Introduction

With more than 1 million new diagnoses reported every year, prostate cancer (PCa) is the second most common cancer among males worldwide that resulting in more than 350,000 deaths annually.

The dataset of this project comes from a Kaggle contest, Prostate cANcer graDe Assessment (PANDA) Challenge, which is about prostate cancer diagnosis using the Gleason grading system.

The result of this contest is published in this paper which shows that the quadratic weighted kappa score of algorithms is higher than general pathologists. This is compared on an external test set (train set comes from EU, external test set comes from US).

Problem Statement

The grading process consists of finding and classifying cancer tissue into so-called Gleason patterns based on the architectural growth patterns of the tumor. After the biopsy is assigned a Gleason score, it is converted into an ISUP grade on a 1-5 scale. 1 means less possibility of having cancer, and 5 means high possibility. A 0 score also exists which indicates a benign case.

In this problem, we are given images of biopsy slices and their Gleason score and ISUP score as train sets and targets. The Gleason score can be used as the target in the training process, but eventually, the score is calculated from the ISUP score.

The score used in the contest is quadratic weighted kappa which measures the degree of agreement between two sets of labels. A 1.0 score means the two sets agree on every observation, and -1.0 means they totally disagree on every case. For example, label = [1, 2, 3, 4, 5] will have a 1.0 score with pred = [1, 2, 3, 4, 5], and label = [1, 1, 1, 1, 1] will have score -1.0 with pred = [5, 5, 5, 5, 5].

The purpose of this project would be to work through the 1st place solution on the Kaggle contest, and also compare its performance with a straightforward solution if time permits.

Details of Data

There are 10617 train images in total. The test image is not given. Each training image is 23904 x 28664 pixels which is too large for training and there are also many pixels with zero values. So the train image is tiled into tile images, each of size 256 x 256. The tile images are then filtered by the information they have (the sum of the value of all pixels). The filtered tile images are then concatenated to become an image that is much smaller than the original one.

If 12 tiles are chosen (3 rows, 4 columns), then the final image will be of size (3 x 256) x (4 x 256).

There is also a .csv file that stores the id, data provider, Gleason score, and ISUP score of each image.

ML Pipeline Analysis Plan

After installing packages and downloading data, the following steps are:

- Straightified 10 fold to hold out 10% data as the test set.
- Group images by their similarity using imagehash.
- Make 5 folds with similar images in the same fold.
- Make tile images for training.
- Train the base model to remove noise. Images with a large distance between the label and the prediction will be removed.
- Train the final model based on clean data with noise removed.

It will take a long time to do all steps on the whole data set and the purpose is to work through the full flow, so will use only a few images here depending on the time of training.