SUPPLEMENTAL MATERIAL

Supplemental Methods 1

Image quality measurements

To assess image quality, we measured the signal-to-noise ratio defined as the mean coronary luminal CT attenuation in Hounsfield units (HU) adjacent to the plaque in a healthy segment divided by the standard deviation of the CT attenuation in the aorta measured in a region of interest at least 2 cm^2 at the level of the left main trunk. Contrast-to-noise ratio was calculated as the mean luminal HU minus the perivascular HU at the site of the plaque divided by the standard deviation of the aortic HU. All measurements were performed on a clinical workstation (IntelliSpace portal, Philips Healthcare, Best, The Netherlands). Detailed information regarding image quality can be found in *table 2*.

Image acquisition

Images were acquired using 256-slice scanner (Brilliance iCT 256, Philips Healthcare, Best, The Netherlands) with prospective ECG-triggered acquisition mode. If the initial heart rate was above 65 beats per minutes we administered heart rate lowering medication (beta blocker or ivabradine, if beta blocker was contraindicated) orally and intravenous to the patients. To ensure optimal visualization of the coronary vessels 0.8 mg of sublingual nitroglycerin was given to all patients 2 minutes before the image acquisition. Images were acquired in cranio-caudal direction during a single breath-hold in inspiration. Four-phasic injection protocol with 90-100 ml of Iomeprol contrast agent was used (Iomeron 400, Bracco Ltd, Milan, Italy) for the coronary CTA examinations.[1] Examinations were performed using 128×0.625 mm detector collimation, 270 ms gantry rotation time, 120 kV, mAs 250-300 depending on patient's body mass index and chest size. All images were reconstructed to a 512×512 matrix with a slice thickness of 0.8 mm and 0.4 mm spacing between slices using an iterative image reconstruction algorithm ($iDOSE^4$ level 5, Philips Healthcare, Best, The Netherlands).

Calculation of radiomic features

Using Radiomics Image Analysis (RIA) software package, we calculated 44 first-order statistics, 3585 gray level co-occurrence matrix (GLCM) based parameters, 55 gray level run length matrix (GLRLM) based metrics and 756 geometry based statistics. For first-order statistics 3D arrays containing the HU values were transformed to a 1D vector, from which the statistics were calculated. For GLCM, GLRLM and geometry based analysis images were discretized by dividing the voxel values into 2, 4, 8, 16 and 32 equally probable bins each containing the same number of voxels. This resulted in 5 replicas of the images. The different bin sizes significantly affect the calculated radiomic feature values. Fewer bins mean more robust values, however result in information loss, while more bins are susceptible to noise, but preserve more information.[2] We conducted our analysis hypothesis free, in a data driven manner by calculating statistics for each discretized image.

GLCM calculations were done based on the concept proposed by Halarick et al.[3] GLCM are matrices, where the element in the i^{th} row and j^{th} column represents the probability of finding a voxel with value j next to a voxel of value i in a given direction and distance. Each statistic was calculated for each of the 26 possible directions in 3D space and then averaged to receive rotationally independent measures. All statistics were calculated for distances 1, 2 and 3 voxels.

GLRLM calculations were done as proposed by Galloway.[4] In the GLRLM matrix the element in the i^{th} row and j^{th} column represents how many times i value voxels occur next to each other j times in a given direction. Each statistic was calculated for each possible run direction in 3D space and then averaged to obtain rotationally independent measures.

Geometry-based statistics were done on raw data as well as discretized images. Surfaces, volumes and radiomic parameters were calculated from the dimensions of the raw image, where the voxels in-plane dimensions were equal to pixel spacing, while the cross-plane dimension was equal to the spacing between the slices. Fractal dimensions were calculated by padding the lesion into an isovolumetric cube with sides equal to the next greatest power of two of the longest dimension of the lesion. Consecutively smaller and smaller cubes were used to cover the lesion and calculate the given statistic. Detailed description of statistical parameters can be found in *supplemental methods 2*.

Supplemental Methods 2

Radiomic features calculated using Radiomics Image Analysis (RIA) Toolbox for Grayscale Images

First-order statistics

First-order statistics discard all spatial information and analyze the data points only considering their values.

For all proceeding first-order statistics let:

x: ordered data points from smallest to largest

x_i: i^th data point, indexing starts from 1

n: number of elements in *x*

Statistics describing the average and spread of the data

MEAN

$$\frac{1}{n}\sum_{i=1}^{n} x_i$$

MEDIAN

$$\begin{cases} x_{ceil(\frac{n}{2})} & x : odd \\ \frac{x_{\frac{n}{2}} + x_{\frac{n}{2}+1}}{2} & x : even \end{cases}$$

MODE

Most frequent value in a data set

HARMONIC MEAN

For all cases if $x_i = 0$, then $x_i = 1$.

$$\frac{1}{\frac{1}{n}\sum_{i=1}^{n}\frac{1}{x_i}}$$

GEOMETRIC MEAN 1

Since the geometric mean of data sets containing negative numbers is not trivial, different geometric means have been proposed. For all cases if $x_i = 0$, then $x_i = 1$.

$$\exp(mean(\log_2 |x|))$$

GEOMETRIC MEAN 2

$$\exp(mean(\frac{x}{|x|}\log_2|x|))$$

GEOMETRIC MEAN 3

$$\exp(mean(\log_2(x+min(x)+1)))$$

TRIMMED MEAN

If *d = 50%*, then the trimmed mean is also called interquartile mean

$$mean(y) \mid y \in [x_{\frac{d}{2}\%}, x_{100-\frac{d}{2}\%}], d = \%\ to\ discard$$

TRIMEAN

$$\frac{x_{25\%} + 2 * x_{50\%} + x_{75\%}}{4}$$

MEAN ABSOLUTE DEVIATION FROM THE MEDIAN

$$mean(|x - median(x)|)$$

MEDIAN ABSOLUTE DEVIATION FROM THE MEDIAN

$$median(|x - median(x)|)$$

MEAN ABSOLUTE DEVIATION FROM THE MEAN

$$mean(|x - mean(x)|)$$

MEDIAN ABSOLUTE DEVIATION FROM THE MEAN

$$median(|x - mean(x)|)$$

MEDIAN ABSOLUTE DEVIATION (MAD)

$$median(|x - median(x)|) * 1.4826$$

MAXIMUM ABSOLUTE DEVIATION FROM THE MEDIAN

$$max(|x - median(x)|)$$

MAXIMUM ABSOLUTE DEVIATION FROM THE MEAN

$$max(|x - mean(x)|)$$

ROOT MEAN SQUARE (RMS)

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}$$

MINIMUM

Lowest value in a data set

MAXIMUM

Highest value in a data set

QUARTILES

$$x_{25\%}, x_{75\%},$$

INTERQUARTILE RANGE (IQR)

$$abs(x_{75\%} - x_{25\%})$$

LOWER-NOTCH

$$x_{25\%} - 1.5 * IQR$$

UPPER-NOTCH

$$x_{75\%} + 1.5 * IQR$$

DECILES

$$x_{10\%}, x_{30\%}, x_{30\%}, x_{40\%}, x_{50\%}, x_{60\%}, x_{70\%}, x_{80\%}, x_{90\%}$$

RANGE

$$abs(max(x) - min(x))$$

Statistics describing the shape of the distribution of data points

VARIANCE

$$\frac{1}{n}\sum_{i=1}^{n}(x_i - mean(x))^2$$

STANDARD DEVIATION (SD)

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - mean(x))^2}$$

SKEWNESS

$$\frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - mean(x))^3}{SD(x)^3}$$

KURTOSIS

$$\frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - mean(x))^4}{SD(x)^4} - 3$$

Statistics describing the diversity of the data points

ENERGY

$$\sum_{i=1}^{n} x_i^2$$

UNIFORMITY

$$\sum_{i=1}^{n} p(x_i)^2$$

ENTROPY

$$\sum_{i=1}^{n} -p(x_i)\log_2 p(x_i)$$

Gray level co-occurrence matrices (GLCM)

Many statistics calculated from GLCMs are a function (*f*) of the elements in the GLCM (*glcm*) matrix multiplied by a weighing matrix (*w*). Using mathematical notation, we can write:

$$w*f(glcm)$$

These modified values are then summed to receive the statistic. By choosing different weights and functions, we can emphasize specific elements of the *glcm* over others, depending on what attribute of heterogeneity we wish to highlight. Basic concepts which help to understand the information stored in the *glcm* are:

- *glcm[i,j]:* the probability of a value *j* occurring next to value *i* at a given angle and direction.
- The main diagonal elements of the glcm store the probabilities of identical voxel occurring next to each other at given distance and direction.
- The further away we move perpendicular to the main diagonal we receive probabilities of voxel occurring next to each other with increasingly different values.
- The upper left quadrant of the matrix holds probabilities of low attenuations voxels occurring next to each other.
- The lower left and the upper right quadrant of the matrix hold probabilities of low attenuations voxels occurring next to high attenuation voxels.
- The lower right quadrant of the matrix holds probabilities of high attenuations voxels occurring next to each other.

For all proceeding glcm statistics let:

g: the number of gray levels the image has been discretized into

g_l: the values of the discretized gray levels, usually g_l = [1, g]

glcm: the gray level co-occurrence matrices matrix, with *g* number of rows and columns

f: function of the elements in the *glcm*

w: the weighing matrix, with *g* number of rows and columns

i: the i^{th} row

j: the j^{th} row

For all calculated statistics the following functions of the glcm are considered:

f(x)=x: *glcm* is unchanged

$f(x)=x^2$: all elements of the *glcm* are squared

$f(x)=-x\log_2(x)$: elements of the *glcm* are replaced by entropy

The following *glcm* matrix is used for calculations:

$$glcm = \begin{bmatrix} 0.14 & 0.07 & 0.03 & 0.01 \\ 0.07 & 0.08 & 0.06 & 0.04 \\ 0.03 & 0.06 & 0.06 & 0.06 \\ 0.01 & 0.04 & 0.06 & 0.18 \end{bmatrix}$$

For all statistics, the *w* matrix is given.

CONTRAST

$$w_{ij} = (i-j)^2 \qquad w = \begin{bmatrix} 0 & 1 & 4 & 9 \\ 1 & 0 & 1 & 4 \\ 4 & 1 & 0 & 1 \\ 9 & 4 & 1 & 0 \end{bmatrix}$$

Contrast gives higher weights in cases where the neighboring voxels have different values. The higher the *Contrast* of an image, the bigger the differences in voxel values of neighboring voxels.

HOMOGENEITY^2

$$w_{ij} = \frac{1}{(i-j)^2+1} \qquad w = \begin{bmatrix} \frac{1}{1} & \frac{1}{2} & \frac{1}{5} & \frac{1}{10} \\ \frac{1}{2} & \frac{1}{1} & \frac{1}{2} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{2} & \frac{1}{1} & \frac{1}{2} \\ \frac{1}{10} & \frac{1}{5} & \frac{1}{2} & \frac{1}{1} \end{bmatrix}$$

Homogeneity^2 is the counterpart of *Contrast*. It takes the same weights, but takes the reciprocal value of them. Therefore, higher weights are given to elements close to the main diagonal, which decreases perpendicular to the main diagonal. The higher the *Homoheneity^2* of an image, the more similar voxels are next to each other.

HOMOGENEITY^2 NON-DIAGONAL

$$w_{ij} = \frac{1}{(i-j)^2+1} \qquad w = \begin{bmatrix} 0.0 & 0.5 & 0.2 & 0.1 \\ 0.5 & 0.0 & 0.5 & 0.2 \\ 0.2 & 0.5 & 0.0 & 0.5 \\ 0.1 & 0.2 & 0.5 & 0.0 \end{bmatrix}$$

Homogeneity^2 non-diagonal is similar to *Homogeneity2* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

DISSIMILARITY

$$w_{ij} = |i-j| \qquad w = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{bmatrix}$$

Dissimilarity gives higher weights in cases where the neighboring voxels have different values. It differs from *Contrast*, in that the weights grow linearly perpendicular to the main diagonal, as opposed to *Contrast*, where the weights grow as a quadratic function.

HOMOGENEITY

$$w_{ij} = \frac{1}{|i-j|+1} \qquad w = \begin{bmatrix} \frac{1}{1} & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{1} & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{2} & \frac{1}{1} & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{3} & \frac{1}{2} & \frac{1}{1} \end{bmatrix}$$

Homogeneity is the counterpart of *Dissimilarity*. It takes the same weights, but takes the reciprocal value of them. Therefore, higher weights are given to elements close to the main diagonal, which decreases perpendicular to the main diagonal. It differs from *Homogeneity^2*, in that the weights decrease linearly perpendicular to the main diagonal, as opposed to *Contrast*, where the weights decrease as a quadratic function.

HOMOGENEITY NON-DIAGONAL

$$w_{ij} = \frac{1}{|i-j|+1} \qquad w = \begin{bmatrix} 0.00 & 0.50 & 0.33 & 0.25 \\ 0.50 & 0.00 & 0.50 & 0.33 \\ 0.33 & 0.50 & 0.00 & 0.50 \\ 0.25 & 0.33 & 0.50 & 0.00 \end{bmatrix}$$

Homogeneity non-diagonal is similar to *Homogeneity* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

DIFFERENCE MOMENTUM NORMALIZED (DMN)

$$w_{ij} = \frac{(i-j)^2}{g^2} \qquad w = \begin{bmatrix} 0.00 & 0.06 & 0.25 & 0.56 \\ 0.06 & 0.00 & 0.06 & 0.25 \\ 0.25 & 0.06 & 0.00 & 0.06 \\ 0.56 & 0.25 & 0.06 & 0.00 \end{bmatrix}$$

DMN is very similar to *Contrast*, except in that it normalizes the weights by the square of the number of gray levels in the image. This results in different weights, where they increase at a slower rate further away from the main diagonal, as compared to *Contrast*.

INVERSE DIFFERENCE MOMENTUM NORMALIZED (IDMN)

$$w_{ij} = \frac{1}{\frac{(i-j)^2}{g^2}+1} \qquad w = \begin{bmatrix} 1.00 & 0.94 & 0.80 & 0.64 \\ 0.94 & 1.00 & 0.94 & 0.80 \\ 0.80 & 0.94 & 1.00 & 0.94 \\ 0.64 & 0.80 & 0.94 & 1.00 \end{bmatrix}$$

IDMN is very similar to *Homogeneity*^{2}, except in that it normalizes the weights by square of the number of gray levels in the image. This results in different weights, where they decline at a slower rate further away from the main diagonal, as compared to *Homogeneity*^{2}.

IDMN NON-DIAGONAL

$$w_{ij} = \frac{1}{\frac{(i-j)^2}{g^2}+1} \qquad w = \begin{bmatrix} 0.00 & 0.94 & 0.80 & 0.64 \\ 0.94 & 0.00 & 0.94 & 0.80 \\ 0.80 & 0.94 & 0.00 & 0.94 \\ 0.64 & 0.80 & 0.94 & 0.00 \end{bmatrix}$$

IDMN non diagonal is very similar to *IDMN* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

DIFFERENCE NORMALIZED (DN)

$$w_{ij} = \frac{|i-j|}{g} \qquad w = \begin{bmatrix} 0.00 & 0.25 & 0.50 & 0.75 \\ 0.25 & 0.00 & 0.25 & 0.50 \\ 0.50 & 0.25 & 0.00 & 0.25 \\ 0.75 & 0.50 & 0.25 & 0.00 \end{bmatrix}$$

DN is very similar to *Dissimilarity*, except in that it normalizes the weights by the number of gray levels in the image. This results in different weights, where they increase at a slower rate further away from the main diagonal, as compared to *Dissimilarity*.

INVERSE DIFFERENCE NORMALIZED (IDN)

$$w_{ij} = \frac{1}{\frac{|i-j|}{g}+1} \qquad w = \begin{bmatrix} 1.00 & 0.80 & 0.67 & 0.57 \\ 0.80 & 1.00 & 0.80 & 0.67 \\ 0.67 & 0.80 & 1.00 & 0.80 \\ 0.57 & 0.67 & 0.80 & 1.00 \end{bmatrix}$$

IDN is very similar to *Homogeneity*, except in that it normalizes the weights by the number of gray levels in the image. This results in different weights, where they decline at a slower rate further away from the main diagonal, as compared to *Homogeneity*.

INVERSE DIFFERENCE NORMALIZED (IDN) NON-DIAGONAL

$$w_{ij} = \frac{1}{\frac{|i-j|}{g} + 1} \qquad w = \begin{bmatrix} 0.00 & 0.80 & 0.67 & 0.57 \\ 0.80 & 0.00 & 0.80 & 0.67 \\ 0.67 & 0.80 & 0.00 & 0.80 \\ 0.57 & 0.67 & 0.80 & 0.00 \end{bmatrix}$$

IDN non-diagonal is very similar to *IDN* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

AUTOCORRELATION

$$w_{ij} = ij \qquad w = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 6 & 8 \\ 3 & 6 & 9 & 12 \\ 4 & 8 & 12 & 16 \end{bmatrix}$$

Autocorrelation uses weights which increase in the direction of the lower right quadrant, therefore emphasizing the lower right quadrant of the *glcm*, where we have the probabilities of high intensity value voxels occurring next to similarly high value voxels.

AUTOCORRELATION NON-DIAGONAL

$$w_{ij} = ij \qquad w = \begin{bmatrix} 0 & 2 & 3 & 4 \\ 2 & 0 & 6 & 8 \\ 3 & 6 & 0 & 12 \\ 4 & 8 & 12 & 0 \end{bmatrix}$$

Autocorrelation non-diagonal is very similar to *Autocorrelation* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE AUTOCORRELATION

$$w_{ij} = \frac{1}{ij} \qquad w = \begin{bmatrix} \frac{1}{1} & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{6} & \frac{1}{8} \\ \frac{1}{3} & \frac{1}{6} & \frac{1}{9} & \frac{1}{12} \\ \frac{1}{4} & \frac{1}{8} & \frac{1}{12} & \frac{1}{16} \end{bmatrix}$$

Inverse autocorrelation is the counterpart of *autocorrelation*, it uses weights which are the reciprocal value of the *autocorrelation* weights and thus increase in the direction of the upper left quadrant, therefore emphasizing the upper left quadrant of the *glcm*, where we have the probabilities of low intensity value voxels occurring next to similarly low value voxels.

INVERSE AUTOCORRELATION NON-DIAGONAL

$$w_{ij} = \frac{1}{ij} \qquad w = \begin{bmatrix} 0.00 & 0.50 & 0.33 & 0.25 \\ 0.50 & 0.00 & 0.17 & 0.12 \\ 0.33 & 0.17 & 0.00 & 0.08 \\ 0.25 & 0.12 & 0.08 & 0.00 \end{bmatrix}$$

Inverse autocorrelation non-diagonal is very similar to *Inverse autocorrelation* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

GAUSSIAN

$$w_{ij} = exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.26 & 0.47 & 0.47 & 0.26 \\ 0.47 & 0.86 & 0.86 & 0.47 \\ 0.47 & 0.86 & 0.86 & 0.47 \\ 0.26 & 0.47 & 0.47 & 0.26 \end{bmatrix}$$

Gaussian uses a 2 dimensional Gaussian distribution as weights. Elements in the middle of the *glcm* which represent voxels with intermediate values next to each other receive the highest weights. The degree of the weights decreases in all directions exponentially.

GAUSSIAN NON-DIAGONAL

$$w_{ij} = exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 0.47 & 0.47 & 0.26 \\ 0.47 & 0.00 & 0.86 & 0.47 \\ 0.47 & 0.86 & 0.00 & 0.47 \\ 0.26 & 0.47 & 0.47 & 0.00 \end{bmatrix}$$

Gaussian non-diagonal is very similar to *Gaussian* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE GAUSSIAN

$$w_{ij} = \frac{1}{exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)}$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 3.86 & 2.12 & 2.12 & 3.86 \\ 2.12 & 1.16 & 1.16 & 2.12 \\ 2.12 & 1.16 & 1.16 & 2.12 \\ 3.86 & 2.12 & 2.12 & 3.86 \end{bmatrix}$$

Inverse Gaussian uses the reciprocal values of a 2 dimensional Gaussian distribution as weights. Elements in the middle of the *glcm* which represent voxels with intermediate values next to each other receive the smallest weights. The degree of the weights increases in all directions exponentially, therefore elements in the four corners of the *glcm* receive higher weights as compared to the center.

