Preprocessing / Miscellaneous

noduleDimensions preprocessing.ipynb
Created noduleDimensions excel sheet
Requires: parsedXML excel sheet, 2 other excel sheets from the MedIX server
Produces: noduleDimensions.xlsx and accompanying data analysis within the notebook

numImages.py
Finds number of slices in each scan
Requires: directory of DICOM images for each scan (on MedIX server)
Produces:

- List of number of slices in each scan
- Min and max number of slices in the dataset

clippingCode.py
Used to find bounds for intensity clipping
Requires: dictionaries of all LIDC pixel data
Produces:

- 1st and 3rd quartiles of pixel intensity values: [-1434, 2446]
- Dictionary of each intensity value and the number of times it appears in the dataset

PosClipping.py
Performs intensity clipping on list of subvolumes
Requires: file containing list of subvolumes
Produces: equivalent list of subvolumes with intensity clipping completed

chunkedSeriesIDs.py
Creates dictionaries of pixel data from DICOM images
Requires: directory of DICOM images from LIDC (on MedIX server)
Produces: 21 dictionaries, each containing reconstructed pixel data from 50 CT scans (only 18 in the last dictionary)

SeriesIDfilecreate.py
Converts dictionaries so there is one dictionary per scan, named by series ID
Requires: dictionaries from chunkedSeriesIDs.py
Produces: one dictionary for each scan, where the key is slice location* and the value is a 2D array of pixel data. The keys are not in order.

*Slice location works for about 75% of the scans, but for the others it does not match the z-coordinates proved in the XML file, leading to difficulties. Image position patient is actually the correct DICOM header field to use in the future.

Hyperparameter Optimization

HP3DCNN3.0.py
Performs hyperparameter random search
Requires:
- Subvolumes representing positive / negative examples (CNNinputDataExtractionV2.py)
- Hyperparameter ranges

What it does:
- Creates a 3D CNN
- Selects 30 random sets of hyperparameters, performs 3-fold cross-validation on 90% of the data using early stopping
- Picks best set of hyperparameters, trains network, and creates ROC curve on remaining 10% of data

Produces:
- Dictionary of each set of hyperparameters and their performance
- Dictionary of best hyperparameters
- ROC curve and AUC score for best set of hyperparameters
- TF graph file
- Graphs of accuracy and loss over the course of final training
- Model files at points in final training with best performance

IC3DCNN3.0.py
Same as above, but all subvolumes undergo intensity clipping

LH3DCNN3.0.py
Same as above, but all subvolumes undergo local histogram equalization

BIL3DCNN3.0.py
Same as above, but with both intensity clipping and histogram equalization

Data Extraction

CNNinputDataExtractionV2.py
Extracts positive and negative example subvolumes
Requires:
- dictionaries of image data for each scan
- noduleDimensions excel sheet

Produces:
- List of 1160 negative subvolumes (2 from each scan, chosen randomly)
- List of 775 positive subvolumes (1 centered on each known nodule)
- List of Series IDs set aside for validation
- List of nodules where the coordinate could not be found in the image dictionary

- List of series IDs where a nodule coordinate couldn’t be found in the image dictionary

CNNinputDataExtractionV3.py
Extracts augmented positive and negative subvolumes
Requires: same as above
Produces:
- List of 5800 negative subvolumes (10 from each scan, chosen randomly)
- List of 5425 positive subvolumes (7 per nodule - translated, rotated, and reflected)
- List of Series IDs set aside for validation
- List of nodules where the coordinate could not be found in the image dictionary
- List of series IDs where a nodule coordinate couldn’t be found in the image dictionary

SlidingPositives.py
Extracts all sliding-box style subvolumes
Requires: same as above
Produces: List of all sliding-box style subvolumes from training scans that contain the center of a nodule

TrainInputExtraction2.0.py
Extracts all sliding-box style negatives from a list of scans
Requires:
- List of series IDs (known to be valid training scans)
- Dictionaries of image data

Produces:
- List of all sliding-box style negative subvolumes
- Number of slices in each scan

FPCounter.py
Applies CNN to set of negative subvolumes to obtain the set of false positives
Requires:
- Model file of a trained CNN
- Dataset of negative subvolumes

Produces:
- List of false positive subvolumes
- List of predicted probabilities for each FP subvolume
- Prints number of FPs

Training CNN

IC3DCNN4.2.py
Creates and trains network
Requires: subvolumes for training / testing data

Produces: trained 3D CNN model, weight files from points with lowest test loss

Evaluating Performance

ValClippedInputDataExtraction.py
Creates and clips sliding-box inputs for validation scans
Requires:

- List of validation series IDs
- Dictionaries of image data for each scan
- noduleDimensions excel sheet

Produces:

- All sliding-box subvolumes for validation scans
- noduleBoxes dictionary - for each scan, for each nodule, lists coordinates of all sliding-box subvolumes overlapping that nodule
- fakeNoduleBoxes dictionary (same thing as noduleBoxes but for low-certainty nodules)
- List of number of slices in each scan

List of validation scans IDs that are broken (nodule coordinates don't match dictionary) and working (all coordinates match)

Wholescanapplication2.1.py
Evaluates CNN on validation scans + makes FROC curves
Requires:

- Model file of a trained CNN
- noduleBoxes and fakeNoduleBoxes dictionaries
- List of validation series IDs
- List of number of slices in each series ID
- List of probability thresholds

Produces:

- List of sensitivities (for certain nodules and all nodules), FP rates, adjusted FP rates, number of TP subvolumes, number of FP subvolumes, and number of nodules detected (for both certain and all nodules) at each threshold
- FROC curves for certain nodules, all nodules, and certain nodules w/ adjusted FP rates

FPHistogramExtraction3.0.py
Generates data necessary for heatmaps / histograms
Requires:

- Model file of a trained CNN
- noduleBoxes and fakeNoduleBoxes dictionaries
- List of validation series IDs
- List of number of slices in each series ID

Produces: Parallel lists of box coordinates and their prediction values for FPs, all TPs, and only high-certainty TPs

Heatmap Generation.ipynb
Generates heatmaps
Requires: Parallel lists of box coordinates and their prediction values
Produces: histograms and heatmaps of box locations, filtered by prediction, as desired by user

Combined FROC plots.ipynb
Creates graph of multiple FROC curves
Requires:

For each FROC curve:

- List of sensitivities at various thresholds
- List of FP rates at various thresholds
- List of adjusted FP rates at various thresholds

Produces: 2 graphs (1 adjusted, 1 not adjusted), each containing several FROC curves