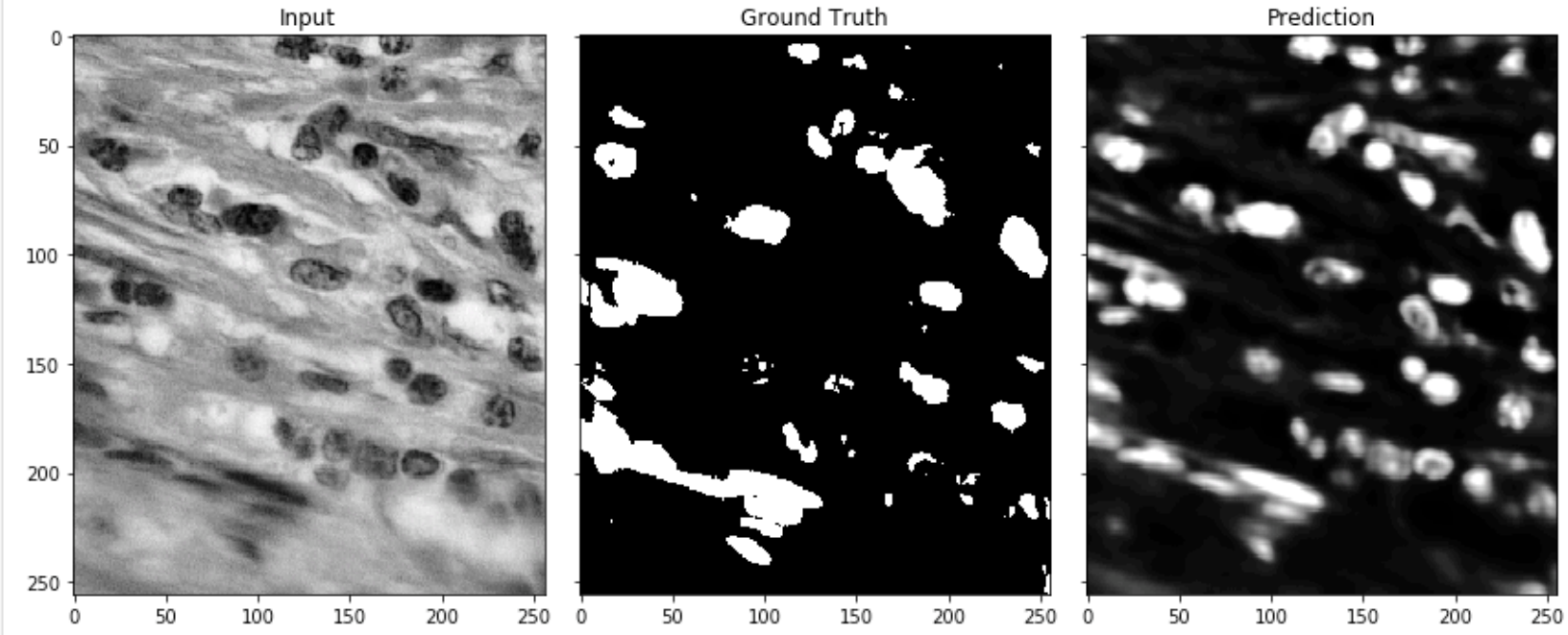
```
Error Rate: 1.69219970703125

CPU times: user 15.1 s, sys: 4.88 s, total: 20 s
Wall time: 1.03 s
```