INVERSE GAUSSIAN NON-DIAGONAL

$$w_{ij} = \frac{1}{exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)}$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 2.12 & 2.12 & 3.86 \\ 2.12 & 0.00 & 1.16 & 2.12 \\ 2.12 & 1.16 & 0.00 & 2.12 \\ 3.86 & 2.12 & 2.12 & 0.00 \end{bmatrix}$$

Inverse Gaussian non-diagonal is very similar to *Inverse Gaussian* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

GAUSSIAN LEFT POLAR

$$w_{ij} = exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 1.00 & 0.74 & 0.30 & 0.07 \\ 0.74 & 0.55 & 0.22 & 0.05 \\ 0.30 & 0.22 & 0.09 & 0.02 \\ 0.07 & 0.05 & 0.02 & 0.00 \end{bmatrix}$$

Gaussian left polar uses a 2 dimensional Gaussian distribution as weights similar to the simple *Gaussian*, except that the center of the distribution is in the top felt of the *w* matrix, therefore the probability of low value voxels occurring next to each other is emphasized.

GAUSSIAN LEFT POLAR NON-DIAGONAL

$$w_{ij} = exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 0.74 & 0.30 & 0.07 \\ 0.74 & 0.00 & 0.22 & 0.05 \\ 0.30 & 0.22 & 0.00 & 0.02 \\ 0.07 & 0.05 & 0.02 & 0.00 \end{bmatrix}$$

Gaussian left polar non-diagonal is very similar to the *Gaussian left polar* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE GAUSSIAN LEFT POLAR

$$w_{ij} = \frac{1}{exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)}$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 1.00 & 1.35 & 3.32 & 14.88 \\ 1.35 & 1.82 & 4.48 & 20.09 \\ 3.32 & 4.48 & 11.02 & 49.40 \\ 14.88 & 20.09 & 49.40 & 221.41 \end{bmatrix}$$

Inverse Gaussian left polar uses the reciprocal values of a 2 dimensional Gaussian distribution as weights. It is very similar to the *Inverse Gaussian*, except that the center of the distribution is in the top left, therefore elements in the top left of the *glcm* which represent voxels with low values next to each other receive the smallest weights.

INVERSE GAUSSIAN LEFT POLAR NON-DIAGONAL

$$w_{ij} = \frac{1}{exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)}$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 1.35 & 3.32 & 14.88 \\ 1.35 & 0.00 & 4.48 & 20.09 \\ 3.32 & 4.48 & 0.00 & 49.40 \\ 14.88 & 20.09 & 49.40 & 0.00 \end{bmatrix}$$

Inverse Gaussian left polar non-diagonal is very similar to *Inverse Gaussian left polar* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

GAUSSIAN LEFT FOCUS

$$w_{ij} = exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.86 & 0.86 & 0.47 & 0.14 \\ 0.86 & 0.86 & 0.47 & 0.14 \\ 0.47 & 0.47 & 0.26 & 0.08 \\ 0.14 & 0.14 & 0.08 & 0.02 \end{bmatrix}$$

Gaussian left focus uses a 2 dimensional Gaussian distribution as weights similar to the simple *Gaussian*, except that the center of the distribution is in the middle of the upper left quadrant of the *w* matrix, therefore the probability of low-intermediate value voxels occurring next to each other is emphasized.

GAUSSIAN LEFT FOCUS NON-DIAGONAL

$$w_{ij} = exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 0.86 & 0.47 & 0.14 \\ 0.86 & 0.00 & 0.47 & 0.14 \\ 0.47 & 0.47 & 0.00 & 0.08 \\ 0.14 & 0.14 & 0.08 & 0.00 \end{bmatrix}$$

Gaussian left focus non-diagonal is very similar to the *Gaussian left focus* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE GAUSSIAN LEFT FOCUS NON-DIAGONAL

$$w_{ij} = \frac{1}{exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)}$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 1.16 & 2.12 & 7.03 \\ 1.16 & 0.00 & 2.12 & 7.03 \\ 2.12 & 2.12 & 0.00 & 12.81 \\ 7.03 & 7.03 & 12.81 & 0.00 \end{bmatrix}$$

Inverse Gaussian left focus non-diagonal is very similar to *Inverse Gaussian left focus* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE GAUSSIAN LEFT FOCUS

$$w_{ij} = \frac{1}{exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)}$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 1.16 & 1.16 & 2.12 & 7.03 \\ 1.16 & 1.16 & 2.12 & 7.03 \\ 2.12 & 2.12 & 3.86 & 12.81 \\ 7.03 & 7.03 & 12.81 & 42.52 \end{bmatrix}$$

Inverse Gaussian left focus uses the reciprocal values of a 2 dimensional Gaussian distribution as weights. It is very similar to the *Inverse Gaussian*, except that the center of the distribution is in the middle of the upper left quadrant of the *w* matrix, therefore elements in the upper left of the *glcm* which represent voxels with low-intermediate values next to each other receive the smallest weights.

GAUSSIAN RIGHT FOCUS

$$w_{ij} = exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.02 & 0.08 & 0.14 & 0.14 \\ 0.08 & 0.26 & 0.47 & 0.47 \\ 0.14 & 0.47 & 0.86 & 0.86 \\ 0.14 & 0.47 & 0.86 & 0.86 \end{bmatrix}$$

Gaussian right focus uses a 2 dimensional Gaussian distribution as weights similar to the simple *Gaussian*, except that the center of the distribution is in the middle of the lower right quadrant of the *w* matrix, therefore the probability of intermediate-high value voxels occurring next to each other is emphasized.

GAUSSIAN RIGHT FOCUS NON-DIAGONAL

$$w_{ij} = exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 0.08 & 0.14 & 0.14 \\ 0.08 & 0.00 & 0.47 & 0.47 \\ 0.14 & 0.47 & 0.00 & 0.86 \\ 0.14 & 0.47 & 0.86 & 0.00 \end{bmatrix}$$

Gaussian right focus non-diagonal is very similar to the *Gaussian right focus* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE GAUSSIAN RIGHT FOCUS NON-DIAGONAL

$$w_{ij} = \frac{1}{exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)}$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 12.81 & 7.03 & 7.03 \\ 12.81 & 0.00 & 2.12 & 2.12 \\ 7.03 & 2.12 & 0.00 & 1.16 \\ 7.03 & 2.12 & 1.16 & 0.00 \end{bmatrix}$$

Inverse Gaussian right focus non-diagonal is very similar to *Inverse Gaussian right focus* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE GAUSSIAN RIGHT FOCUS

$$w_{ij} = \frac{1}{exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)}$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 42.52 & 12.81 & 7.03 & 7.03 \\ 12.81 & 3.86 & 2.12 & 2.12 \\ 7.03 & 2.12 & 1.16 & 1.16 \\ 7.03 & 2.12 & 1.16 & 1.16 \end{bmatrix}$$

Inverse Gaussian right focus uses the reciprocal values of a 2 dimensional Gaussian distribution as weights. It is very similar to the *Inverse Gaussian*, except that the center of the distribution is in the middle of the lower right quadrant of the *w* matrix, therefore elements in the lower right of the *glcm* which represent voxels with intermediate-high values next to each other receive the smallest weights.

GAUSSIAN RIGHT POLAR

$$w_{ij} = exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 0.02 & 0.05 & 0.07 \\ 0.02 & 0.09 & 0.22 & 0.30 \\ 0.05 & 0.22 & 0.55 & 0.74 \\ 0.07 & 0.30 & 0.74 & 1.00 \end{bmatrix}$$

Gaussian right polar uses a 2 dimensional Gaussian distribution as weights similar to the simple *Gaussian*, except that the center of the distribution is in the lower right of the *w* matrix, therefore the probability of high value voxels occurring next to each other is emphasized.

GAUSSIAN RIGHT POLAR NON-DIAGONAL

$$w_{ij} = exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 0.02 & 0.05 & 0.07 \\ 0.02 & 0.00 & 0.22 & 0.30 \\ 0.05 & 0.22 & 0.00 & 0.74 \\ 0.07 & 0.30 & 0.74 & 0.00 \end{bmatrix}$$

Gaussian right polar non-diagonal is very similar to the *Gaussian right polar* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE GAUSSIAN RIGHT POLAR NON-DIAGONAL

$$w_{ij} = \frac{1}{exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)}$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 49.40 & 20.09 & 14.88 \\ 49.40 & 0.00 & 4.48 & 3.32 \\ 20.09 & 4.48 & 0.00 & 1.35 \\ 14.88 & 3.32 & 1.35 & 0.00 \end{bmatrix}$$

Inverse Gaussian right polar non-diagonal is very similar to *Inverse Gaussian right polar* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE GAUSSIAN RIGHT POLAR

$$w_{ij} = \frac{1}{exp(-\frac{1}{2\sigma^2}(i-\mu)^2) * exp(-\frac{1}{2\sigma^2}(j-\mu)^2)}$$

$$\mu = mean([1,g]) \qquad \sigma = sd([1,g])$$

$$w = \begin{bmatrix} 221.41 & 49.40 & 20.09 & 14.88 \\ 49.40 & 11.02 & 4.48 & 3.32 \\ 20.09 & 4.48 & 1.82 & 1.35 \\ 14.88 & 3.32 & 1.35 & 1.00 \end{bmatrix}$$

Inverse Gaussian right polar uses the reciprocal values of a 2 dimensional Gaussian distribution as weights. It is very similar to the *Inverse Gaussian*, except that the center of the distribution is in the lower right of the *w* matrix, therefore elements in the lower right of the *glcm* which represent voxels with high values next to each other receive the smallest weights.

GAUSSIAN 2 FOCUS

$$w_{ij} = \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-\mu_1)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-\mu_1)^2) + \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-\mu_2)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-\mu_2)^2)$$

$$\mu_1 = mean([1, ceil(g/2)]) \ \mu_2 = mean([floor(g/2), g])$$

$$\sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.88 & 0.94 & 0.61 & 0.28 \\ 0.94 & 1.12 & 0.94 & 0.61 \\ 0.61 & 0.94 & 1.12 & 0.94 \\ 0.28 & 0.61 & 0.94 & 0.88 \end{bmatrix}$$

Gaussian 2 focus uses two Gaussian functions. One is centered in the middle of the upper left quadrant, while the other is centered at the lower right quadrant. The resulting *w* is the sum of the two Gaussians. Elements in the top left and lower right (low value voxels with low value neighbors and high value voxels with high value neighbors) are emphasized over voxels where low value voxels occur next to high value ones

GAUSSIAN 2 FOCUS NON-DIAGONAL

$$w_{ij} = \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-\mu_1)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-\mu_1)^2) + \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-\mu_2)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-\mu_2)^2)$$

$$\mu_1 = mean([1, ceil(g/2)]) \;\; \mu_2 = mean([floor(g/2), g])$$

$$\sigma = sd([1, g])$$

$$w = \begin{bmatrix} 0.00 & 0.94 & 0.61 & 0.28 \\ 0.94 & 0.00 & 0.94 & 0.61 \\ 0.61 & 0.94 & 0.00 & 0.94 \\ 0.28 & 0.61 & 0.94 & 0.00 \end{bmatrix}$$

Gaussian 2 focus non-diagonal is very similar to *Gaussian 2 focus* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE GAUSSIAN 2 FOCUS

$$w_{ij} = \frac{1}{\sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-\mu_1)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-\mu_1)^2)} + \frac{1}{\sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-\mu_2)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-\mu_2)^2)}$$

$$\mu_1 = mean([1, ceil(g/2)]) \;\; \mu_2 = mean([floor(g/2), g])$$

$$\sigma = sd([1, g])$$

$$w = \begin{bmatrix} 43.68 & 13.97 & 9.15 & 14.06 \\ 13.97 & 5.02 & 4.23 & 9.15 \\ 9.15 & 4.23 & 5.02 & 13.97 \\ 14.06 & 9.15 & 13.97 & 43.68 \end{bmatrix}$$

Inverse Gaussian 2 focus uses the reciprocal value of two Gaussian functions. One is centered in the middle of the upper left quadrant, while the other is centered at the lower right quadrant. The resulting *w* is the sum of the two Gaussians. Elements on the perimeter of the matrix are emphasized over values in the middle of the matrix in a way, that elements closer to the main diagonal receive higher weights.

INVERSE GAUSSIAN 2 FOCUS NON-DIAGONAL

$$w_{ij} = \frac{1}{\sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-\mu_1)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-\mu_1)^2)} + \frac{1}{\sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-\mu_2)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-\mu_2)^2)}$$

$$\mu_1 = mean([1, ceil(g/2)]) \;\; \mu_2 = mean([floor(g/2), g])$$

$$\sigma = sd([1, g])$$

$$w = \begin{bmatrix} 0.00 & 13.97 & 9.15 & 14.06 \\ 13.97 & 0.00 & 4.23 & 9.15 \\ 9.15 & 4.23 & 0.00 & 13.97 \\ 14.06 & 9.15 & 13.97 & 0.00 \end{bmatrix}$$

Inverse Gaussian 2 focus non-diagonal is very similar to *Inverse Gaussian 2 focus* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

GAUSSIAN 2 POLAR

$$w_{ij} = \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-1)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-1)^2) + \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-g)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-g)^2)$$

$$\sigma = sd([1, g])$$

$$w = \begin{bmatrix} 1.00 & 0.76 & 0.35 & 0.13 \\ 0.76 & 0.64 & 0.45 & 0.35 \\ 0.35 & 0.45 & 0.64 & 0.76 \\ 0.13 & 0.35 & 0.76 & 1.00 \end{bmatrix}$$

Inverse Gaussian 2 polar uses two Gaussian functions. One is centered in the top left, the other is centered in the bottom right. The resulting *w* is the sum of the two Gaussians. Elements in the top left and lower right (low value voxels with low value neighbors and high value voxels with high value neighbors) are emphasized over voxels where low value voxels occur next to high value ones.

GAUSSIAN 2 POLAR NON-DIAGONAL

$$w_{ij} = \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-1)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-1)^2) + \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-g)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-g)^2)$$

$$\sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 0.76 & 0.35 & 0.13 \\ 0.76 & 0.00 & 0.45 & 0.35 \\ 0.35 & 0.45 & 0.00 & 0.76 \\ 0.13 & 0.35 & 0.76 & 0.00 \end{bmatrix}$$

Inverse Gaussian 2 polar non-diagonal is very similar to *Inverse Gaussian 2 polar* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE GAUSSIAN 2 POLAR

$$w_{ij} = \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-1)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-1)^2) + \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-g)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-g)^2)$$

$$\sigma = sd([1,g])$$

$$w = \begin{bmatrix} 222.41 & 50.75 & 23.41 & 29.76 \\ 50.75 & 12.85 & 8.96 & 23.41 \\ 23.41 & 8.96 & 12.85 & 50.75 \\ 29.76 & 23.41 & 50.75 & 222.41 \end{bmatrix}$$

Inverse Gaussian 2 polar uses the reciprocal value of two Gaussian functions. One is centered in the top left, the other is centered in the bottom right. The resulting *w* is the sum of the two Gaussians. Elements on the perimeter of the matrix are emphasized over values in the middle of the matrix in a way, that elements closer to the main diagonal receive higher weights.

INVERSE GAUSSIAN 2 POLAR NON-DIAGONAL

$$w_{ij} = \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-1)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-1)^2) + \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(i-g)^2) * \sqrt{2\pi}\sigma exp(-\frac{1}{2\sigma^2}(j-g)^2)$$

$$\sigma = sd([1,g])$$

$$w = \begin{bmatrix} 0.00 & 50.75 & 23.41 & 29.76 \\ 50.75 & 0.00 & 8.96 & 23.41 \\ 23.41 & 8.96 & 0.00 & 50.75 \\ 29.76 & 23.41 & 50.75 & 0.00 \end{bmatrix}$$

Inverse Gaussian 2 polar non-diagonal is very similar to *Inverse Gaussian 2 polar* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

CLUSTER PROMINENCE

$$w_{ij} = (i + j - \mu_x(i) - \mu_y(j))^4$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 10.38 & 57.61 & 198.81 & 467.53 \\ 57.61 & 190.47 & 494.23 & 990.49 \\ 198.81 & 494.23 & 1066.76 & 1909.00 \\ 467.53 & 990.49 & 1909.00 & 3172.51 \end{bmatrix}$$

Cluster prominence multiplies the elements of the *glcm* with a *w* matrix where the elements are equal to the values of the two compared voxels, minus the average value we expect next to a *i* value voxel and the average value we expect to a *j* value voxel. This difference is then taken to the fourth power.

CLUSTER PROMINENCE NON-DIAGONAL

$$w_{ij} = (i + j - \mu_x(i) - \mu_y(j))^4$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.00 & 57.61 & 198.81 & 467.53 \\ 57.61 & 0.00 & 494.23 & 990.49 \\ 198.81 & 494.23 & 0.00 & 1909.00 \\ 467.53 & 990.49 & 1909.00 & 0.00 \end{bmatrix}$$

Cluster prominence non-diagonal is very similar to *Cluster prominence* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE CLUSTER PROMINENCE

$$w_{ij} = \frac{1}{(i + j + \mu_x(i) + \mu_y(i))^4}$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.10 & 0.02 & 0.01 & 0.00 \\ 0.02 & 0.01 & 0.00 & 0.00 \\ 0.01 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 \end{bmatrix}$$

Inverse cluster prominence takes the reciprocal value of the weights of *Cluster prominence*.

INVERSE CLUSTER PROMINENCE NON-DIAGONAL

$$w_{ij} = \frac{1}{(i + j + \mu_x(i) + \mu_y(i))^4}$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.00 & 0.02 & 0.01 & 0.00 \\ 0.02 & 0.00 & 0.00 & 0.00 \\ 0.01 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 \end{bmatrix}$$

Inverse cluster prominence non-diagonal is very similar to *Inverse cluster prominence* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

CLUSTER SHADE

$$w_{ij} = (i + j - \mu_x(i) - \mu_y(j))^3$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 5.78 & 20.91 & 52.95 & 100.54 \\ 20.91 & 51.27 & 104.82 & 176.56 \\ 52.95 & 104.82 & 186.66 & 288.80 \\ 100.54 & 176.56 & 288.80 & 422.72 \end{bmatrix}$$

Cluster shade multiplies the elements of the *glcm* with a *w* matrix where the elements are equal to the values of the two compared voxels, minus the average value we expect next to a *i* value voxel and the average value we expect to a *j* value voxel. This difference is then taken to the third power.

CLUSTER SHADE NON-DIAGONAL

$$w_{ij} = (i + j - \mu_x(i) - \mu_y(j))^3$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.00 & 20.91 & 52.95 & 100.54 \\ 20.91 & 0.00 & 104.82 & 176.56 \\ 52.95 & 104.82 & 0.00 & 288.80 \\ 100.54 & 176.56 & 288.80 & 0.00 \end{bmatrix}$$

Cluster shade non-diagonal is very similar to *Cluster shade* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE CLUSTER SHADE

$$w_{ij} = \frac{1}{(i + j + \mu_x(i) + \mu_y(i))^3}$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.17 & 0.05 & 0.02 & 0.01 \\ 0.05 & 0.02 & 0.01 & 0.01 \\ 0.02 & 0.01 & 0.01 & 0.00 \\ 0.01 & 0.01 & 0.00 & 0.00 \end{bmatrix}$$

Inverse cluster shade takes the reciprocal value of the weights of *Cluster shade*.

