

## Appendix 1

### A. PyRadiomics

It is an open-source python package for the extraction of Radiomics features from medical imaging. This package provides a tested and maintained open-source platform for easy and reproducible Radiomic Feature extraction. The platform supports both the feature extraction in 2D and 3D and can be used to calculate single values per feature for a region of interest (“segment-based”) or to generate feature maps (“voxel-based”).

If the python version is 3.5, 3.6 or 3.7(64-bits), the pyradiomics can install by  
conda install -c radiomics pyradiomics

Then, instantiate the ‘featureextractor’ method with ‘extractor = featureextractor.RadiomicsFeatureExtractor()’. After that, used ‘result = extractor.execute(imagePath, labelPath)’ to automatically extract all radiomics features. Where, ‘imagePath’ represents the original image path, and ‘labelPath’ represents the binary image path of the lesion area.

A process of feature extraction as follows:

```
from radiomics import featureextractor
# This module is used for interaction with pyradiomics

# Store the file paths of our testing image and label map into two variables
imagePath = "..\IMG-0015-00116.dcm";
labelPath = "..\116.dcm";
extractor = featureextractor.RadiomicsFeatureExtractor(label= .....)
result = extractor.execute(imagePath, labelPath)

with open('.....\116.csv', 'w') as f:
    [f.write('{0},{1}\n'.format(key, value)) for key, value in result.items()]
```

If you want to extract 3D region features, it is recommended to build the original image and annotation region files into .nrrd files respectively.

## B. Contents of the params.yaml file

setting:

binWidth: 25

label: 1

interpolator: 'sitkBSpline' # This is an enumerated value, here None is not allowed

resampledPixelSpacing: # This disables resampling, as it is interpreted as None, to enable it, specify spacing in x, y, z as [x, y, z]

weightingNorm: # If no value is specified, it is interpreted as None

# Image types to use: "Original" for unfiltered image, for possible filters, see documentation.

# Some feature extraction requirements are added here

imageType:

Original: {} # for dictionaries / mappings, None values are not allowed, '{}' is interpreted as an empty dictionary

LoG: {}

Wavelet: {}

Square: {}

SquareRoot: {}

Logarithm: {}

Exponential: {}

Gradient: {}

LBP3D: {}

# Featureclasses, from which features must be calculated. If a featureclass is not mentioned, no features are calculated

# for that class. Otherwise, the specified features are calculated, or, if none are specified, all are calculated (excluding redundant/deprecated features).

featureClass:

# redundant Compactness 1, Compactness 2 an Spherical Disproportion features are disabled by default, they can be

# enabled by specifying individual feature names (as is done for glm) and including them in the list.

shape:

shape2D:

firstorder: [] # specifying an empty list has the same effect as specifying nothing.

glcm: # Disable SumAverage by specifying all other GLCM features available

glrlm: # for lists none values are allowed, in this case, all features are enabled

glszm:

gldm: # contains deprecated features, but as no individual features are specified, the deprecated features are not enabled

ngtdm:

### C. feature classes and filters

1. First Order Features ----First-order statistics describe the distribution of voxel intensities within the image region defined by the mask through commonly used and basic metrics.

2. Shape Features (3D)----In this group of features we included descriptors of the three-dimensional size and shape of the ROI. These features are independent from the gray level intensity distribution in the ROI and are therefore only calculated on the non-derived image and mask.

3. Gray Level Co-occurrence Matrix (GLCM) Features-----A Gray Level Co-occurrence Matrix (GLCM) of size  $N_g \times N_g$  describes the second-order joint probability function of an image region constrained by the mask and is defined as  $P(i,j|\delta,\theta)$ . The  $(i,j)$ th element of this matrix represents the number of times the combination of levels  $i$  and  $j$  occur in two pixels in the image, that are separated by a distance of  $\delta$  pixels along angle  $\theta$ . Pyradiomics by default computes symmetrical GLCM

4. Gray Level Size Zone Matrix (GLSZM) Features-----A Gray Level Size Zone (GLSZM) quantifies gray level zones in an image. A gray level zone is defined as a the number of connected voxels that share the same gray level intensity.

5. Gray Level Run Length Matrix (GLRLM) Features-----A Gray Level Run Length Matrix (GLRLM) quantifies gray level runs, which are defined as the length in number of pixels, of consecutive pixels that have the same gray level value. In a gray level run length matrix  $P(i,j|\theta)$ , the  $(i,j)$ th element describes the number of runs with gray level  $i$  and length  $j$  occur in the image (ROI) along angle  $\theta$ .

6. Neighbouring Gray Tone Difference Matrix (NGTDM) Features-----A Neighbouring Gray Tone Difference Matrix quantifies the difference between a gray value and the

average gray value of its neighbours within distance  $\delta$ . The sum of absolute differences for gray level  $i$  is stored in the matrix.

7. Gray Level Dependence Matrix (GLDM) Features-----A Gray Level Dependence Matrix (GLDM) quantifies gray level dependencies in an image. A gray level dependency is defined as the number of connected voxels within distance  $\delta$  that are dependent on the center voxel.

Aside from the feature classes, there are also some built-in optional filters:

1. Laplacian of Gaussian (LoG, based on SimpleITK functionality)-----Applies a Laplacian of Gaussian filter to the input image and yields a derived image for each sigma value specified. A Laplacian of Gaussian image is obtained by convolving the image with the second derivative (Laplacian) of a Gaussian kernel.
2. Wavelet (using the PyWavelets package)-----Applies wavelet filter to the input image and yields the decompositions and the approximation.
3. Square-----Computes the square of the image intensities.
4. Square Root----- Computes the square root of the absolute value of image intensities.
5. Logarithm----- Computes the logarithm of the absolute value of the original image +
6. Exponential----- Computes the exponential of the original image.
7. Gradient----- Compute and return the Gradient Magnitude in the image.
8. Local Binary Pattern (3D)----- Compute and return the Local Binary Pattern (LBP) in 3D using spherical harmonics.

the features extracted in the study as follows:

14 original\_shape features; 18 original\_firstorder features; 24 original\_glcmm features; 16 original\_glrmm features; 16 original\_glszm features; 14 original\_gldm features; 5 original\_ngtdm features; 744 wavelet and subclass filters features; 93 square filters features; 93 squareroot filters features; 93 logarithm filters features; 93 exponential filters features; 93 gradient filters features; 279 lbp-3D and subclass filters features; 5 diagnostics\_Image/Mask features.

**Table S1** F-test results (P value) of patient information in three data sets

Features	Training-testing	Training-validation	Testing-validation
Age (years)	0.8099	0.1965	0.1318
Gender	0.0764	<0.05	0.7138
Segment	0.4943	0.9842	0.5132
Margin	<0.05	<0.05	<0.05
Lobulation	0.4682	0.06544	<0.05
Spiculation	0.3168	<0.05	<0.05
Pleural attachment	0.8173	0.07822	0.1251
Air bronchogram	0.9824	0.2638	0.258
Vessel change	0.7454	<0.05	<0.05
Bubble lucency	<0.05	0.6105	<0.05
Average major axis (mm)	<0.05	<0.05	<0.05
Average minor axis (mm)	<0.05	<0.05	<0.05

**Table S2** Comparison of nomogram performance with other studies

Origin of method	Training set	Testing set	Validation set
Paper 1	0.831	0.792	0.833
Paper 2	0.84	0.76	0.79
Paper 3	0.885	0.808	–
Our study	0.877	0.893	0.851