Examples as of Aug. 2019

Problem: Focus issues

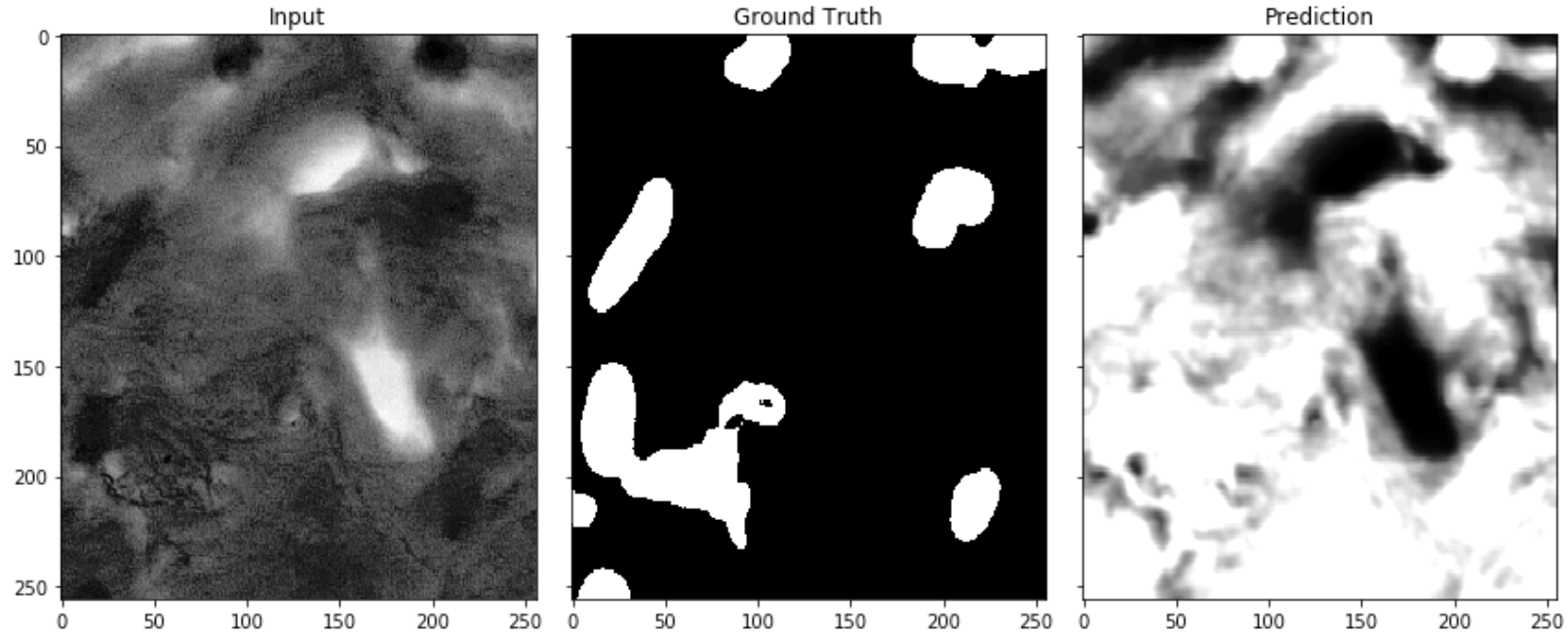
```
Error Rate: 11.5264892578125

CPU times: user 15.1 s, sys: 4.97 s, total: 20.1 s
Wall time: 967 ms
```



Examples as of Aug. 2019

Problem: Output is logically wrong

```
Error Rate: 14.1326904296875

CPU times: user 15.6 s, sys: 4.12 s, total: 19.7 s
Wall time: 1.01 s
```



Updates as of Nov. 2019

- 0.5 Prediction Ceiling
- Non-binary Predictions
- Loss Function Experimentation
- Additional Diagnostics for later Model Eval/Tuning

Updates as of Nov. 2019

- 0.5 prediction ceiling:
  - Removal of the normalization operation in the softmax layer yields predictions in desired [0,1) interval.
  - Max predictions of ~0.96 in a prior model using Adam optimizer on the base model.

```
unique, counts = np.unique(pred_scores
print(np.asarray((unique, counts)).T)
```

```
[[0.01697851 1.        ]
 [0.01698254 1.        ]
 [0.01698915 1.        ]
 ...
 [0.8105042  1.        ]
 [0.81308359 1.        ]
 [0.81338227 1.        ]]
```

```
#count of prediction values
len(np.unique(pred_scores))
```

```
65063
```