INVERSE CLUSTER SHADE NON-DIAGONAL

$$w_{ij} = \frac{1}{(i+j+\mu_x(i)+\mu_y(i))^3}$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.00 & 0.05 & 0.02 & 0.01 \\ 0.05 & 0.00 & 0.01 & 0.01 \\ 0.02 & 0.01 & 0.00 & 0.00 \\ 0.01 & 0.01 & 0.00 & 0.00 \end{bmatrix}$$

Inverse cluster shade non-diagonal is very similar to *Inverse cluster shade* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

CLUSTER TENDENCY

$$w_{ij} = (i+j-\mu_x(i)-\mu_y(j))^2$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 3.22 & 7.59 & 14.10 & 21.62 \\ 7.59 & 13.80 & 22.23 & 31.47 \\ 14.10 & 22.23 & 32.66 & 43.69 \\ 21.62 & 31.47 & 43.69 & 56.33 \end{bmatrix}$$

Cluster tendency multiplies the elements of the *glcm* with a *w* matrix where the elements are equal to the values of the two compared voxels, minus the average value we expect next to a *i* value voxel and the average value we expect to a *j* value voxel. This difference is then taken to the second power.

CLUSTER TENDENCY NON-DIAGONAL

$$w_{ij} = (i+j-\mu_x(i)-\mu_y(j))^2$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.00 & 7.59 & 14.10 & 21.62 \\ 7.59 & 0.00 & 22.23 & 31.47 \\ 14.10 & 22.23 & 0.00 & 43.69 \\ 21.62 & 31.47 & 43.69 & 0.00 \end{bmatrix}$$

Cluster tendency non-diagonal is very similar to *Cluster tendency* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE CLUSTER TENDENCY

$$w_{ij} = \frac{1}{(i+j+\mu_x(i)+\mu_y(i))^2}$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.31 & 0.13 & 0.07 & 0.05 \\ 0.13 & 0.07 & 0.04 & 0.03 \\ 0.07 & 0.04 & 0.03 & 0.02 \\ 0.05 & 0.03 & 0.02 & 0.02 \end{bmatrix}$$

Inverse cluster *tendency* takes the reciprocal value of the weights of C*luster tendency*.

INVERSE CLUSTER TENDENCY NON-DIAGONAL

$$w_{ij} = \frac{1}{(i+j+\mu_x(i)+\mu_y(i))^2}$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.00 & 0.13 & 0.07 & 0.05 \\ 0.13 & 0.00 & 0.04 & 0.03 \\ 0.07 & 0.04 & 0.00 & 0.02 \\ 0.05 & 0.03 & 0.02 & 0.00 \end{bmatrix}$$

Inverse cluster *tendency non-diagonal* is very similar to *Inverse cluster tendency* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

CLUSTER DIFFERENCE

$$w_{ij} = |i + j - \mu_x(i) - \mu_y(j)|$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 1.80 & 2.75 & 3.75 & 4.65 \\ 2.75 & 3.71 & 4.71 & 5.61 \\ 3.75 & 4.71 & 5.71 & 6.61 \\ 4.65 & 5.61 & 6.61 & 7.51 \end{bmatrix}$$

Cluster difference multiplies the elements of the *glcm* with a *w* matrix where the elements are equal to the values of the two compared voxels, minus the average value we expect next to a *i* value voxel and the average value we expect to a *j* value voxel.

CLUSTER DIFFERENCE NON-DIAGONAL

$$w_{ij} = |i + j - \mu_x(i) - \mu_y(j)|$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.00 & 2.75 & 3.75 & 4.65 \\ 2.75 & 0.00 & 4.71 & 5.61 \\ 3.75 & 4.71 & 0.00 & 6.61 \\ 4.65 & 5.61 & 6.61 & 0.00 \end{bmatrix}$$

Cluster difference non-diagonal is very similar to *Cluster difference* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

INVERSE CLUSTER DIFFERENCE

$$w_{ij} = \frac{1}{|i + j + \mu_x(i) + \mu_y(i)|}$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.56 & 0.36 & 0.27 & 0.22 \\ 0.36 & 0.27 & 0.21 & 0.18 \\ 0.27 & 0.21 & 0.17 & 0.15 \\ 0.22 & 0.18 & 0.15 & 0.13 \end{bmatrix}$$

Inverse c*luster difference* takes the reciprocal value of the weights of C*luster difference*.

INVERSE CLUSTER DIFFERENCE NON-DIAGONAL

$$w_{ij} = \frac{1}{|i + j + \mu_x(i) + \mu_y(i)|}$$

$$\mu_x(i) = mean(glcm_i * g_l) \quad \mu_y(j) = mean(glcm_j * g_l)$$

$$w = \begin{bmatrix} 0.00 & 0.36 & 0.27 & 0.22 \\ 0.36 & 0.00 & 0.21 & 0.18 \\ 0.27 & 0.21 & 0.00 & 0.15 \\ 0.22 & 0.18 & 0.15 & 0.00 \end{bmatrix}$$

Inverse c*luster difference non-diagonal* is very similar to Inverse c*luster difference* except that the diagonal elements of *w* are 0, therefore same value voxel pairs are not considered in the statistic.

MEAN

$$w_{ij} = g_l$$

$$w = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 \end{bmatrix}$$

Mean is a measure of the average *f(glcm)* values. Since the elements of the *glcm* are symmetrical, therefore calculations based on rows (*i*) are equivalent if calculations were done on columns (*j*).

VARIANCE

$$w_{ij} = (i - \mu)^2 \qquad \mu = mean(glcm * g_l)$$

$$w = \begin{bmatrix} 2.37 & 2.37 & 2.37 & 2.37 \\ 0.29 & 0.29 & 0.29 & 0.29 \\ 0.21 & 0.21 & 0.21 & 0.21 \\ 2.13 & 2.13 & 2.13 & 2.13 \end{bmatrix}$$

Variance is a measure of the variation of the elements in the *glcm*. Since the elements of the *glcm* are symmetrical, therefore calculations based on rows (*i*) are equivalent if calculations were done on columns (*j*).

$$w = \begin{bmatrix} 1.79 & 0.63 & -0.53 & -1.69 \\ 0.63 & 0.22 & -0.19 & -0.59 \\ -0.53 & -0.19 & 0.16 & 0.51 \\ -1.69 & -0.59 & 0.51 & 1.60 \end{bmatrix}$$

CORRELATION

$$w_{ij} = \frac{(i-\mu)*(j-\mu)}{\sigma^2}$$

$$\mu = mean(glcm * g_l) \quad \sigma = sum(glcm * (g_l - \mu)^2)$$

Correlation is a measure of the linear dependency of neighboring voxels. As opposed to previous cases, here the weight matrix is a function of *f(glcm)*, therefore for each statistical measure we have a separate weight matrix.

Previous statistics used different weights for emphasizing specific elements of the *glcm*. The following statistics aggregate the *glcm* values based on some equation to prioritize given *glcm* elements over others.

Sum

Sum statistics groups the *glcm* elements based on which row and column they are in. Values where *i+j* is the same are combined together. This results in aggregating together elements of the *glcm* which are on one-line perpendicular to the main diagonal. This is indicated in the mask matrix (*m*), where same value elements will be grouped together in the *glcm* to calculate the statistic. Each of the statistics takes a function (*f*) of these combined values and multiplies these values with given weights (*we*).

$$m = \begin{bmatrix} 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 \\ 4 & 5 & 6 & 7 \\ 5 & 6 & 7 & 8 \end{bmatrix}$$

$$p_{x+y}(k) = \sum_{i=1}^{g}\sum_{j=1}^{g} glcm_{ij} \quad | \; i+j=k; k \in [2, 2g]$$

SUM AVERAGE

$$we = k \qquad f(x) = x$$

$$\sum_{k=2}^{2g} k p_{x+y}(k)$$

SUM ENERGY

$$we = k \qquad\qquad f(x) = x^2$$

$$\sum_{k=2}^{2g} kp_{x+y}(k)^2$$

SUM ENTROPY

$$we = -p_{x+y}(k) \qquad\qquad f(x) = \log_2 x$$

$$\sum_{k=2}^{2g} -p_{x+y}(k)\log_2(p_{x+y}(k))$$

SUM VARIANCE

$$we = (k - SE)^2\ ;\ SE = Sum\ entropy \qquad\qquad f(x) = x$$

$$\sum_{k=2}^{2g} (k - SE)^2 p_{x+y}(k)$$

Difference

Difference statistics groups the *glcm* elements based on which row and column they are in. Values where |*i*-*j*| is the same are combined together. This results in aggregating together elements of the *glcm* which are parallel to the main diagonal. This is indicated in the mask matrix (*m*), where same value elements will be grouped together in the *glcm* to calculate the statistic. Each of the statistics takes a function (*f*) of these combined values and multiplies these values with given weights (*we*).

$$p_{x-y}(k) = \sum_{i=1}^{g}\sum_{j=1}^{g} glcm_{ij} \quad |\ i - j = k; k \in [0, g-1] \qquad\qquad m = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{bmatrix}$$

DIFFERENCE AVERAGE

$$we = k \qquad\qquad f(x) = x$$

$$\sum_{k=0}^{g-1} kp_{x-y}(k)$$

DIFFERENCE ENERGY

$$we = k \qquad f(x) = x^2$$

$$\sum_{k=0}^{g-1} kp_{x-y}(k)^2$$

DIFFERENCE ENTROPY

$$we = -p_{x-y}(k) \qquad f(x) = \log_2 x$$

$$\sum_{k=0}^{g-1} -p_{x-y}(k)\log_2(p_{x-y}(k))$$

DIFFERENCE VARIANCE

$$we = (k - DE)^2\,;\ DE = Difference\,entropy \qquad f(x) = x$$

$$\sum_{k=0}^{g-1} (k - DE)^2 p_{x-y}(k)$$

Inverse sum

Inverse statistics groups the *glcm* elements based on which row and column they are in. Values where *i+j* is the same are combined together. This results in aggregating together elements of the *glcm* which are on one-line perpendicular to the main diagonal. This is indicated in the mask matrix (*m*), where same value elements will be grouped together in the *glcm* to calculate the statistic. Each of the statistics takes a function (*f*) of these combined values and multiplies these values with given weights (*we*). Inverse sum is similar to sum statistics, except that it uses the reciprocal values of the weights, therefore the opposite elements are emphasized as compared to sum statistics. Entropy does not use weights proportional to the row or column value, it would be equal to sum entropy, therefore it is undefined.

$$p_{x+y}(k) = \sum_{i=1}^{g}\sum_{j=1}^{g} glcm_{ij} \quad \mid i+j = k; k \in [2, 2g] \qquad m = \begin{bmatrix} 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 \\ 4 & 5 & 6 & 7 \\ 5 & 6 & 7 & 8 \end{bmatrix}$$

INVERSE SUM AVERAGE

$$we = \frac{1}{k} \qquad f(x) = x$$

$$\sum_{k=2}^{2g} \frac{p_{x+y}(k)}{k}$$

INVERSE SUM ENERGY

$$we = \frac{1}{k} \qquad f(x) = x^2$$

$$\sum_{k=2}^{2g} \frac{p_{x+y}(k)^2}{k}$$

INVERSE SUM VARIANCE

$$we = \frac{1}{(k-SE)^2}\ ;\ SE = Sum\, entropy \qquad f(x) = x$$

$$\sum_{k=2}^{2g} \frac{p_{x+y}(k)}{(k-SE)^2}$$

Inverse difference

Inverse difference statistics groups the *glcm* elements based on which row and column they are in. Values where |*i-j*| is the same are combined together. This results in aggregating together elements of the *glcm* which are parallel to the main diagonal. This is indicated in the mask matrix (*m*), where same value elements will be grouped together in the *glcm* to calculate the statistic. Each of the statistics takes a function (*f*) of these combined values and multiplies these values with given weights (*we*). Inverse difference is similar to difference statistics, except that it uses the reciprocal values of the weights, therefore the opposite elements are emphasized as compared to sum statistics. Since division by 0 is undefined, main diagonal elements are considered to be 0. Entropy does not use weights proportional to the row or column value, it would be equal to difference entropy, therefore it is undefined.

$$p_{x-y}(k) = \sum_{i=1}^{g}\sum_{j=1}^{g} glcm_{ij} \quad \mid i-j=k; k \in [0, g-1]$$

$$m = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{bmatrix}$$

INVERSE DIFFERENCE AVERAGE

$$we = \frac{1}{k} \qquad f(x) = x$$

$$\sum_{k=0}^{g-1} \frac{p_{x-y}(k)}{k}$$

INVERSE DIFFERENCE ENERGY

$$we = \frac{1}{k} \qquad f(x) = x^2$$

$$\sum_{k=0}^{g-1} \frac{p_{x-y}(k)^2}{k}$$

INVERSE DIFFERENCE VARIANCE

$$we = \frac{1}{(k-DE)^2}\ ;\ DE = Difference\ entropy \qquad f(x) = x$$

$$\sum_{k=0}^{g-1} \frac{p_{x-y}(k)}{(k-DE)^2}$$

Further glcm functions

The following metrics cannot be grouped into either of the previous cases. These metrics are standalone functions of the elements of the *glcm*.

INFORMATION MEASURE OF CORRELATION 1 (IMC1)

$$H = entropy(glcm)$$
$$HX = entropy(p_x)$$
$$HY = entropy(p_y)$$

$$HXY1 = \sum_{i=1}^{g}\sum_{j=1}^{g} -glcm_{ij}\log_2(p_x(i)p_y(j))$$
$$p_x = (row\ marginal\ distribution)$$
$$p_y = (column\ marginal\ distribution)$$

$$IMC1 = \frac{H - HXY1}{max(HX, HY)}$$

INFORMATION MEASURE OF CORRELATION 2 (IMC2)

$$H = entropy(glcm)$$
$$HX = entropy(p_x)$$
$$HY = entropy(p_y)$$

$$HXY2 = \sum_{i=1}^{g} \sum_{j=1}^{g} -p_x(i)p_y(j) \log_2(p_x(i)p_y(j))$$
$$p_x = (row\ marginal\ distribution)$$
$$p_y = (column\ marginal\ distribution)$$

$$IMC2 = \sqrt{1 - e^{-2(HXY2-H)}}$$

ENERGY

$$\sum_{i=1}^{g} \sum_{j=1}^{g} glcm_{ij}^2$$

ENTROPY

$$\sum_{i=1}^{g} \sum_{j=1}^{g} -glcm_{ij} \log_2 glcm_{ij}$$

First-order statistics of GLCM

All GLCMs can be seen as an array of probability values, and therefore first-order statistics can be used to describe different aspects of the distribution.

Gray level run length matrix (GLRLM)

Many statistics calculated from GLRLMs are a sum of: the elements in the GLRLM (*glrlm*) matrix multiplied by a weighing matrix (*w*). Using mathematical notation, we can write:

w*glrlm

By choosing different weights, we can emphasize specific elements of the *glrlm* over others, depending on what attribute of the run lengths we wish to highlight. Basic concepts which help to understand the information stored in the *glrlm* are:

- *glrlm[i,j]:* the number of times *i* value voxels are next to each other *j* times
- The first column stores the number of times voxels do not have same value neighbors
- The upper left quadrant of the matrix holds frequencies of how many times low attenuation voxels have few same value neighbors
- The lower left quadrant of the matrix stores frequencies of how many times high attenuation voxels have few same value neighbors
- The upper right quadrant of the matrix holds frequencies of how many times low attenuation voxels have many same value neighbors
- The lower right quadrant of the matrix stores frequencies of how many times high attenuation voxels have many same value neighbors

For all proceeding glcm statistics let:

dim: the maximum number of voxels present in the given direction

g: the number of gray levels the image has been discretized into

glrlm: the gray level run length matrix, with *g* number of rows and *dim* number columns

w: the weighing matrix, with *g* number of rows and *dim* number columns

i: the i^{th} row

j: the j^{th} row

n_r: number of run lengths

n_v: number of voxels

For all statistics, the examples will be given for the following 4x5 *glrlm* matrix

$$glrlm = \begin{bmatrix} 25 & 16 & 11 & 7 & 7 \\ 105 & 20 & 13 & 5 & 2 \\ 122 & 27 & 8 & 2 & 1 \\ 124 & 25 & 10 & 3 & 0 \end{bmatrix}$$

To achieve comparable results between different images, the results can be divided by n_r, which is a normalizing factor.

For all statistics, the *w* matrix is given.

Weighed matrix statistics

SHORT RUN EMPHASIS (SRE)

$$w = \frac{1}{j^2}$$

$$w = \begin{bmatrix} 1 & 0.25 & 0.11 & 0.06 & 0.04 \\ 1 & 0.25 & 0.11 & 0.06 & 0.04 \\ 1 & 0.25 & 0.11 & 0.06 & 0.04 \\ 1 & 0.25 & 0.11 & 0.06 & 0.04 \end{bmatrix}$$

SRE gives higher weights to short run lengths, therefore images where intensity values change quickly in the given direction have higher values, while images with many same value voxels next to each other receive lower values.

LONG RUN EMPHASIS (LRE)

$$w = j^2$$

$$w = \begin{bmatrix} 1 & 4 & 9 & 16 & 25 \\ 1 & 4 & 9 & 16 & 25 \\ 1 & 4 & 9 & 16 & 25 \\ 1 & 4 & 9 & 16 & 25 \end{bmatrix}$$

LRE gives higher weights to long run lengths, therefore images where intensity values change slowly in the given direction have higher values, while images with many different value voxels next to each other receive lower values.

LOW GRAY LEVEL RUN EMPHASIS (LGLRE)

$$w = \frac{1}{i^2}$$

$$w = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0.25 & 0.25 & 0.25 & 0.25 & 0.25 \\ 0.11 & 0.11 & 0.11 & 0.11 & 0.11 \\ 0.06 & 0.06 & 0.06 & 0.06 & 0.06 \end{bmatrix}$$

LGLRE gives higher weights low value voxels, therefore images with predominantly low attenuation values will receive higher values as compared to images with higher attenuation voxels.

HIGH GRAY LEVEL RUN EMPHASIS (HGLRE)

$$w = i^2$$

$$w = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 4 & 4 & 4 & 4 & 4 \\ 9 & 9 & 9 & 9 & 9 \\ 16 & 16 & 16 & 16 & 16 \end{bmatrix}$$

HGLRE gives higher weights to voxels with high attenuation values, therefore images with predominantly high voxel values will receive higher values as compared to images with lower attenuation voxels.

SHORT RUN LOW GRAY LEVEL EMPHASIS (SRLGLE)

$$w = \frac{1}{i^2 * j^2}$$

$$w = \begin{bmatrix} 1 & 0.25 & 0.11 & 0.06 & 0.04 \\ 0.25 & 0.06 & 0.03 & 0.02 & 0.01 \\ 0.11 & 0.03 & 0.01 & 0.01 & 0.00 \\ 0.06 & 0.02 & 0.01 & 0.00 & 0.00 \end{bmatrix}$$

SRLGLE gives higher weights low value and low run lengths, therefore images with predominantly low attenuation values which do not occur repeatedly will receive higher values as compared to images with higher attenuation voxels frequently occurring next to each other.

LONG RUN HIGH GRAY LEVEL EMPHASIS (LRHGLE)

$$w = i^2 * j^2$$

$$w = \begin{bmatrix} 1 & 4 & 9 & 16 & 25 \\ 4 & 16 & 36 & 64 & 100 \\ 9 & 36 & 81 & 144 & 225 \\ 16 & 64 & 144 & 256 & 400 \end{bmatrix}$$

LRHGLE gives higher weights high value and long run lengths, therefore images with predominantly high attenuation values which occur repeatedly next to each other will receive higher values as compared to images where low attenuation voxels occur randomly next to each other.

SHORT RUN HIGH GRAY LEVEL EMPHASIS (SRHGLE)

$$w = \frac{i^2}{j^2}$$

$$w = \begin{bmatrix} 1 & 0.25 & 0.11 & 0.06 & 0.04 \\ 4 & 1 & 0.44 & 0.25 & 0.16 \\ 9 & 2.25 & 1 & 0.56 & 0.36 \\ 16 & 4 & 1.78 & 1 & 0.64 \end{bmatrix}$$

SRHGLE gives higher weights high value and low run lengths, therefore images with predominantly high attenuation values which do not occur repeatedly will receive higher values as compared to images with lower attenuation voxels frequently occurring next to each other.

LONG RUN LOW GRAY LEVEL EMPHASIS (LRLGLE)

$$w = \frac{j^2}{i^2}$$

$$w = \begin{bmatrix} 1 & 4 & 9 & 16 & 25 \\ 0.25 & 1 & 2.25 & 4 & 6.25 \\ 0.11 & 0.44 & 1 & 1.78 & 2.78 \\ 0.06 & 0.25 & 0.56 & 1 & 1.56 \end{bmatrix}$$

LRLGLE gives higher weights low value and long run lengths, therefore images with predominantly low attenuation values which occur repeatedly will receive higher values as compared to images with higher attenuation voxels which do not occur frequently next to each other.

RUN PERCENTAGE (RP)

$$w = \frac{1}{n_v} \qquad w = \frac{1}{796}$$

RP weighs all elements equally. The more short run lengths there are in the image, the higher the value.

Summed matrix statistics

The following statistics are calculated by summing the values of the *glrlm* either by rows or columns.

GRAY LEVEL NONUNIFORMITY (GLN)

$$\sum_{i=1}^{g}(\sum_{j=1}^{dim} glrlm[i,j])^2$$

GLN first add up the elements of the *glrlm* by row and then squares them and sums them. When runs are equally distributed for all gray levels, then it takes up its minimum.

RUN LENGTH NONUNIFORMITY (RLN)

$$\sum_{j=1}^{dim}(\sum_{i=1}^{g} glrlm[i,j])^2$$

RLN first add up the elements of the *glrlm* by columns and then squares them and sums them. When run lengths for all lengths, then it takes up its minimum

Shape-based metrics

Shape-based measures derive parameters from the geometrical properties of the lesion.

1-, 2-, 3-dimensional statistics

These metrics are calculated from the space occupied by the abnormality

VOLUME (V)

$$dim_{xy} = Pixel\ Spacing$$
$$dim_z = Spacing\ Between\ Slices$$
$$n * dim_{xy}^2 * dim_z$$

VOLUME RATIO

$$\frac{V_{ROI}}{V_{total}}$$

SURFACE (A)

$$dim_{xy} = Pixel\ Spacing$$

$$dim_z = Spacing\ Between\ Slices$$
$$n_{yz} : number\ of\ voxels\ without\ any\ neighboor\ in\ direction\ x$$
$$n_{xz} : number\ of\ voxels\ without\ any\ neighboor\ in\ direction\ y$$
$$n_{xy} : number\ of\ voxels\ without\ any\ neighboor\ in\ direction\ z$$
$$n_{yz} * dim_{xy} * dim_z + n_{xz} * dim_{xy} * dim_z + dim_{xy} * dim_z^2$$

SURFACE RATIO

$$\frac{A_{ROI}}{A_{total}}$$

SURFACE TO VOLUME RATIO

$$\frac{A_{ROI}}{V_{ROI}}$$

COMPACTNESS1

$$\frac{V}{\sqrt{\pi}A^{\frac{2}{3}}}$$

COMPACTNESS2

$$36\pi\frac{V^2}{A^3}$$

SPHERICAL DISPROPORTION

$$\frac{A}{4\pi[(\frac{3V}{4\pi})^{\frac{1}{3}}]^2}$$

SPHERICITY

$$\frac{6V^{\frac{2}{3}}\pi^{\frac{1}{3}}}{A}$$

MAXIMUM DIAMETER

$$x_i y_i z_i : spatial\ coordinate\ of\ voxel\ i$$

$$max(\sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2 + (z_i - z_{i+1})^2}$$

Fractal dimensions

Fractal dimensions enumerate the self-symmetry of an object. The lesions are padded to a isovolumetric cube with sides equal to the next greatest power of two of the longest dimension of the lesion. Smaller and smaller bounding boxes are used to cover the lesion. Limits are approximated by the slope of the regression line through the points at each given scale on a log-log plot.

$p_i: \ normalized\ probability\ of\ a\ voxel\ with\ any\ value\ present\ in\ bounding\ box\ i\ with\ dimension\ \varepsilon$

$\varepsilon: \ the\ number\ of\ boxes\ needed\ to\ cover\ the\ padded\ box\ \ in\ one\ dimension$

BOX-COUNTING DIMENSION

$$\lim_{\epsilon \to \infty} \frac{\log_2 \sum_{i=1}^{\epsilon^3} p_i^0}{\log_2 \epsilon}$$

INFORMATION DIMENSION

$$\lim_{\epsilon \to \infty} \frac{\sum_{i=1}^{\epsilon^3} -p_i \log_2 p_i}{\log_2 \epsilon}$$

CORRELATION DIMENSION

$$\lim_{\epsilon \to \infty} \frac{\log_2 \sum_{i=1}^{\epsilon^3} p_i^2}{\log_2 \epsilon}$$

Correlation dimension is strictly calculated from distances of the data points. A generalization of the Rényi entropy is used to approximate the correlation dimension.

Supplemental Table

Supplemental table 1. Diagnostic performance of radiomic parameters with AUC values above 0.8

Variable	Case	IQR	Control	IQR	p	AUC	95% CI (AUC)	Sensitivity	Specificity	PPV	NPV
First order statistics											
Deciles30__orig	53.50	[36.50; 74.08]	93.70	[75.50; 135.75]	0.00054425	0.827	[0.716; 0.921]	0.833	0.733	0.758	0.815
Quartiles25__orig	40.00	[29.25; 62.06]	82.50	[65.50; 122.00]	0.00062135	0.826	[0.712; 0.922]	0.767	0.800	0.793	0.774
Deciles20__orig	31.00	[15.50; 53.30]	71.00	[56.00; 106.25]	0.00087011	0.826	[0.713; 0.924]	0.800	0.767	0.774	0.793
Har_mean__orig	65.79	[53.74; 80.10]	106.27	[85.37; 141.20]	0.00283237	0.823	[0.708; 0.922]	0.767	0.800	0.793	0.774
Tri_mean__orig	67.88	[47.25; 95.88]	111.00	[88.62; 155.25]	0.00071495	0.812	[0.696; 0.910]	0.867	0.667	0.722	0.833
Deciles40__orig	70.50	[50.50; 99.35]	119.00	[93.75; 165.75]	0.00054393	0.812	[0.695; 0.909]	0.867	0.667	0.722	0.833
Geo_mean__orig	524.51	[342.84; 884.73]	1000.31	[736.51; 1516.67]	0.00160946	0.803	[0.684; 0.902]	0.633	0.900	0.864	0.711
IQ_mean__orig	100.96	[71.20; 131.57]	146.32	[121.76; 190.18]	0.00075437	0.802	[0.684; 0.902]	0.600	0.933	0.900	0.700
GLCM											
IQR__ep_b4_d1_avg	0.05	[0.05; 0.06]	0.04	[0.04; 0.05]	0.00012117	0.867	[0.769; 0.948]	0.700	0.900	0.875	0.750
Low_notch__ep_b4_d1_avg	-0.06	[-0.07; -0.05]	-0.03	[-0.05; -0.01]	0.00012017	0.866	[0.763; 0.948]	0.967	0.633	0.725	0.950
Gauss_rf_s_nd__ep_b16_d3_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00045383	0.859	[0.759; 0.940]	0.767	0.867	0.852	0.788
Md_AD_mn__ep_b4_d1_avg	0.04	[0.03; 0.04]	0.03	[0.02; 0.03]	0.00019997	0.856	[0.744; 0.946]	0.867	0.767	0.788	0.852
Gauss_rf_s__ep_b32_d3_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00134475	0.851	[0.743; 0.936]	0.667	0.933	0.909	0.737
Gauss_rf_s_nd__ep_b32_d3_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00128411	0.849	[0.743; 0.936]	0.600	1.000	1.000	0.714
Sum_energy__ep_b32_d1_avg	0.53	[0.51; 0.54]	0.58	[0.54; 0.62]	0.00006803	0.848	[0.740; 0.937]	0.967	0.633	0.725	0.950
IMC1__ep_b2_d1_avg	-2.23	[-2.27; -2.20]	-2.15	[-2.18; -2.12]	0.00028174	0.847	[0.736; 0.939]	0.933	0.700	0.757	0.913

Autocorrelation_s_nd__ep_b16_d3_avg	0.28	[0.26; 0.34]	0.38	[0.32; 0.51]	0.00045426	0.847	[0.738; 0.931]	0.667	0.933	0.909	0.737
Cluster_t_s__ep_b16_d3_avg	1.42	[1.33; 1.76]	1.96	[1.60; 2.71]	0.00033289	0.847	[0.741; 0.930]	0.667	0.900	0.870	0.730
Gauss_rp_s_nd__ep_b32_d3_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00056330	0.847	[0.740; 0.929]	0.633	0.933	0.905	0.718
Inv_Cluster_d_e_nd__ep_b2_d1_avg	0.31	[0.30; 0.33]	0.35	[0.34; 0.37]	0.00021110	0.846	[0.734; 0.939]	1.000	0.600	0.714	1.000
Dif_variance__ep_b2_d1_avg	0.47	[0.44; 0.50]	0.52	[0.51; 0.53]	0.00044666	0.846	[0.737; 0.937]	0.933	0.733	0.778	0.917
Inv_Cluster_d_e__ep_b32_d2_avg	0.45	[0.43; 0.48]	0.41	[0.38; 0.43]	0.00003623	0.846	[0.734; 0.934]	0.900	0.733	0.771	0.880
Gauss_rp_s__ep_b32_d3_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00055069	0.846	[0.740; 0.929]	0.567	0.967	0.944	0.690
Cluster_p_s__ep_b32_d3_avg	3816.59	[3315.39; 5643.63]	7016.05	[5387.94; 11777.20]	0.00053153	0.846	[0.743; 0.930]	0.667	0.867	0.833	0.722
Inv_Cluster_t_e_nd__ep_b2_d1_avg	0.14	[0.13; 0.14]	0.15	[0.15; 0.16]	0.00016112	0.844	[0.736; 0.933]	0.800	0.767	0.774	0.793
Gauss_rf_s__ep_b16_d3_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00039954	0.844	[0.740; 0.930]	0.700	0.867	0.840	0.743
Cluster_t_s__ep_b32_d3_avg	2.00	[1.83; 2.84]	3.37	[2.72; 5.37]	0.00102218	0.844	[0.739; 0.936]	0.667	0.933	0.909	0.737
Contrast_e__ep_b2_d1_avg	0.71	[0.68; 0.75]	0.79	[0.77; 0.83]	0.00030475	0.843	[0.733; 0.936]	0.933	0.733	0.778	0.917
Homogeneity2_e_nd__ep_b2_d1_avg	0.36	[0.34; 0.38]	0.40	[0.39; 0.42]	0.00030475	0.843	[0.727; 0.939]	0.933	0.733	0.778	0.917
Dissimilarity_e__ep_b2_d1_avg	0.71	[0.68; 0.75]	0.79	[0.77; 0.83]	0.00030475	0.843	[0.730; 0.939]	0.933	0.733	0.778	0.917
Homogeneity1_e_nd__ep_b2_d1_avg	0.36	[0.34; 0.38]	0.40	[0.39; 0.42]	0.00030475	0.843	[0.729; 0.939]	0.933	0.733	0.778	0.917
DMN_e__ep_b2_d1_avg	0.18	[0.17; 0.19]	0.20	[0.19; 0.21]	0.00030475	0.843	[0.731; 0.938]	0.933	0.733	0.778	0.917
IDMN_e_nd__ep_b2_d1_avg	0.57	[0.55; 0.60]	0.63	[0.62; 0.67]	0.00030475	0.843	[0.729; 0.938]	0.933	0.733	0.778	0.917
DN_e__ep_b2_d1_avg	0.36	[0.34; 0.38]	0.40	[0.39; 0.42]	0.00030475	0.843	[0.730; 0.936]	0.933	0.733	0.778	0.917
IDN_e_nd__ep_b2_d1_avg	0.48	[0.46; 0.50]	0.53	[0.51; 0.56]	0.00030475	0.843	[0.729; 0.937]	0.933	0.733	0.778	0.917
Autocorrelation_e_nd__ep_b2_d1_avg	1.43	[1.37; 1.50]	1.59	[1.54; 1.67]	0.00030475	0.843	[0.730; 0.937]	0.933	0.733	0.778	0.917
Inv_autocorrelation_e_nd__ep_b2_d1_avg	0.36	[0.34; 0.38]	0.40	[0.39; 0.42]	0.00030475	0.843	[0.731; 0.936]	0.933	0.733	0.778	0.917
Gauss_e_nd__ep_b2_d1_avg	0.43	[0.41; 0.46]	0.48	[0.47; 0.51]	0.00030475	0.843	[0.732; 0.937]	0.933	0.733	0.778	0.917
Gauss_lp_e_nd__ep_b2_d1_avg	0.26	[0.25; 0.28]	0.29	[0.28; 0.31]	0.00030475	0.843	[0.729; 0.938]	0.933	0.733	0.778	0.917
Gauss_lf_e_nd__ep_b2_d1_avg	0.26	[0.25; 0.28]	0.29	[0.28; 0.31]	0.00030475	0.843	[0.730; 0.936]	0.933	0.733	0.778	0.917
Gauss_rf_e_nd__ep_b2_d1_avg	0.26	[0.25; 0.28]	0.29	[0.28; 0.31]	0.00030475	0.843	[0.732; 0.938]	0.933	0.733	0.778	0.917

Gauss_rp_e_nd__ep_b2_d1_avg	0.26	[0.25; 0.28]	0.29	[0.28; 0.31]	0.00030475	0.843	[0.731; 0.937]	0.933	0.733	0.778	0.917
Inv_Gauss_e_nd__ep_b2_d1_avg	1.18	[1.13; 1.24]	1.31	[1.27; 1.37]	0.00030475	0.843	[0.729; 0.937]	0.933	0.733	0.778	0.917
Inv_Gauss_lp_e_nd__ep_b2_d1_avg	1.94	[1.86; 2.04]	2.16	[2.10; 2.26]	0.00030475	0.843	[0.730; 0.936]	0.933	0.733	0.778	0.917
Inv_Gauss_lf_e_nd__ep_b2_d1_avg	1.94	[1.86; 2.04]	2.16	[2.10; 2.26]	0.00030475	0.843	[0.731; 0.937]	0.933	0.733	0.778	0.917
Inv_Gauss_rf_e_nd__ep_b2_d1_avg	1.94	[1.86; 2.04]	2.16	[2.10; 2.26]	0.00030475	0.843	[0.730; 0.937]	0.933	0.733	0.778	0.917
Inv_Gauss_rp_e_nd__ep_b2_d1_avg	1.94	[1.86; 2.04]	2.16	[2.10; 2.26]	0.00030475	0.843	[0.729; 0.936]	0.933	0.733	0.778	0.917
Gauss_2f_e_nd__ep_b2_d1_avg	0.53	[0.50; 0.55]	0.58	[0.57; 0.61]	0.00030475	0.843	[0.730; 0.937]	0.933	0.733	0.778	0.917
Inv_Gauss_2f_e_nd__ep_b2_d1_avg	3.88	[3.71; 4.09]	4.31	[4.19; 4.53]	0.00030475	0.843	[0.732; 0.938]	0.933	0.733	0.778	0.917
Gauss_2p_e_nd__ep_b2_d1_avg	0.53	[0.50; 0.55]	0.58	[0.57; 0.61]	0.00030475	0.843	[0.731; 0.936]	0.933	0.733	0.778	0.917
Inv_Gauss_2p_e_nd__ep_b2_d1_avg	3.88	[3.71; 4.09]	4.31	[4.19; 4.53]	0.00030475	0.843	[0.731; 0.937]	0.933	0.733	0.778	0.917
Inv_Cluster_t_nd__ep_b2_d1_avg	0.05	[0.04; 0.05]	0.05	[0.05; 0.06]	0.00018844	0.843	[0.734; 0.936]	0.967	0.633	0.725	0.950
Dif_entropy__ep_b2_d1_avg	0.77	[0.74; 0.81]	0.85	[0.83; 0.89]	0.00032424	0.843	[0.731; 0.938]	0.933	0.733	0.778	0.917
Inv_Cluster_s_nd__ep_b2_d1_avg	0.02	[0.02; 0.02]	0.02	[0.02; 0.03]	0.00016416	0.842	[0.728; 0.933]	1.000	0.600	0.714	1.000
Md_AD_md__ep_b4_d1_avg	0.03	[0.03; 0.04]	0.02	[0.02; 0.03]	0.00020597	0.842	[0.734; 0.930]	0.967	0.567	0.690	0.944
MAD__ep_b4_d1_avg	0.05	[0.05; 0.05]	0.03	[0.03; 0.05]	0.00020597	0.842	[0.733; 0.932]	0.967	0.567	0.690	0.944
Gauss_2f__ep_b8_d1_avg	0.87	[0.87; 0.88]	0.85	[0.83; 0.86]	0.00033525	0.842	[0.728; 0.936]	0.933	0.667	0.737	0.909
Cluster_t_s_nd__ep_b16_d3_avg	1.27	[1.17; 1.57]	1.72	[1.43; 2.26]	0.00044625	0.842	[0.738; 0.929]	0.667	0.967	0.952	0.744
Inv_Cluster_d_e_nd__ep_b32_d2_avg	0.41	[0.40; 0.43]	0.38	[0.36; 0.40]	0.00004454	0.842	[0.734; 0.928]	0.800	0.767	0.774	0.793
Autocorrelation_s__ep_b32_d3_avg	0.45	[0.41; 0.64]	0.75	[0.61; 1.22]	0.00088436	0.842	[0.738; 0.929]	0.667	0.900	0.870	0.730
Cluster_s_s_nd__ep_b32_d3_avg	74.87	[68.25; 111.77]	131.20	[101.02; 226.78]	0.00072906	0.842	[0.731; 0.930]	0.633	0.967	0.950	0.725
Inv_Cluster_p_nd__ep_b2_d1_avg	0.01	[0.01; 0.01]	0.01	[0.01; 0.01]	0.00014653	0.841	[0.728; 0.931]	0.767	0.800	0.793	0.774
Inv_Cluster_s_e_nd__ep_b2_d1_avg	0.06	[0.06; 0.06]	0.07	[0.06; 0.07]	0.00013362	0.841	[0.734; 0.929]	0.767	0.800	0.793	0.774
Inv_Cluster_d_nd__ep_b2_d1_avg	0.10	[0.09; 0.11]	0.12	[0.12; 0.14]	0.00022209	0.841	[0.728; 0.933]	0.967	0.633	0.725	0.950
Variance_s__ep_b16_d3_avg	0.40	[0.37; 0.50]	0.55	[0.46; 0.73]	0.00030695	0.841	[0.736; 0.928]	0.667	0.933	0.909	0.737
Inv_Cluster_p_s_nd__ep_b2_d1_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00018653	0.840	[0.727; 0.931]	0.967	0.633	0.725	0.950