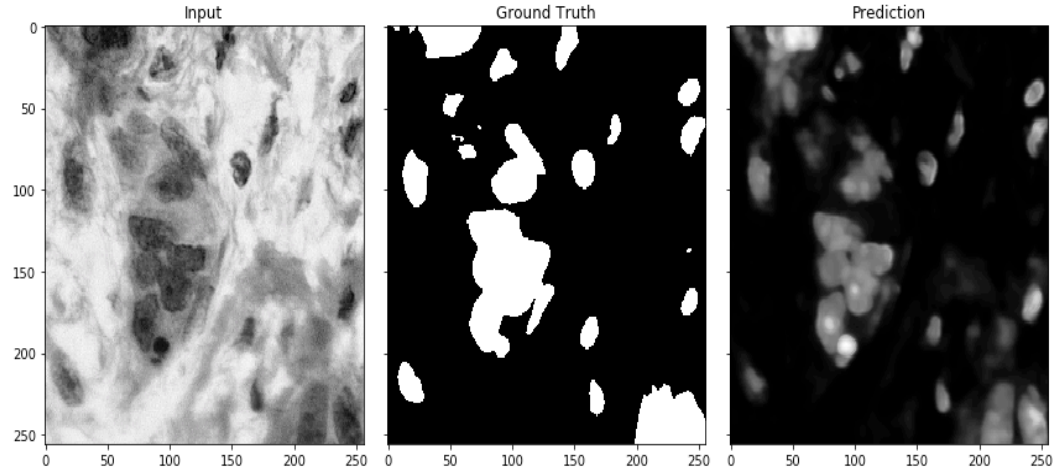
Updates as of Nov. 2019

- Continuous valued predictions:
  - Investigation of loss function:
    - Tried Dice coefficient loss, but model yielded pure prediction masks (all black). This was expected as DC Loss is known to be vulnerable to the exploding gradient problem.
    - Model uses Weighted Cross Entropy to approximate the Paper's Learnable Distance-to-nearest-cell Cross Entropy function. Weight parameters will need tuning once predictions are binarized.
  - Prediction layer thresholding:
    - Applied 0.5 thresholding directly to the unet pixel_wise_softmax predictions. Predictions still returned continuous values but with max predictions at ~0.85

Updates as of Nov. 2019

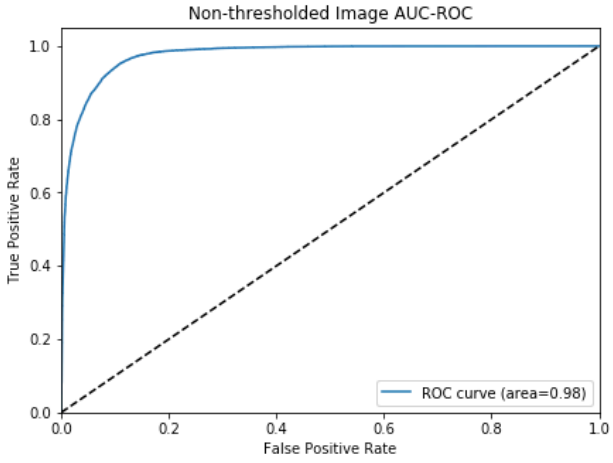
- Added additional sklearn diagnostics to evaluate models and hyperparameters.

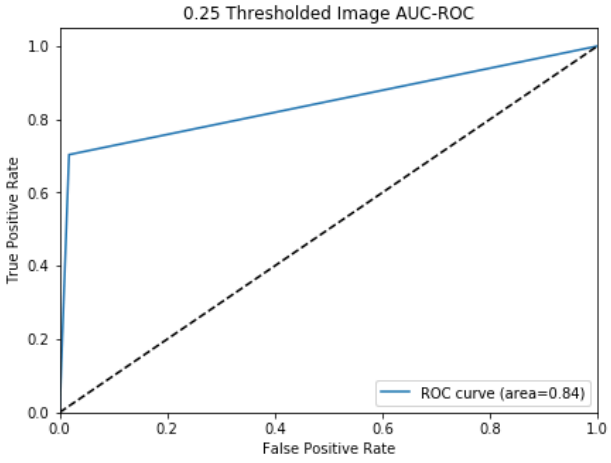
```
Error Rate: 15.64788818359375

CPU times: user 14.5 s, sys: 3.29 s, total: 17.8 s
Wall time: 1.17 s
```

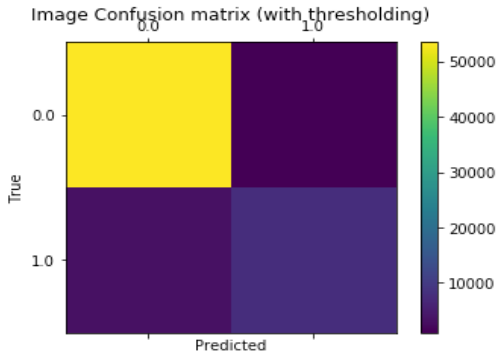


- Output is logically correct, but out of focus. Continuous valued predictions is main obstacle to mask resolution and is throwing off diagnostics. The ROC should be non-smooth. Confusion matrix and F1 is done using post-hoc thresholding.

- Thesholding predictions at 0.25 reduces performance significantly. Thresholded predictions show the model is significantly better at predicting negative class than negative class (with empty class weightings). 0.5 thresholding worsens performance.








```
prec, rec, f1, supp = precision_recall_
print('F1 Score:', round(f1*100,2))
```

```
F1 Score: 78.65
```

Updates as of Nov. 2019

- In progress/next:
- Investigating the continuous valued predictions issue
- Experiment with Class Weightings