Har_mean__ep_b2_d1_avg	0.17	[0.16; 0.18]	0.20	[0.19; 0.21]	0.00028717	0.840	[0.730; 0.933]	0.933	0.667	0.737	0.909
Md_AD_mn__ep_b8_d1_avg	0.01	[0.01; 0.01]	0.01	[0.01; 0.01]	0.00065296	0.840	[0.728; 0.936]	0.967	0.633	0.725	0.950
Cluster_d_s_nd__ep_b16_d2_avg	0.07	[0.07; 0.07]	0.08	[0.07; 0.09]	0.00059495	0.840	[0.732; 0.928]	0.667	0.967	0.952	0.744
Autocorrelation_s__ep_b16_d3_avg	0.32	[0.29; 0.38]	0.45	[0.35; 0.61]	0.00038264	0.840	[0.734; 0.926]	0.933	0.567	0.683	0.895
Gauss_lf_e__ep_b32_d2_avg	4.09	[3.81; 4.23]	3.54	[3.13; 3.93]	0.00006457	0.840	[0.733; 0.923]	1.000	0.533	0.682	1.000
Gauss_rf_s_nd__ep_b32_d2_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00040025	0.840	[0.731; 0.928]	0.767	0.767	0.767	0.767
Autocorrelation_s_nd__ep_b32_d3_avg	0.40	[0.37; 0.56]	0.69	[0.56; 1.10]	0.00093074	0.840	[0.730; 0.930]	0.600	1.000	1.000	0.714
Cluster_p_s_nd__ep_b32_d3_avg	3279.61	[2994.39; 5170.88]	5975.60	[4631.04; 10723.84]	0.00058442	0.840	[0.730; 0.927]	0.667	0.900	0.870	0.730
Inv_Cluster_p_e_nd__ep_b2_d1_avg	0.03	[0.03; 0.03]	0.03	[0.03; 0.03]	0.00011863	0.839	[0.730; 0.928]	0.733	0.833	0.815	0.758
Inv_Cluster_d_e__ep_b32_d1_avg	0.48	[0.47; 0.50]	0.45	[0.43; 0.47]	0.00004147	0.839	[0.727; 0.934]	1.000	0.600	0.714	1.000
Sum_energy__ep_b32_d2_avg	0.58	[0.54; 0.62]	0.66	[0.61; 0.73]	0.00012438	0.839	[0.730; 0.928]	0.900	0.633	0.711	0.864
Cluster_s_s__ep_b32_d3_avg	83.34	[74.55; 125.21]	146.22	[119.65; 246.32]	0.00067158	0.839	[0.728; 0.927]	0.667	0.900	0.870	0.730
Variance_s__ep_b32_d3_avg	0.56	[0.51; 0.81]	0.96	[0.76; 1.47]	0.00111326	0.839	[0.729; 0.927]	0.600	1.000	1.000	0.714
Contrast__ep_b2_d1_avg	0.23	[0.21; 0.26]	0.28	[0.27; 0.32]	0.00026917	0.838	[0.723; 0.934]	0.867	0.767	0.788	0.852
Homogeneity2__ep_b2_d1_avg	0.88	[0.87; 0.89]	0.86	[0.84; 0.86]	0.00026917	0.838	[0.723; 0.933]	0.867	0.767	0.788	0.852
Homogeneity2_nd__ep_b2_d1_avg	0.12	[0.11; 0.13]	0.14	[0.14; 0.16]	0.00026917	0.838	[0.724; 0.931]	0.867	0.767	0.788	0.852
Dissimilarity__ep_b2_d1_avg	0.23	[0.21; 0.26]	0.28	[0.27; 0.32]	0.00026917	0.838	[0.724; 0.931]	0.867	0.767	0.788	0.852
Homogeneity1__ep_b2_d1_avg	0.88	[0.87; 0.89]	0.86	[0.84; 0.86]	0.00026917	0.838	[0.722; 0.932]	0.867	0.767	0.788	0.852
Homogeneity1_nd__ep_b2_d1_avg	0.12	[0.11; 0.13]	0.14	[0.14; 0.16]	0.00026917	0.838	[0.726; 0.933]	0.867	0.767	0.788	0.852
DMN__ep_b2_d1_avg	0.06	[0.05; 0.06]	0.07	[0.07; 0.08]	0.00026917	0.838	[0.723; 0.933]	0.867	0.767	0.788	0.852
IDMN__ep_b2_d1_avg	0.95	[0.95; 0.96]	0.94	[0.94; 0.95]	0.00026917	0.838	[0.722; 0.934]	0.867	0.767	0.788	0.852
IDMN_nd__ep_b2_d1_avg	0.19	[0.17; 0.20]	0.23	[0.22; 0.25]	0.00026917	0.838	[0.723; 0.931]	0.867	0.767	0.788	0.852
DN__ep_b2_d1_avg	0.12	[0.11; 0.13]	0.14	[0.14; 0.16]	0.00026917	0.838	[0.724; 0.932]	0.867	0.767	0.788	0.852
IDN__ep_b2_d1_avg	0.92	[0.91; 0.93]	0.91	[0.89; 0.91]	0.00026917	0.838	[0.722; 0.932]	0.867	0.767	0.788	0.852
IDN_nd__ep_b2_d1_avg	0.16	[0.14; 0.17]	0.19	[0.18; 0.21]	0.00026917	0.838	[0.722; 0.931]	0.867	0.767	0.788	0.852

Autocorrelation_nd__ep_b2_d1_avg	0.47	[0.43; 0.51]	0.57	[0.54; 0.63]	0.00026917	0.838	[0.724; 0.931]	0.867	0.767	0.788	0.852
Inv_autocorrelation_nd__ep_b2_d1_avg	0.12	[0.11; 0.13]	0.14	[0.14; 0.16]	0.00026917	0.838	[0.727; 0.933]	0.867	0.767	0.788	0.852
Gauss_nd__ep_b2_d1_avg	0.14	[0.13; 0.15]	0.17	[0.16; 0.19]	0.00026917	0.838	[0.721; 0.933]	0.867	0.767	0.788	0.852
Gauss_lp_nd__ep_b2_d1_avg	0.09	[0.08; 0.09]	0.10	[0.10; 0.12]	0.00026917	0.838	[0.724; 0.937]	0.867	0.767	0.788	0.852
Gauss_lf_nd__ep_b2_d1_avg	0.09	[0.08; 0.09]	0.10	[0.10; 0.12]	0.00026917	0.838	[0.724; 0.933]	0.867	0.767	0.788	0.852
Gauss_rf_nd__ep_b2_d1_avg	0.09	[0.08; 0.09]	0.10	[0.10; 0.12]	0.00026917	0.838	[0.726; 0.934]	0.867	0.767	0.788	0.852
Gauss_rp_nd__ep_b2_d1_avg	0.09	[0.08; 0.09]	0.10	[0.10; 0.12]	0.00026917	0.838	[0.727; 0.931]	0.867	0.767	0.788	0.852
Inv_Gauss_nd__ep_b2_d1_avg	0.39	[0.35; 0.42]	0.47	[0.45; 0.52]	0.00026917	0.838	[0.723; 0.933]	0.867	0.767	0.788	0.852
Inv_Gauss_lp_nd__ep_b2_d1_avg	0.64	[0.58; 0.69]	0.77	[0.73; 0.86]	0.00026917	0.838	[0.724; 0.932]	0.867	0.767	0.788	0.852
Inv_Gauss_lf_nd__ep_b2_d1_avg	0.64	[0.58; 0.69]	0.77	[0.73; 0.86]	0.00026917	0.838	[0.723; 0.934]	0.867	0.767	0.788	0.852
Inv_Gauss_rf_nd__ep_b2_d1_avg	0.64	[0.58; 0.69]	0.77	[0.73; 0.86]	0.00026917	0.838	[0.721; 0.934]	0.867	0.767	0.788	0.852
Inv_Gauss_rp_nd__ep_b2_d1_avg	0.64	[0.58; 0.69]	0.77	[0.73; 0.86]	0.00026917	0.838	[0.721; 0.932]	0.867	0.767	0.788	0.852
Gauss_2f__ep_b2_d1_avg	1.04	[1.03; 1.05]	1.02	[1.01; 1.03]	0.00026917	0.838	[0.727; 0.934]	0.867	0.767	0.788	0.852
Gauss_2f_nd__ep_b2_d1_avg	0.17	[0.16; 0.19]	0.21	[0.20; 0.23]	0.00026917	0.838	[0.726; 0.933]	0.867	0.767	0.788	0.852
Inv_Gauss_2f__ep_b2_d1_avg	7.70	[7.64; 7.76]	7.55	[7.46; 7.59]	0.00026917	0.838	[0.723; 0.933]	0.867	0.767	0.788	0.852
Inv_Gauss_2f_nd__ep_b2_d1_avg	1.27	[1.16; 1.39]	1.55	[1.47; 1.71]	0.00026917	0.838	[0.721; 0.933]	0.867	0.767	0.788	0.852
Gauss_2p__ep_b2_d1_avg	1.04	[1.03; 1.05]	1.02	[1.01; 1.03]	0.00026917	0.838	[0.726; 0.933]	0.867	0.767	0.788	0.852
Gauss_2p_nd__ep_b2_d1_avg	0.17	[0.16; 0.19]	0.21	[0.20; 0.23]	0.00026917	0.838	[0.724; 0.933]	0.867	0.767	0.788	0.852
Inv_Gauss_2p__ep_b2_d1_avg	7.70	[7.64; 7.76]	7.55	[7.46; 7.59]	0.00026917	0.838	[0.721; 0.933]	0.867	0.767	0.788	0.852
Inv_Gauss_2p_nd__ep_b2_d1_avg	1.27	[1.16; 1.39]	1.55	[1.47; 1.71]	0.00026917	0.838	[0.726; 0.933]	0.867	0.767	0.788	0.852
Inv_Cluster_d_e__ep_b2_d1_avg	0.87	[0.85; 0.88]	0.91	[0.90; 0.92]	0.00036272	0.838	[0.727; 0.931]	0.933	0.667	0.737	0.909
Dif_average__ep_b2_d1_avg	0.23	[0.21; 0.26]	0.28	[0.27; 0.32]	0.00026917	0.838	[0.723; 0.931]	0.867	0.767	0.788	0.852
Inv_dif_average__ep_b2_d1_avg	0.23	[0.21; 0.26]	0.28	[0.27; 0.32]	0.00026917	0.838	[0.723; 0.934]	0.867	0.767	0.788	0.852
Mode__ep_b2_d1_avg	0.12	[0.11; 0.13]	0.14	[0.14; 0.16]	0.00026917	0.838	[0.723; 0.932]	0.867	0.767	0.788	0.852
High_notch__ep_b4_d1_avg	0.11	[0.10; 0.11]	0.10	[0.09; 0.10]	0.00030923	0.838	[0.721; 0.932]	0.833	0.833	0.833	0.833

IQR__ep_b8_d1_avg	0.02	[0.02; 0.02]	0.01	[0.01; 0.02]	0.00074011	0.838	[0.723; 0.933]	0.900	0.700	0.750	0.875
Sum_energy__ep_b16_d1_avg	0.54	[0.52; 0.55]	0.57	[0.55; 0.62]	0.00007671	0.838	[0.726; 0.929]	0.967	0.667	0.744	0.952
Cluster_t_s_nd__ep_b32_d3_avg	1.82	[1.69; 2.62]	3.18	[2.56; 4.90]	0.00109577	0.838	[0.729; 0.927]	0.600	1.000	1.000	0.714
Inv_Cluster_d_e_nd__ep_b32_d3_avg	0.39	[0.37; 0.41]	0.35	[0.32; 0.37]	0.00005582	0.838	[0.732; 0.923]	0.667	0.833	0.800	0.714
Inv_Cluster_s_s_nd__ep_b2_d1_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00019995	0.837	[0.724; 0.931]	0.967	0.633	0.725	0.950
Geo_mean3__ep_b2_d1_avg	0.21	[0.20; 0.21]	0.22	[0.22; 0.23]	0.00027511	0.837	[0.719; 0.930]	0.800	0.767	0.774	0.793
Mn_AD_md__ep_b2_d1_avg	0.13	[0.12; 0.14]	0.11	[0.09; 0.11]	0.00028230	0.837	[0.719; 0.931]	0.867	0.767	0.788	0.852
Md_AD_mn__ep_b2_d1_avg	0.13	[0.12; 0.14]	0.11	[0.09; 0.11]	0.00028230	0.837	[0.722; 0.934]	0.867	0.767	0.788	0.852
Inv_autocorrelation_e__ep_b2_d1_avg	1.01	[0.99; 1.02]	1.05	[1.03; 1.07]	0.00025896	0.836	[0.723; 0.926]	0.800	0.767	0.774	0.793
Geo_mean__ep_b2_d1_avg	0.10	[0.10; 0.11]	0.11	[0.11; 0.12]	0.00026838	0.836	[0.727; 0.930]	0.767	0.800	0.793	0.774
Geo_mean2__ep_b2_d1_avg	0.10	[0.10; 0.11]	0.11	[0.11; 0.12]	0.00026838	0.836	[0.723; 0.930]	0.767	0.800	0.793	0.774
Gauss_2f__ep_b4_d1_avg	0.92	[0.92; 0.93]	0.90	[0.89; 0.91]	0.00034678	0.836	[0.721; 0.933]	0.767	0.833	0.821	0.781
Gauss_2f__ep_b16_d1_avg	0.85	[0.84; 0.86]	0.82	[0.80; 0.84]	0.00034530	0.836	[0.720; 0.933]	0.767	0.800	0.793	0.774
Inv_Cluster_t_s_nd__ep_b2_d1_avg	0.01	[0.00; 0.01]	0.01	[0.01; 0.01]	0.00021605	0.834	[0.719; 0.930]	0.833	0.767	0.781	0.821
Mn_AD_mn__ep_b2_d1_avg	0.13	[0.12; 0.14]	0.11	[0.09; 0.12]	0.00026282	0.834	[0.720; 0.929]	0.900	0.700	0.750	0.875
Inv_Cluster_d_e__ep_b16_d2_avg	0.68	[0.65; 0.70]	0.63	[0.60; 0.65]	0.00004456	0.834	[0.723; 0.929]	0.867	0.700	0.743	0.840
Cluster_s_s_nd__ep_b16_d3_avg	26.11	[23.20; 33.05]	36.59	[29.45; 50.76]	0.00041754	0.834	[0.727; 0.922]	0.633	0.900	0.864	0.711
Correlation__ep_b2_d1_avg	0.52	[0.47; 0.56]	0.42	[0.36; 0.45]	0.00034132	0.833	[0.719; 0.931]	0.900	0.767	0.794	0.885
Min__ep_b2_d1_avg	0.12	[0.11; 0.13]	0.14	[0.14; 0.16]	0.00030375	0.833	[0.719; 0.930]	0.867	0.767	0.788	0.852
Low_notch__ep_b8_d1_avg	-0.02	[-0.02; -0.02]	-0.01	[-0.02; -0.01]	0.00080425	0.833	[0.718; 0.931]	0.933	0.700	0.757	0.913
Sum_energy__ep_b16_d2_avg	0.58	[0.55; 0.61]	0.65	[0.61; 0.69]	0.00010014	0.833	[0.723; 0.924]	0.967	0.600	0.707	0.947
Gauss_2f__ep_b32_d1_avg	0.84	[0.83; 0.85]	0.81	[0.79; 0.83]	0.00032886	0.833	[0.720; 0.931]	0.967	0.600	0.707	0.947
Gauss_e__ep_b2_d1_avg	1.07	[1.04; 1.08]	1.11	[1.09; 1.14]	0.00024153	0.832	[0.717; 0.922]	0.767	0.800	0.793	0.774
Inv_Gauss_e__ep_b2_d1_avg	2.90	[2.84; 2.95]	3.03	[2.96; 3.09]	0.00024153	0.832	[0.721; 0.926]	0.767	0.800	0.793	0.774
Cluster_d_nd__ep_b2_d1_avg	0.53	[0.49; 0.59]	0.65	[0.61; 0.70]	0.00033591	0.831	[0.712; 0.930]	0.900	0.700	0.750	0.875

Cluster_d_e_nd__ep_b2_d1_avg	1.63	[1.57; 1.73]	1.80	[1.74; 1.86]	0.00048968	0.831	[0.714; 0.930]	0.933	0.700	0.757	0.913
Inv_Cluster_d_s_nd__ep_b2_d1_avg	0.01	[0.01; 0.01]	0.02	[0.02; 0.02]	0.00023539	0.831	[0.718; 0.929]	0.833	0.767	0.781	0.821
Low_notch__ep_b2_d1_avg	-0.24	[-0.27; -0.20]	-0.14	[-0.18; -0.09]	0.00055495	0.831	[0.710; 0.931]	0.833	0.800	0.806	0.828
Gauss_lp_e__ep_b32_d2_avg	2.04	[1.89; 2.25]	1.73	[1.55; 1.94]	0.00006297	0.831	[0.719; 0.922]	0.900	0.633	0.711	0.864
Gauss_lp_e__ep_b2_d1_avg	0.86	[0.84; 0.87]	0.89	[0.87; 0.90]	0.00028006	0.830	[0.720; 0.921]	0.800	0.767	0.774	0.793
Gauss_lf_e__ep_b2_d1_avg	0.86	[0.84; 0.87]	0.89	[0.87; 0.90]	0.00028006	0.830	[0.718; 0.922]	0.800	0.767	0.774	0.793
Inv_Gauss_rf_e__ep_b2_d1_avg	6.33	[6.20; 6.40]	6.56	[6.45; 6.62]	0.00028006	0.830	[0.718; 0.922]	0.800	0.767	0.774	0.793
Inv_Gauss_rp_e__ep_b2_d1_avg	6.33	[6.20; 6.40]	6.56	[6.45; 6.62]	0.00028006	0.830	[0.717; 0.924]	0.800	0.767	0.774	0.793
Gauss_lf_e_nd__ep_b32_d2_avg	3.87	[3.62; 3.99]	3.38	[2.99; 3.70]	0.00007989	0.830	[0.720; 0.918]	1.000	0.533	0.682	1.000
Inv_Cluster_d_e__ep_b32_d3_avg	0.42	[0.39; 0.44]	0.37	[0.34; 0.39]	0.00007517	0.830	[0.722; 0.921]	0.767	0.733	0.742	0.759
Homogeneity2_s__ep_b2_d1_avg	0.31	[0.31; 0.33]	0.29	[0.27; 0.30]	0.00020625	0.829	[0.719; 0.921]	0.767	0.800	0.793	0.774
Homogeneity1_s__ep_b2_d1_avg	0.31	[0.31; 0.33]	0.29	[0.27; 0.30]	0.00020625	0.829	[0.717; 0.922]	0.767	0.800	0.793	0.774
IDMN_s__ep_b2_d1_avg	0.32	[0.32; 0.34]	0.30	[0.28; 0.31]	0.00021402	0.829	[0.713; 0.922]	0.900	0.633	0.711	0.864
IDMN_e__ep_b2_d1_avg	1.62	[1.59; 1.64]	1.68	[1.64; 1.71]	0.00024368	0.829	[0.718; 0.921]	0.767	0.767	0.767	0.767
Cluster_d_s_nd__ep_b2_d1_avg	0.07	[0.05; 0.08]	0.10	[0.08; 0.11]	0.00028662	0.829	[0.708; 0.928]	0.867	0.733	0.765	0.846
Average_e__ep_b2_d1_avg	2.64	[2.58; 2.68]	2.75	[2.69; 2.81]	0.00024126	0.829	[0.714; 0.923]	1.000	0.567	0.698	1.000
Gauss_lp_e__ep_b32_d3_avg	1.82	[1.61; 2.00]	1.48	[1.33; 1.65]	0.00010072	0.829	[0.717; 0.919]	0.967	0.533	0.674	0.941
IDN_s__ep_b2_d1_avg	0.32	[0.31; 0.34]	0.29	[0.27; 0.31]	0.00020980	0.828	[0.716; 0.921]	0.767	0.767	0.767	0.767
Gauss_2f_s__ep_b2_d1_avg	0.36	[0.35; 0.38]	0.33	[0.31; 0.35]	0.00020932	0.828	[0.716; 0.921]	0.767	0.767	0.767	0.767
Inv_Gauss_2f_s__ep_b2_d1_avg	2.68	[2.61; 2.81]	2.45	[2.29; 2.59]	0.00020932	0.828	[0.713; 0.921]	0.767	0.767	0.767	0.767
Gauss_2p_s__ep_b2_d1_avg	0.36	[0.35; 0.38]	0.33	[0.31; 0.35]	0.00020932	0.828	[0.718; 0.921]	0.767	0.767	0.767	0.767
Inv_Gauss_2p_s__ep_b2_d1_avg	2.68	[2.61; 2.81]	2.45	[2.29; 2.59]	0.00020932	0.828	[0.717; 0.920]	0.767	0.767	0.767	0.767
Cluster_s_s_nd__ep_b2_d1_avg	0.35	[0.29; 0.41]	0.50	[0.42; 0.57]	0.00036133	0.828	[0.709; 0.926]	0.900	0.700	0.750	0.875
Variance_e__ep_b2_d1_avg	2.82	[2.48; 2.97]	3.38	[3.04; 3.76]	0.00022488	0.828	[0.714; 0.923]	0.767	0.800	0.793	0.774
Contrast__ep_b4_d1_avg	0.87	[0.82; 0.98]	1.10	[1.04; 1.26]	0.00039608	0.828	[0.704; 0.934]	0.933	0.767	0.800	0.920

DMN__ep_b4_d1_avg	0.05	[0.05; 0.06]	0.07	[0.06; 0.08]	0.00039608	0.828	[0.702; 0.934]	0.933	0.767	0.800	0.920
Gauss_lf_e__ep_b32_d3_avg	3.77	[3.46; 4.04]	3.23	[2.84; 3.56]	0.00007833	0.828	[0.712; 0.917]	1.000	0.500	0.667	1.000
Contrast_s__ep_b2_d1_avg	0.03	[0.02; 0.03]	0.04	[0.04; 0.05]	0.00025863	0.827	[0.708; 0.927]	0.867	0.733	0.765	0.846
Homogeneity2_s_nd__ep_b2_d1_avg	0.01	[0.01; 0.02]	0.02	[0.02; 0.03]	0.00025863	0.827	[0.710; 0.926]	0.867	0.733	0.765	0.846
Dissimilarity_s__ep_b2_d1_avg	0.03	[0.02; 0.03]	0.04	[0.04; 0.05]	0.00025863	0.827	[0.709; 0.926]	0.867	0.733	0.765	0.846
Homogeneity1_s_nd__ep_b2_d1_avg	0.01	[0.01; 0.02]	0.02	[0.02; 0.03]	0.00025863	0.827	[0.708; 0.926]	0.867	0.733	0.765	0.846
DMN_s__ep_b2_d1_avg	0.01	[0.01; 0.01]	0.01	[0.01; 0.01]	0.00025863	0.827	[0.709; 0.926]	0.867	0.733	0.765	0.846
IDMN_s_nd__ep_b2_d1_avg	0.02	[0.02; 0.03]	0.03	[0.03; 0.04]	0.00025863	0.827	[0.710; 0.927]	0.867	0.733	0.765	0.846
DN_s__ep_b2_d1_avg	0.01	[0.01; 0.02]	0.02	[0.02; 0.03]	0.00025863	0.827	[0.711; 0.927]	0.867	0.733	0.765	0.846
IDN_s_nd__ep_b2_d1_avg	0.02	[0.02; 0.02]	0.03	[0.02; 0.03]	0.00025863	0.827	[0.707; 0.923]	0.867	0.733	0.765	0.846
Autocorrelation_s_nd__ep_b2_d1_avg	0.06	[0.05; 0.07]	0.08	[0.07; 0.10]	0.00025863	0.827	[0.711; 0.927]	0.867	0.733	0.765	0.846
Inv_autocorrelation_s_nd__ep_b2_d1_avg	0.01	[0.01; 0.02]	0.02	[0.02; 0.03]	0.00025863	0.827	[0.710; 0.927]	0.867	0.733	0.765	0.846
Gauss_s_nd__ep_b2_d1_avg	0.02	[0.01; 0.02]	0.03	[0.02; 0.03]	0.00025863	0.827	[0.709; 0.924]	0.867	0.733	0.765	0.846
Gauss_lp_s_nd__ep_b2_d1_avg	0.01	[0.01; 0.01]	0.02	[0.01; 0.02]	0.00025863	0.827	[0.711; 0.924]	0.867	0.733	0.765	0.846
Gauss_lf_s_nd__ep_b2_d1_avg	0.01	[0.01; 0.01]	0.02	[0.01; 0.02]	0.00025863	0.827	[0.710; 0.926]	0.867	0.733	0.765	0.846
Gauss_rf_s_nd__ep_b2_d1_avg	0.01	[0.01; 0.01]	0.02	[0.01; 0.02]	0.00025863	0.827	[0.710; 0.924]	0.867	0.733	0.765	0.846
Gauss_rp_s_nd__ep_b2_d1_avg	0.01	[0.01; 0.01]	0.02	[0.01; 0.02]	0.00025863	0.827	[0.709; 0.926]	0.867	0.733	0.765	0.846
Inv_Gauss_s_nd__ep_b2_d1_avg	0.05	[0.04; 0.06]	0.07	[0.06; 0.08]	0.00025863	0.827	[0.711; 0.923]	0.867	0.733	0.765	0.846
Inv_Gauss_lp_s_nd__ep_b2_d1_avg	0.08	[0.06; 0.09]	0.11	[0.10; 0.14]	0.00025863	0.827	[0.708; 0.927]	0.867	0.733	0.765	0.846
Inv_Gauss_lf_s_nd__ep_b2_d1_avg	0.08	[0.06; 0.09]	0.11	[0.10; 0.14]	0.00025863	0.827	[0.708; 0.923]	0.867	0.733	0.765	0.846
Inv_Gauss_rf_s_nd__ep_b2_d1_avg	0.08	[0.06; 0.09]	0.11	[0.10; 0.14]	0.00025863	0.827	[0.707; 0.927]	0.867	0.733	0.765	0.846
Inv_Gauss_rp_s_nd__ep_b2_d1_avg	0.08	[0.06; 0.09]	0.11	[0.10; 0.14]	0.00025863	0.827	[0.710; 0.923]	0.867	0.733	0.765	0.846
Gauss_2f_s_nd__ep_b2_d1_avg	0.02	[0.02; 0.02]	0.03	[0.03; 0.04]	0.00025863	0.827	[0.708; 0.924]	0.867	0.733	0.765	0.846
Inv_Gauss_2f_s_nd__ep_b2_d1_avg	0.16	[0.13; 0.18]	0.23	[0.20; 0.28]	0.00025863	0.827	[0.708; 0.924]	0.867	0.733	0.765	0.846
Gauss_2p_s_nd__ep_b2_d1_avg	0.02	[0.02; 0.02]	0.03	[0.03; 0.04]	0.00025863	0.827	[0.707; 0.926]	0.867	0.733	0.765	0.846

Inv_Gauss_2p_s_nd__ep_b2_d1_avg	0.16	[0.13; 0.18]	0.23	[0.20; 0.28]	0.00025863	0.827	[0.710; 0.924]	0.867	0.733	0.765	0.846
Cluster_t_s_nd__ep_b2_d1_avg	0.15	[0.12; 0.18]	0.22	[0.19; 0.26]	0.00032041	0.827	[0.711; 0.928]	0.867	0.733	0.765	0.846
Cluster_d_s_nd__ep_b16_d3_avg	0.07	[0.07; 0.08]	0.09	[0.08; 0.11]	0.00106986	0.827	[0.714; 0.921]	0.700	0.867	0.840	0.743
Gauss_lp_e_nd__ep_b32_d3_avg	1.72	[1.51; 1.86]	1.40	[1.25; 1.56]	0.00009474	0.827	[0.717; 0.917]	0.967	0.533	0.674	0.941
Dif_energy__ep_b2_d1_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00032839	0.826	[0.710; 0.926]	0.900	0.667	0.730	0.870
Inv_dif_energy__ep_b2_d1_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00032839	0.826	[0.709; 0.926]	0.900	0.667	0.730	0.870
Har_mean__ep_b8_d1_avg	0.00	[0.00; 0.01]	0.01	[0.01; 0.01]	0.00048208	0.826	[0.711; 0.920]	0.733	0.833	0.815	0.758
High_notch__ep_b8_d1_avg	0.03	[0.03; 0.03]	0.03	[0.03; 0.03]	0.00093615	0.826	[0.709; 0.927]	0.767	0.833	0.821	0.781
IMC2__ep_b8_d3_avg	0.30	[0.28; 0.33]	0.36	[0.33; 0.40]	0.00263154	0.826	[0.712; 0.924]	0.767	0.833	0.821	0.781
Cluster_p_s_nd__ep_b16_d3_avg	588.24	[510.88; 724.90]	822.46	[655.98; 1218.28]	0.00054586	0.826	[0.712; 0.917]	0.633	0.867	0.826	0.703
Gauss_lp_e_nd__ep_b32_d2_avg	1.90	[1.75; 2.08]	1.64	[1.45; 1.81]	0.00008702	0.826	[0.716; 0.918]	0.900	0.600	0.692	0.857
Gauss_s__ep_b2_d1_avg	0.20	[0.19; 0.21]	0.19	[0.18; 0.19]	0.00022384	0.824	[0.710; 0.918]	1.000	0.500	0.667	1.000
Inv_Gauss_s__ep_b2_d1_avg	0.54	[0.53; 0.56]	0.50	[0.48; 0.53]	0.00022384	0.824	[0.710; 0.919]	1.000	0.500	0.667	1.000
Variance__ep_b2_d1_avg	0.03	[0.02; 0.03]	0.02	[0.01; 0.02]	0.00022384	0.824	[0.712; 0.917]	1.000	0.500	0.667	1.000
Energy__ep_b2_d1_avg	0.33	[0.32; 0.34]	0.31	[0.29; 0.32]	0.00022384	0.824	[0.712; 0.918]	1.000	0.500	0.667	1.000
Cluster_d_s__ep_b16_d3_avg	0.08	[0.08; 0.10]	0.10	[0.09; 0.13]	0.00070038	0.824	[0.710; 0.918]	0.667	0.900	0.870	0.730
Average_s__ep_b16_d3_avg	0.04	[0.04; 0.05]	0.05	[0.05; 0.06]	0.00070354	0.824	[0.709; 0.918]	0.667	0.900	0.870	0.730
Cluster_t_s_nd__ep_b32_d2_avg	1.70	[1.55; 2.21]	2.39	[2.06; 3.82]	0.00045036	0.824	[0.710; 0.916]	0.600	0.900	0.857	0.692
Cluster_t_nd__ep_b2_d1_avg	1.22	[1.12; 1.34]	1.47	[1.36; 1.58]	0.00043194	0.823	[0.704; 0.926]	0.900	0.700	0.750	0.875
RMS__ep_b2_d1_avg	0.29	[0.28; 0.29]	0.28	[0.27; 0.28]	0.00022007	0.823	[0.709; 0.918]	1.000	0.500	0.667	1.000
IDMN__ep_b4_d1_avg	0.95	[0.95; 0.96]	0.94	[0.94; 0.95]	0.00043567	0.823	[0.697; 0.931]	0.933	0.733	0.778	0.917
Sum_energy__ep_b8_d2_avg	0.62	[0.58; 0.64]	0.68	[0.63; 0.72]	0.00010531	0.823	[0.708; 0.920]	0.833	0.667	0.714	0.800
Cluster_s_s__ep_b16_d3_avg	29.64	[26.79; 35.93]	42.62	[33.40; 60.35]	0.00038278	0.823	[0.709; 0.914]	0.533	0.967	0.941	0.674
Gauss_lp_e__ep_b32_d1_avg	2.24	[2.16; 2.33]	2.03	[1.93; 2.16]	0.00008033	0.823	[0.712; 0.917]	0.933	0.600	0.700	0.900
Cluster_p_s_nd__ep_b2_d1_avg	0.79	[0.66; 0.94]	1.13	[0.95; 1.32]	0.00041105	0.822	[0.703; 0.923]	0.900	0.700	0.750	0.875

Dissimilarity_e__ep_b4_d1_avg	2.75	[2.64; 2.98]	3.21	[3.06; 3.45]	0.00055431	0.822	[0.690; 0.932]	0.933	0.767	0.800	0.920
DN_e__ep_b4_d1_avg	0.69	[0.66; 0.74]	0.80	[0.76; 0.86]	0.00055431	0.822	[0.690; 0.933]	0.933	0.767	0.800	0.920
Dif_entropy__ep_b4_d1_avg	1.39	[1.34; 1.46]	1.51	[1.49; 1.57]	0.00061625	0.822	[0.693; 0.934]	0.933	0.767	0.800	0.920
Gauss_lf_e_nd__ep_b32_d3_avg	3.60	[3.31; 3.83]	3.11	[2.73; 3.41]	0.00007677	0.822	[0.709; 0.912]	1.000	0.500	0.667	1.000
IDN_e__ep_b2_d1_avg	1.52	[1.50; 1.54]	1.57	[1.54; 1.60]	0.00025156	0.821	[0.708; 0.916]	1.000	0.500	0.667	1.000
Cluster_d_e__ep_b2_d1_avg	4.00	[3.94; 4.08]	4.17	[4.10; 4.21]	0.00060619	0.821	[0.706; 0.922]	0.933	0.633	0.718	0.905
Contrast_e__ep_b4_d1_avg	4.18	[3.80; 4.72]	5.25	[5.00; 5.74]	0.00041853	0.821	[0.692; 0.930]	0.933	0.767	0.800	0.920
DMN_e__ep_b4_d1_avg	0.26	[0.24; 0.30]	0.33	[0.31; 0.36]	0.00041853	0.821	[0.696; 0.932]	0.933	0.767	0.800	0.920
Sum_energy__ep_b4_d2_avg	0.72	[0.68; 0.73]	0.77	[0.73; 0.81]	0.00013942	0.821	[0.707; 0.917]	0.767	0.733	0.742	0.759
Contrast__ep_b8_d1_avg	3.32	[3.01; 3.76]	4.28	[3.89; 4.84]	0.00051908	0.821	[0.693; 0.930]	1.000	0.667	0.750	1.000
DMN__ep_b8_d1_avg	0.05	[0.05; 0.06]	0.07	[0.06; 0.08]	0.00051908	0.821	[0.693; 0.929]	1.000	0.667	0.750	1.000
Dif_entropy__ep_b8_d1_avg	2.13	[2.06; 2.20]	2.28	[2.24; 2.35]	0.00087352	0.821	[0.688; 0.933]	0.933	0.767	0.800	0.920
Homogeneity2_e_nd__ep_b16_d1_avg	1.25	[1.22; 1.29]	1.14	[1.09; 1.22]	0.00032285	0.821	[0.698; 0.921]	1.000	0.567	0.698	1.000
Dif_variance__ep_b32_d1_avg	24.30	[20.92; 28.13]	32.58	[30.08; 36.90]	0.00049184	0.821	[0.690; 0.933]	0.933	0.767	0.800	0.920
IDMN__ep_b8_d1_avg	0.96	[0.95; 0.96]	0.94	[0.94; 0.95]	0.00052552	0.820	[0.694; 0.928]	1.000	0.633	0.732	1.000
Homogeneity1_nd__ep_b16_d1_avg	0.26	[0.25; 0.26]	0.24	[0.24; 0.25]	0.00029283	0.820	[0.702; 0.920]	1.000	0.567	0.698	1.000
IDMN__ep_b16_d1_avg	0.96	[0.95; 0.96]	0.95	[0.94; 0.95]	0.00056161	0.820	[0.694; 0.929]	0.933	0.733	0.778	0.917
Contrast__ep_b32_d1_avg	52.37	[46.05; 58.34]	67.21	[60.51; 76.13]	0.00053805	0.820	[0.693; 0.930]	1.000	0.633	0.732	1.000
DMN__ep_b32_d1_avg	0.05	[0.04; 0.06]	0.07	[0.06; 0.07]	0.00053805	0.820	[0.693; 0.928]	1.000	0.633	0.732	1.000
Gauss_rf_s__ep_b32_d2_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00038226	0.820	[0.707; 0.911]	0.833	0.633	0.694	0.792
Cluster_d_s__ep_b32_d3_avg	0.06	[0.05; 0.08]	0.09	[0.07; 0.13]	0.00202400	0.820	[0.703; 0.916]	0.700	0.867	0.840	0.743
Average_s__ep_b32_d3_avg	0.03	[0.02; 0.04]	0.05	[0.04; 0.07]	0.00202555	0.820	[0.704; 0.917]	0.700	0.867	0.840	0.743
SD__ep_b2_d1_avg	0.16	[0.15; 0.17]	0.13	[0.11; 0.15]	0.00022934	0.819	[0.703; 0.912]	1.000	0.500	0.667	1.000
Uniformity__ep_b4_d3_avg	0.12	[0.12; 0.12]	0.13	[0.13; 0.14]	0.00063803	0.819	[0.702; 0.921]	0.767	0.767	0.767	0.767
Inv_Cluster_d_e_nd__ep_b8_d3_avg	0.70	[0.69; 0.71]	0.68	[0.66; 0.69]	0.00017600	0.819	[0.704; 0.914]	0.933	0.600	0.700	0.900

Inv_dif_average__ep_b16_d1_avg	0.43	[0.42; 0.43]	0.39	[0.38; 0.42]	0.00036295	0.819	[0.698; 0.923]	0.767	0.800	0.793	0.774
Gauss_rp_s_nd__ep_b16_d3_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00075244	0.819	[0.707; 0.912]	0.900	0.567	0.675	0.850
Inv_Cluster_d_e_nd__ep_b16_d3_avg	0.57	[0.55; 0.59]	0.53	[0.51; 0.56]	0.00006433	0.819	[0.704; 0.913]	0.567	0.933	0.895	0.683
IDMN__ep_b32_d1_avg	0.96	[0.95; 0.96]	0.95	[0.94; 0.95]	0.00055927	0.819	[0.691; 0.929]	1.000	0.633	0.732	1.000
Gauss_2f_e__ep_b2_d1_avg	1.71	[1.68; 1.73]	1.77	[1.73; 1.80]	0.00025348	0.818	[0.706; 0.914]	1.000	0.500	0.667	1.000
Inv_Gauss_2f_e__ep_b2_d1_avg	12.65	[12.44; 12.82]	13.07	[12.82; 13.28]	0.00025348	0.818	[0.703; 0.916]	1.000	0.500	0.667	1.000
Gauss_2p_e__ep_b2_d1_avg	1.71	[1.68; 1.73]	1.77	[1.73; 1.80]	0.00025348	0.818	[0.699; 0.912]	1.000	0.500	0.667	1.000
Inv_Gauss_2p_e__ep_b2_d1_avg	12.65	[12.44; 12.82]	13.07	[12.82; 13.28]	0.00025348	0.818	[0.704; 0.913]	1.000	0.500	0.667	1.000
Cluster_s_nd__ep_b2_d1_avg	2.78	[2.59; 3.08]	3.32	[3.09; 3.61]	0.00057232	0.818	[0.697; 0.921]	0.900	0.700	0.750	0.875
Contrast__ep_b16_d1_avg	13.14	[11.62; 14.72]	16.84	[15.23; 19.14]	0.00054155	0.818	[0.689; 0.929]	1.000	0.633	0.732	1.000
DMN__ep_b16_d1_avg	0.05	[0.05; 0.06]	0.07	[0.06; 0.07]	0.00054155	0.818	[0.688; 0.926]	1.000	0.633	0.732	1.000
Inv_Gauss_lf_s_nd__ep_b32_d3_avg	0.02	[0.02; 0.04]	0.05	[0.03; 0.08]	0.00108584	0.818	[0.704; 0.910]	0.533	0.967	0.941	0.674
IQR__ep_b2_d1_avg	0.23	[0.22; 0.26]	0.19	[0.17; 0.21]	0.00078649	0.817	[0.693; 0.921]	0.833	0.800	0.806	0.828
Inv_Cluster_d_e__ep_b8_d2_avg	0.89	[0.88; 0.91]	0.85	[0.82; 0.88]	0.00008946	0.817	[0.702; 0.913]	0.900	0.633	0.711	0.864
Gauss_lp_e__ep_b16_d2_avg	1.83	[1.69; 1.95]	1.56	[1.43; 1.76]	0.00010507	0.817	[0.703; 0.913]	0.867	0.667	0.722	0.833
Autocorrelation_s_nd__ep_b32_d2_avg	0.39	[0.35; 0.49]	0.56	[0.46; 0.88]	0.00048820	0.817	[0.702; 0.910]	0.633	0.833	0.792	0.694
Gauss_2f_e__ep_b32_d2_avg	7.08	[6.74; 7.22]	6.56	[6.22; 6.85]	0.00047812	0.817	[0.696; 0.914]	0.667	0.900	0.870	0.730
Cluster_s_s_nd__ep_b32_d2_avg	71.09	[61.94; 93.19]	107.61	[82.91; 181.04]	0.00054312	0.817	[0.702; 0.910]	0.567	0.933	0.895	0.683
Cluster_d_s_nd__ep_b32_d3_avg	0.05	[0.05; 0.07]	0.09	[0.07; 0.12]	0.00212885	0.817	[0.700; 0.914]	0.733	0.833	0.815	0.758
Cluster_t_e_nd__ep_b2_d1_avg	3.72	[3.59; 3.93]	4.09	[3.95; 4.19]	0.00087824	0.816	[0.693; 0.921]	0.900	0.700	0.750	0.875
Correlation_e__ep_b2_d1_avg	0.87	[0.86; 0.87]	0.88	[0.87; 0.89]	0.00031647	0.816	[0.698; 0.911]	1.000	0.500	0.667	1.000
Mn_AD_md__ep_b4_d1_avg	0.04	[0.03; 0.04]	0.03	[0.03; 0.03]	0.00057022	0.816	[0.691; 0.923]	1.000	0.600	0.714	1.000
Cluster_t_s_nd__ep_b8_d2_avg	1.01	[0.92; 1.08]	1.15	[1.06; 1.34]	0.00064150	0.816	[0.698; 0.913]	0.900	0.667	0.730	0.870
Dif_variance__ep_b16_d1_avg	5.54	[5.20; 6.28]	6.83	[6.66; 7.78]	0.00057748	0.816	[0.690; 0.926]	0.900	0.767	0.794	0.885
Sum_energy__ep_b32_d3_avg	0.65	[0.61; 0.70]	0.75	[0.71; 0.84]	0.00066772	0.814	[0.698; 0.914]	0.767	0.767	0.767	0.767

Dif_entropy__ep_b16_d1_avg	2.97	[2.90; 3.06]	3.15	[3.09; 3.22]	0.00107351	0.813	[0.687; 0.927]	0.933	0.767	0.800	0.920
Gauss_lf_e__ep_b16_d2_avg	3.36	[3.19; 3.46]	3.04	[2.73; 3.28]	0.00011152	0.813	[0.697; 0.907]	1.000	0.500	0.667	1.000
Gauss_lp_e_nd__ep_b16_d3_avg	1.46	[1.29; 1.56]	1.24	[1.14; 1.36]	0.00014119	0.813	[0.696; 0.910]	0.533	0.967	0.941	0.674
Inv_Cluster_d_e__ep_b16_d3_avg	0.64	[0.61; 0.67]	0.60	[0.56; 0.62]	0.00011185	0.813	[0.701; 0.912]	1.000	0.467	0.652	1.000
Homogeneity2_e__ep_b2_d1_avg	1.40	[1.38; 1.42]	1.44	[1.42; 1.46]	0.00028432	0.812	[0.696; 0.910]	1.000	0.500	0.667	1.000
Homogeneity1_e__ep_b2_d1_avg	1.40	[1.38; 1.42]	1.44	[1.42; 1.46]	0.00028432	0.812	[0.694; 0.912]	1.000	0.500	0.667	1.000
Autocorrelation_e__ep_b2_d1_avg	4.04	[3.98; 4.10]	4.19	[4.11; 4.26]	0.00025560	0.812	[0.694; 0.910]	1.000	0.500	0.667	1.000
Cluster_p_nd__ep_b2_d1_avg	6.34	[5.92; 6.98]	7.58	[7.02; 8.09]	0.00078079	0.812	[0.689; 0.920]	0.900	0.700	0.750	0.875
Homogeneity2_e__ep_b8_d1_avg	2.56	[2.52; 2.64]	2.43	[2.32; 2.49]	0.00064257	0.812	[0.689; 0.919]	0.867	0.767	0.788	0.852
Dissimilarity_e__ep_b8_d1_avg	8.19	[7.79; 8.92]	9.66	[8.91; 10.28]	0.00085316	0.812	[0.683; 0.923]	0.933	0.733	0.778	0.917
DN_e__ep_b8_d1_avg	1.02	[0.97; 1.12]	1.21	[1.11; 1.28]	0.00085316	0.812	[0.682; 0.923]	0.933	0.733	0.778	0.917
Dissimilarity__ep_b4_d1_avg	0.62	[0.60; 0.66]	0.72	[0.67; 0.79]	0.00069965	0.811	[0.683; 0.920]	0.967	0.667	0.744	0.952
DN__ep_b4_d1_avg	0.16	[0.15; 0.16]	0.18	[0.17; 0.20]	0.00069965	0.811	[0.682; 0.919]	0.967	0.667	0.744	0.952
Correlation__ep_b4_d1_avg	0.63	[0.59; 0.66]	0.53	[0.50; 0.59]	0.00089093	0.811	[0.682; 0.920]	1.000	0.633	0.732	1.000
Dif_average__ep_b4_d1_avg	0.62	[0.60; 0.66]	0.72	[0.67; 0.79]	0.00069965	0.811	[0.680; 0.921]	0.967	0.667	0.744	0.952
Contrast_e__ep_b8_d1_avg	21.94	[19.90; 25.56]	28.59	[25.80; 31.01]	0.00068670	0.811	[0.679; 0.924]	0.933	0.733	0.778	0.917
DMN_e__ep_b8_d1_avg	0.34	[0.31; 0.40]	0.45	[0.40; 0.48]	0.00068670	0.811	[0.679; 0.926]	0.933	0.733	0.778	0.917
Cluster_d_s_nd__ep_b8_d2_avg	0.11	[0.11; 0.12]	0.12	[0.12; 0.13]	0.00063538	0.811	[0.691; 0.913]	0.667	0.900	0.870	0.730
Gauss_lf_e_nd__ep_b16_d3_avg	2.92	[2.70; 3.00]	2.56	[2.38; 2.82]	0.00010129	0.811	[0.690; 0.904]	1.000	0.500	0.667	1.000
Contrast_s__ep_b32_d1_avg	0.06	[0.05; 0.07]	0.10	[0.07; 0.13]	0.00127562	0.811	[0.694; 0.911]	0.733	0.833	0.815	0.758
DMN_s__ep_b32_d1_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00127562	0.811	[0.693; 0.911]	0.733	0.833	0.815	0.758
Gauss_lp_e__ep_b8_d2_avg	1.57	[1.48; 1.65]	1.39	[1.29; 1.52]	0.00014081	0.810	[0.698; 0.907]	0.900	0.567	0.675	0.850
Contrast_e__ep_b16_d1_avg	108.91	[97.41; 126.40]	142.49	[125.01; 155.56]	0.00075693	0.810	[0.680; 0.923]	0.967	0.700	0.763	0.955
Homogeneity2_nd__ep_b16_d1_avg	0.18	[0.17; 0.18]	0.16	[0.15; 0.17]	0.00047650	0.810	[0.686; 0.921]	0.867	0.733	0.765	0.846
Homogeneity1_e_nd__ep_b16_d1_avg	1.86	[1.85; 1.89]	1.79	[1.75; 1.86]	0.00028835	0.810	[0.689; 0.908]	0.933	0.600	0.700	0.900

DMN_e__ep_b16_d1_avg	0.43	[0.38; 0.49]	0.56	[0.49; 0.61]	0.00075693	0.810	[0.679; 0.922]	0.967	0.700	0.763	0.955
Cluster_p_s__ep_b16_d3_avg	687.00	[608.33; 822.64]	1021.47	[752.03; 1522.02]	0.00059338	0.810	[0.694; 0.906]	0.867	0.600	0.684	0.818
Dif_entropy__ep_b32_d1_avg	3.88	[3.81; 3.97]	4.07	[3.98; 4.14]	0.00154169	0.810	[0.680; 0.922]	0.933	0.733	0.778	0.917
Dissimilarity__ep_b8_d1_avg	1.33	[1.25; 1.41]	1.54	[1.42; 1.67]	0.00074400	0.809	[0.680; 0.919]	0.967	0.667	0.744	0.952
DN__ep_b8_d1_avg	0.17	[0.16; 0.18]	0.19	[0.18; 0.21]	0.00074400	0.809	[0.680; 0.921]	0.967	0.667	0.744	0.952
Dif_average__ep_b8_d1_avg	1.33	[1.25; 1.41]	1.54	[1.42; 1.67]	0.00074400	0.809	[0.681; 0.920]	0.967	0.667	0.744	0.952
Autocorrelation_s_nd__ep_b8_d2_avg	0.24	[0.22; 0.25]	0.26	[0.25; 0.31]	0.00138123	0.809	[0.693; 0.910]	0.900	0.667	0.730	0.870
Gauss_lf_e_nd__ep_b8_d3_avg	1.99	[1.91; 2.08]	1.87	[1.76; 1.95]	0.00018147	0.809	[0.696; 0.906]	0.933	0.567	0.683	0.895
Dissimilarity__ep_b16_d1_avg	2.71	[2.53; 2.87]	3.11	[2.87; 3.39]	0.00078434	0.809	[0.679; 0.920]	0.967	0.667	0.744	0.952
DN__ep_b16_d1_avg	0.17	[0.16; 0.18]	0.19	[0.18; 0.21]	0.00078434	0.809	[0.684; 0.920]	0.967	0.667	0.744	0.952
Dif_average__ep_b16_d1_avg	2.71	[2.53; 2.87]	3.11	[2.87; 3.39]	0.00078434	0.809	[0.679; 0.921]	0.967	0.667	0.744	0.952
Dissimilarity__ep_b32_d1_avg	5.44	[5.08; 5.75]	6.24	[5.77; 6.81]	0.00074720	0.809	[0.681; 0.917]	0.967	0.667	0.744	0.952
DN__ep_b32_d1_avg	0.17	[0.16; 0.18]	0.20	[0.18; 0.21]	0.00074720	0.809	[0.677; 0.920]	0.967	0.667	0.744	0.952
Dif_average__ep_b32_d1_avg	5.44	[5.08; 5.75]	6.24	[5.77; 6.81]	0.00074720	0.809	[0.681; 0.922]	0.967	0.667	0.744	0.952
Gauss_2f_e_nd__ep_b32_d2_avg	6.69	[6.44; 6.82]	6.24	[5.93; 6.52]	0.00059038	0.809	[0.691; 0.909]	0.733	0.800	0.786	0.750
Homogeneity1_e__ep_b8_d1_avg	2.88	[2.84; 2.93]	2.79	[2.72; 2.84]	0.00063031	0.808	[0.689; 0.910]	0.767	0.767	0.767	0.767
Inv_Cluster_t_e_nd__ep_b8_d3_avg	0.12	[0.12; 0.13]	0.11	[0.11; 0.12]	0.00023320	0.808	[0.691; 0.907]	0.967	0.500	0.659	0.938
Dissimilarity_e__ep_b16_d1_avg	21.65	[20.42; 23.29]	25.13	[22.82; 26.55]	0.00113649	0.808	[0.679; 0.922]	1.000	0.633	0.732	1.000
DN_e__ep_b16_d1_avg	1.35	[1.28; 1.46]	1.57	[1.43; 1.66]	0.00113649	0.808	[0.678; 0.919]	1.000	0.633	0.732	1.000
Contrast_s__ep_b32_d2_avg	0.15	[0.12; 0.22]	0.23	[0.19; 0.32]	0.00307124	0.808	[0.682; 0.911]	0.700	0.933	0.913	0.757
Homogeneity1_e_nd__ep_b32_d2_avg	1.52	[1.44; 1.57]	1.39	[1.31; 1.48]	0.00088722	0.808	[0.687; 0.908]	0.733	0.800	0.786	0.750
DMN_s__ep_b32_d2_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00307124	0.808	[0.683; 0.914]	0.700	0.933	0.913	0.757
Tri_mean__ep_b2_d1_avg	0.17	[0.16; 0.18]	0.18	[0.17; 0.19]	0.00025636	0.807	[0.688; 0.904]	0.867	0.667	0.722	0.833
Mn_AD_mn__ep_b4_d1_avg	0.04	[0.03; 0.04]	0.03	[0.03; 0.03]	0.00078729	0.807	[0.678; 0.918]	1.000	0.600	0.714	1.000
Gauss_rf_s_nd__ep_b8_d3_avg	0.00	[0.00; 0.00]	0.01	[0.00; 0.01]	0.00061390	0.807	[0.690; 0.903]	0.967	0.500	0.659	0.938

Cluster_d_s__ep_b8_d3_avg	0.15	[0.14; 0.16]	0.17	[0.15; 0.18]	0.00082661	0.807	[0.693; 0.906]	0.567	0.933	0.895	0.683
Contrast_s__ep_b16_d1_avg	0.05	[0.04; 0.05]	0.07	[0.05; 0.08]	0.00056312	0.807	[0.688; 0.907]	0.733	0.800	0.786	0.750
DMN_s__ep_b16_d1_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00056312	0.807	[0.689; 0.908]	0.733	0.800	0.786	0.750
Gauss_lf_e__ep_b16_d3_avg	3.25	[2.96; 3.31]	2.80	[2.58; 3.08]	0.00013944	0.807	[0.689; 0.902]	1.000	0.500	0.667	1.000
Cluster_d_s_nd__ep_b32_d2_avg	0.05	[0.04; 0.06]	0.06	[0.06; 0.09]	0.00087226	0.807	[0.691; 0.906]	0.700	0.833	0.808	0.735
Cluster_s_e_nd__ep_b2_d1_avg	8.50	[8.19; 8.91]	9.26	[8.93; 9.54]	0.00174377	0.806	[0.681; 0.914]	0.867	0.733	0.765	0.846
IDN__ep_b8_d1_avg	0.87	[0.87; 0.88]	0.86	[0.84; 0.87]	0.00093098	0.806	[0.677; 0.918]	0.933	0.700	0.757	0.913
Gauss_2f_e__ep_b16_d2_avg	5.90	[5.77; 5.96]	5.63	[5.44; 5.85]	0.00063737	0.806	[0.684; 0.907]	0.667	0.833	0.800	0.714
Gauss_lp_e__ep_b16_d3_avg	1.64	[1.46; 1.79]	1.40	[1.27; 1.53]	0.00018975	0.806	[0.692; 0.904]	0.933	0.533	0.667	0.889
Homogeneity1_e_nd__ep_b32_d1_avg	1.81	[1.76; 1.86]	1.67	[1.57; 1.76]	0.00068272	0.806	[0.680; 0.908]	0.767	0.800	0.793	0.774
Inv_Gauss_lf_s__ep_b32_d3_avg	0.04	[0.03; 0.04]	0.06	[0.05; 0.11]	0.00095605	0.806	[0.687; 0.906]	0.800	0.767	0.774	0.793
Inv_dif_variance__ep_b2_d1_avg	11.71	[6.68; 18.45]	30.96	[17.65; 127.28]	0.31728832	0.804	[0.682; 0.911]	0.900	0.633	0.711	0.864
Inv_Gauss_e_nd__ep_b4_d1_avg	4.12	[3.97; 4.36]	4.67	[4.36; 4.91]	0.00103725	0.804	[0.676; 0.919]	0.967	0.700	0.763	0.955
Gauss_lp_e_nd__ep_b8_d3_avg	1.11	[1.03; 1.19]	1.00	[0.92; 1.07]	0.00020773	0.804	[0.688; 0.900]	0.967	0.467	0.644	0.933
Gauss_lf_e__ep_b8_d3_avg	2.50	[2.31; 2.55]	2.22	[2.10; 2.41]	0.00026051	0.804	[0.684; 0.902]	1.000	0.500	0.667	1.000
Average_s__ep_b8_d3_avg	0.07	[0.07; 0.08]	0.08	[0.08; 0.09]	0.00086817	0.804	[0.688; 0.904]	0.567	0.933	0.895	0.683
IDN__ep_b4_d1_avg	0.88	[0.88; 0.89]	0.87	[0.86; 0.88]	0.00092975	0.803	[0.674; 0.916]	0.900	0.733	0.771	0.880
Autocorrelation_s_nd__ep_b8_d3_avg	0.24	[0.22; 0.27]	0.29	[0.26; 0.32]	0.00107192	0.803	[0.684; 0.901]	0.900	0.567	0.675	0.850
Cluster_t_s_nd__ep_b16_d2_avg	1.25	[1.14; 1.34]	1.51	[1.28; 1.81]	0.00082373	0.803	[0.684; 0.900]	0.933	0.600	0.700	0.900
Gauss_e_nd__ep_b4_d1_avg	1.19	[1.15; 1.22]	1.27	[1.23; 1.32]	0.00162978	0.802	[0.672; 0.914]	0.900	0.733	0.771	0.880
Gauss_lp_e__ep_b4_d2_avg	1.31	[1.26; 1.35]	1.21	[1.14; 1.30]	0.00014855	0.802	[0.684; 0.899]	1.000	0.467	0.652	1.000
Gauss_lf_e__ep_b4_d3_avg	1.73	[1.64; 1.75]	1.61	[1.56; 1.69]	0.00044460	0.802	[0.680; 0.902]	0.967	0.533	0.674	0.941
Cluster_t_s_nd__ep_b8_d3_avg	1.08	[0.99; 1.21]	1.26	[1.16; 1.48]	0.00070981	0.802	[0.684; 0.899]	0.600	0.867	0.818	0.684
Gauss_rp_s__ep_b16_d3_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00079455	0.802	[0.684; 0.902]	0.900	0.633	0.711	0.864
Sum_energy__ep_b16_d3_avg	0.65	[0.61; 0.69]	0.71	[0.68; 0.76]	0.00049550	0.802	[0.681; 0.904]	0.800	0.700	0.727	0.778

Variance_s__ep_b32_d2_avg	0.52	[0.46; 0.68]	0.80	[0.64; 1.28]	0.00047100	0.802	[0.684; 0.899]	0.933	0.500	0.651	0.882
Homogeneity2_e__ep_b4_d1_avg	2.34	[2.31; 2.35]	2.28	[2.24; 2.32]	0.00067366	0.801	[0.679; 0.907]	0.667	0.867	0.833	0.722
Contrast_s__ep_b8_d1_avg	0.04	[0.04; 0.05]	0.05	[0.05; 0.07]	0.00045913	0.801	[0.677; 0.906]	0.833	0.700	0.735	0.808
DMN_s__ep_b8_d1_avg	0.00	[0.00; 0.00]	0.00	[0.00; 0.00]	0.00045913	0.801	[0.676; 0.907]	0.833	0.700	0.735	0.808
Uniformity__ep_b16_d2_avg	0.08	[0.06; 0.11]	0.13	[0.11; 0.17]	0.00114574	0.801	[0.680; 0.909]	0.733	0.800	0.786	0.750
Gauss_lf_e__ep_b32_d1_avg	4.12	[4.01; 4.26]	3.93	[3.56; 4.07]	0.00017479	0.801	[0.681; 0.902]	0.967	0.533	0.674	0.941
Cluster_p_s_nd__ep_b32_d2_avg	3297.66	[2815.16; 4243.05]	5156.04	[3688.72; 8551.98]	0.00091852	0.801	[0.681; 0.898]	0.533	0.933	0.889	0.667
Homogeneity2__ep_b4_d1_avg	0.71	[0.70; 0.73]	0.68	[0.65; 0.70]	0.00103465	0.800	[0.669; 0.913]	0.933	0.667	0.737	0.909
Cluster_d_e_nd__ep_b4_d1_avg	9.95	[9.37; 10.29]	10.85	[10.16; 11.50]	0.00143029	0.800	[0.671; 0.911]	0.933	0.633	0.718	0.905
Tri_mean__ep_b4_d1_avg	0.04	[0.04; 0.04]	0.05	[0.04; 0.05]	0.00114398	0.800	[0.677; 0.910]	1.000	0.533	0.682	1.000
Md_AD_md__ep_b8_d1_avg	0.01	[0.01; 0.01]	0.01	[0.01; 0.01]	0.00105766	0.800	[0.680; 0.901]	0.967	0.500	0.659	0.938
MAD__ep_b8_d1_avg	0.01	[0.01; 0.01]	0.01	[0.01; 0.01]	0.00105766	0.800	[0.681; 0.904]	0.967	0.500	0.659	0.938
Uniformity__ep_b8_d3_avg	0.06	[0.05; 0.08]	0.09	[0.08; 0.12]	0.00147930	0.800	[0.676; 0.903]	0.667	0.900	0.870	0.730
IDN__ep_b32_d1_avg	0.87	[0.86; 0.87]	0.85	[0.84; 0.86]	0.00090753	0.800	[0.670; 0.912]	0.933	0.700	0.757	0.913
Gauss_2f_e__ep_b32_d1_avg	7.61	[7.46; 7.71]	7.29	[6.99; 7.50]	0.00096657	0.800	[0.680; 0.901]	0.700	0.800	0.778	0.727
GLRLM											
SRLGLE__ep_b4_avg	0.20	[0.19; 0.21]	0.23	[0.22; 0.27]	0.00001347	0.918	[0.822; 0.996]	1.000	0.867	0.882	1.000
LRLGLE__ep_b2_avg	6.34	[5.35; 7.30]	3.56	[2.95; 4.32]	0.00005396	0.894	[0.799; 0.970]	1.000	0.733	0.789	1.000
LRE__ep_b2_avg	9.15	[7.86; 10.58]	5.32	[4.78; 6.27]	0.00008797	0.888	[0.791; 0.962]	0.933	0.767	0.800	0.920
LRLGLE__ep_b4_avg	1.65	[1.48; 1.74]	1.13	[0.99; 1.23]	0.00011617	0.888	[0.778; 0.974]	0.967	0.867	0.879	0.963
SRLGLE__ep_b2_avg	0.29	[0.27; 0.31]	0.38	[0.33; 0.44]	0.00004815	0.881	[0.783; 0.957]	0.900	0.767	0.794	0.885
SRLGLE__ep_b8_avg	0.12	[0.11; 0.13]	0.14	[0.13; 0.15]	0.00006380	0.879	[0.769; 0.968]	1.000	0.767	0.811	1.000
LRE__ep_b4_avg	3.81	[3.51; 4.34]	2.75	[2.57; 3.21]	0.00034104	0.874	[0.772; 0.957]	1.000	0.733	0.789	1.000
RP__ep_b2_avg	0.46	[0.42; 0.49]	0.57	[0.51; 0.60]	0.00006535	0.871	[0.771; 0.951]	1.000	0.667	0.750	1.000
LRHGLE__ep_b2_avg	19.67	[16.78; 23.82]	12.46	[11.44; 14.13]	0.00019445	0.870	[0.768; 0.953]	0.933	0.767	0.800	0.920

RP__ep_b4_avg	0.64	[0.60; 0.67]	0.73	[0.69; 0.76]	0.00014208	0.859	[0.751; 0.947]	0.900	0.767	0.794	0.885
LRE__ep_b8_avg	2.14	[1.98; 2.30]	1.77	[1.66; 1.91]	0.00085685	0.854	[0.742; 0.944]	0.967	0.700	0.763	0.955
LRLGLE__ep_b8_avg	0.53	[0.46; 0.56]	0.40	[0.37; 0.44]	0.00048520	0.854	[0.733; 0.957]	0.933	0.800	0.824	0.923
SRE__ep_b32_avg	0.95	[0.95; 0.96]	0.97	[0.96; 0.97]	0.00029171	0.853	[0.747; 0.942]	1.000	0.633	0.732	1.000
RP__ep_b8_avg	0.79	[0.76; 0.81]	0.83	[0.82; 0.86]	0.00032175	0.852	[0.740; 0.943]	0.967	0.733	0.784	0.957
SRE__ep_b8_avg	0.85	[0.83; 0.86]	0.89	[0.87; 0.90]	0.00022819	0.851	[0.740; 0.939]	0.967	0.667	0.744	0.952
SRE__ep_b4_avg	0.74	[0.70; 0.76]	0.81	[0.76; 0.83]	0.00017530	0.844	[0.737; 0.936]	0.967	0.633	0.725	0.950
SRE__ep_b16_avg	0.91	[0.91; 0.93]	0.94	[0.93; 0.94]	0.00037202	0.844	[0.731; 0.938]	0.933	0.733	0.778	0.917
LGLRE__ep_b8_avg	0.16	[0.16; 0.16]	0.17	[0.17; 0.18]	0.00035794	0.833	[0.710; 0.936]	0.933	0.767	0.800	0.920
LGLRE__ep_b32_avg	0.04	[0.04; 0.04]	0.05	[0.04; 0.05]	0.00053183	0.833	[0.718; 0.930]	0.900	0.700	0.750	0.875
SRE__ep_b2_avg	0.59	[0.53; 0.60]	0.68	[0.60; 0.71]	0.00022688	0.832	[0.719; 0.927]	0.900	0.700	0.750	0.875
RP__ep_b32_avg	0.94	[0.93; 0.94]	0.95	[0.94; 0.96]	0.00053179	0.830	[0.714; 0.927]	1.000	0.567	0.698	1.000
SRLGLE__ep_b32_avg	0.04	[0.03; 0.04]	0.04	[0.04; 0.04]	0.00041122	0.830	[0.714; 0.927]	0.933	0.667	0.737	0.909
RP__ep_b16_avg	0.88	[0.87; 0.89]	0.91	[0.89; 0.92]	0.00063217	0.829	[0.712; 0.924]	0.967	0.600	0.707	0.947
LRHGLE__ep_b4_avg	26.30	[23.66; 31.66]	19.99	[19.00; 23.42]	0.00191054	0.826	[0.711; 0.920]	0.900	0.667	0.730	0.870
LRE__ep_b32_avg	1.22	[1.19; 1.27]	1.17	[1.14; 1.20]	0.00109687	0.822	[0.706; 0.919]	0.867	0.733	0.765	0.846
SRLGLE__ep_b16_avg	0.07	[0.06; 0.07]	0.08	[0.07; 0.08]	0.00057258	0.820	[0.701; 0.924]	0.867	0.767	0.788	0.852
LRE__ep_b16_avg	1.49	[1.42; 1.57]	1.35	[1.30; 1.41]	0.00147469	0.817	[0.701; 0.918]	0.800	0.767	0.774	0.793
LRLGLE__ep_b16_avg	0.19	[0.17; 0.21]	0.16	[0.15; 0.18]	0.00067542	0.814	[0.693; 0.914]	0.933	0.633	0.718	0.905
LRLGLE__ep_b32_avg	0.08	[0.07; 0.09]	0.07	[0.06; 0.08]	0.00061639	0.808	[0.689; 0.906]	0.700	0.833	0.808	0.735
LGLRE__ep_b16_avg	0.08	[0.08; 0.09]	0.09	[0.09; 0.09]	0.00161389	0.802	[0.672; 0.912]	0.833	0.800	0.806	0.828
Geometry based parameters											
s_ratio_to_all_2__ep_2	0.90	[0.85; 0.96]	0.80	[0.73; 0.84]	0.00006297	0.890	[0.801; 0.960]	0.833	0.833	0.833	0.833
s_ratio_to_all_7__ep_8	0.40	[0.36; 0.46]	0.31	[0.27; 0.35]	0.00004832	0.888	[0.796; 0.958]	0.933	0.733	0.778	0.917
s_ratio_to_all_22__ep_32	0.12	[0.11; 0.14]	0.09	[0.08; 0.10]	0.00005196	0.883	[0.787; 0.959]	0.767	0.900	0.885	0.794

s_ratio_to_all_14__ep_16	0.22	[0.20; 0.25]	0.17	[0.14; 0.19]	0.00006811	0.882	[0.790; 0.954]	0.833	0.833	0.833	0.833
s_ratio_to_all_16__ep_32	0.11	[0.10; 0.13]	0.09	[0.08; 0.10]	0.00005204	0.882	[0.789; 0.957]	0.967	0.700	0.763	0.955
s_ratio_to_all_11__ep_16	0.22	[0.20; 0.26]	0.17	[0.16; 0.19]	0.00006581	0.881	[0.787; 0.958]	0.767	0.867	0.852	0.788
s_ratio_to_all_27__ep_32	0.11	[0.11; 0.13]	0.09	[0.08; 0.10]	0.00019188	0.876	[0.777; 0.954]	0.900	0.700	0.750	0.875
s_ratio_to_all_25__ep_32	0.11	[0.11; 0.14]	0.09	[0.08; 0.10]	0.00004843	0.874	[0.780; 0.949]	0.667	0.933	0.909	0.737
s_ratio_to_all_6__ep_8	0.40	[0.37; 0.46]	0.32	[0.29; 0.35]	0.00007621	0.871	[0.772; 0.950]	0.867	0.800	0.812	0.857
s_ratio_to_all_8__ep_16	0.21	[0.19; 0.26]	0.17	[0.15; 0.18]	0.00008007	0.871	[0.777; 0.948]	0.867	0.767	0.788	0.852
s_ratio_to_all_13__ep_16	0.22	[0.20; 0.27]	0.17	[0.15; 0.20]	0.00005517	0.870	[0.772; 0.947]	0.933	0.633	0.718	0.905
s_ratio_to_all_28__ep_32	0.12	[0.11; 0.14]	0.09	[0.08; 0.10]	0.00004802	0.869	[0.774; 0.946]	0.800	0.800	0.800	0.800
s_ratio_to_all_13__ep_32	0.11	[0.10; 0.13]	0.09	[0.08; 0.10]	0.00008194	0.868	[0.764; 0.950]	0.800	0.833	0.828	0.806
s_ratio_to_all_7__ep_16	0.21	[0.20; 0.26]	0.17	[0.15; 0.18]	0.00011391	0.867	[0.761; 0.952]	0.833	0.833	0.833	0.833
s_ratio_to_all_3__ep_4	0.65	[0.61; 0.73]	0.54	[0.50; 0.59]	0.00008614	0.864	[0.767; 0.943]	0.867	0.767	0.788	0.852
s_ratio_to_all_10__ep_16	0.21	[0.20; 0.26]	0.17	[0.15; 0.19]	0.00011581	0.861	[0.761; 0.944]	0.767	0.867	0.852	0.788
s_ratio_to_all_29__ep_32	0.11	[0.10; 0.13]	0.08	[0.08; 0.10]	0.00009006	0.860	[0.760; 0.942]	0.833	0.767	0.781	0.821
s_ratio_to_all_12__ep_16	0.23	[0.20; 0.26]	0.17	[0.15; 0.19]	0.00011572	0.858	[0.752; 0.940]	0.833	0.800	0.806	0.828
s_ratio_to_all_15__ep_16	0.22	[0.19; 0.24]	0.16	[0.15; 0.18]	0.00010811	0.858	[0.751; 0.942]	0.833	0.800	0.806	0.828
s_ratio_to_all_14__ep_32	0.12	[0.11; 0.13]	0.09	[0.08; 0.10]	0.00021378	0.858	[0.753; 0.948]	0.900	0.733	0.771	0.880
s_ratio_to_all_24__ep_32	0.12	[0.11; 0.14]	0.09	[0.08; 0.10]	0.00016338	0.857	[0.754; 0.942]	0.867	0.767	0.788	0.852
s_ratio_to_all_4__ep_8	0.37	[0.34; 0.45]	0.30	[0.28; 0.34]	0.00013644	0.856	[0.749; 0.940]	0.833	0.767	0.781	0.821
s_ratio_to_all_5__ep_8	0.39	[0.36; 0.44]	0.31	[0.29; 0.35]	0.00015322	0.856	[0.754; 0.938]	0.800	0.767	0.774	0.793
s_ratio_to_all_4__ep_4	0.59	[0.55; 0.65]	0.48	[0.41; 0.53]	0.00019620	0.854	[0.751; 0.938]	0.800	0.767	0.774	0.793
s_ratio_to_all_9__ep_16	0.21	[0.19; 0.26]	0.17	[0.16; 0.19]	0.00028789	0.854	[0.747; 0.941]	0.833	0.767	0.781	0.821
s_ratio_to_all_20__ep_32	0.11	[0.10; 0.14]	0.09	[0.07; 0.10]	0.00015859	0.854	[0.751; 0.939]	0.867	0.733	0.765	0.846
s_ratio_to_all_30__ep_32	0.11	[0.10; 0.13]	0.09	[0.08; 0.10]	0.00014978	0.853	[0.754; 0.936]	0.767	0.800	0.793	0.774
s_ratio_to_all_21__ep_32	0.12	[0.10; 0.14]	0.09	[0.08; 0.10]	0.00021819	0.852	[0.749; 0.939]	0.833	0.767	0.781	0.821

s_ratio_to_all_15__ep_32	0.12	[0.11; 0.14]	0.09	[0.08; 0.10]	0.00021498	0.850	[0.741; 0.937]	0.800	0.767	0.774	0.793
s_ratio_to_all_19__ep_32	0.11	[0.10; 0.13]	0.09	[0.08; 0.10]	0.00014215	0.849	[0.742; 0.934]	0.700	0.900	0.875	0.750
s_ratio_to_all_26__ep_32	0.11	[0.10; 0.15]	0.09	[0.08; 0.10]	0.00013039	0.848	[0.742; 0.933]	0.967	0.567	0.690	0.944
s_ratio_to_all_10__ep_32	0.11	[0.10; 0.14]	0.08	[0.08; 0.10]	0.00015899	0.844	[0.737; 0.927]	1.000	0.567	0.698	1.000
s_ratio_to_all_3__ep_32	0.10	[0.10; 0.13]	0.08	[0.08; 0.09]	0.00014020	0.839	[0.730; 0.932]	0.767	0.867	0.852	0.788
s_ratio_to_all_5__ep_32	0.11	[0.10; 0.13]	0.08	[0.08; 0.10]	0.00004671	0.839	[0.732; 0.930]	0.733	0.867	0.846	0.765
s_ratio_to_all_2__ep_32	0.10	[0.09; 0.12]	0.08	[0.07; 0.09]	0.00023945	0.837	[0.729; 0.927]	0.933	0.600	0.700	0.900
s_ratio_to_all_8__ep_32	0.12	[0.10; 0.14]	0.09	[0.08; 0.10]	0.00019267	0.837	[0.728; 0.927]	0.700	0.867	0.840	0.743
s_ratio_to_all_31__ep_32	0.11	[0.10; 0.13]	0.08	[0.07; 0.10]	0.00025030	0.837	[0.726; 0.927]	0.800	0.767	0.774	0.793
s_ratio_to_all_4__ep_16	0.20	[0.18; 0.25]	0.17	[0.15; 0.18]	0.00026688	0.832	[0.719; 0.927]	0.633	0.933	0.905	0.718
s_ratio_to_all_5__ep_16	0.20	[0.19; 0.26]	0.16	[0.15; 0.19]	0.00017100	0.831	[0.717; 0.923]	0.967	0.567	0.690	0.944
s_ratio_to_all_6__ep_16	0.22	[0.19; 0.26]	0.17	[0.15; 0.19]	0.00018603	0.831	[0.719; 0.926]	0.733	0.833	0.815	0.758
s_ratio_to_all_23__ep_32	0.11	[0.10; 0.14]	0.09	[0.08; 0.10]	0.00035395	0.831	[0.718; 0.923]	0.800	0.767	0.774	0.793
s_ratio_to_all_12__ep_32	0.11	[0.10; 0.13]	0.09	[0.08; 0.10]	0.00039034	0.830	[0.717; 0.923]	0.667	0.867	0.833	0.722
s_ratio_to_all_7__ep_32	0.11	[0.10; 0.12]	0.08	[0.08; 0.10]	0.00068719	0.829	[0.710; 0.928]	0.900	0.733	0.771	0.880
s_ratio_to_all_6__ep_32	0.12	[0.10; 0.14]	0.08	[0.08; 0.09]	0.00029502	0.827	[0.712; 0.924]	0.800	0.767	0.774	0.793
s_ratio_to_all_3__ep_16	0.21	[0.18; 0.24]	0.16	[0.14; 0.18]	0.00014313	0.823	[0.707; 0.919]	0.633	0.933	0.905	0.718
s_ratio_to_all_17__ep_32	0.12	[0.10; 0.13]	0.09	[0.08; 0.10]	0.00057304	0.820	[0.704; 0.919]	0.900	0.667	0.730	0.870
fractal_bc_d_3__ep_32	1.11	[1.06; 1.18]	1.01	[0.96; 1.06]	0.00036927	0.817	[0.702; 0.913]	0.767	0.733	0.742	0.759
s_ratio_to_all_18__ep_32	0.11	[0.10; 0.15]	0.09	[0.08; 0.10]	0.00037915	0.816	[0.700; 0.912]	0.833	0.700	0.735	0.808
fractal_bc_d_8__ep_32	1.13	[1.07; 1.18]	1.03	[0.98; 1.07]	0.00051135	0.816	[0.696; 0.919]	0.667	0.933	0.909	0.737
s_ratio_to_all_3__ep_8	0.36	[0.34; 0.45]	0.31	[0.28; 0.34]	0.00026422	0.814	[0.699; 0.914]	0.767	0.767	0.767	0.767
s_ratio_to_all_11__ep_32	0.12	[0.10; 0.13]	0.09	[0.08; 0.10]	0.00017438	0.814	[0.694; 0.917]	0.767	0.800	0.793	0.774
s_ratio_to_all_8__ep_8	0.35	[0.31; 0.38]	0.27	[0.22; 0.30]	0.00064242	0.811	[0.689; 0.917]	0.733	0.867	0.846	0.765
s_ratio_to_all_9__ep_32	0.11	[0.10; 0.12]	0.09	[0.08; 0.10]	0.00039948	0.809	[0.688; 0.911]	0.767	0.767	0.767	0.767

surface_volume_r_1__orig	3.49	[3.11; 4.03]	4.82	[4.01; 5.39]	0.00040076	0.807	[0.689; 0.911]	0.933	0.600	0.700	0.900

Data is presented as median with interquartile ranges or frequency and percentage of the most frequent element, as appropriate.

First-order statistical names are generated as: “statistic”_“orig” indicating calculation done on original images.

GLCM statistical names are generated as: “statistic”_“X”_“ep”_“N”_“D”_“avg”. X is either empty indicating no manipulation done on the GLCM matrix, or *s* for squared, where the GLCM element were squared or *e* where the entropy of the elements was used. ep: equal probability binning. N: the number of bins used. D: the distance of the reference and the observed voxels. “avg” indicates that statistics were averaged using all directions.

GLRLM statistical names are generated as: “statistic” _ “ep”_“N”_”avg”. ep: equal probability binning. N: the number of bins used. “avg” indicates that statistics were averaged using all directions.

Geometry based statistical names were generated as: “statistic”_”S”_“ep”_“N”. S: subcomponent used, 1 if original image was used. ep: equal probability binning. N: the number of bins used.

Supplemental References

1. Karady J, Panajotu A, Kolossvary M, Szilveszter B, Jermendy AL, Bartykowszki A, Karolyi M, Celeng C, Merkely B and Maurovich-Horvat P. The effect of four-phasic versus three-phasic contrast media injection protocols on extravasation rate in coronary CT angiography: a randomized controlled trial. *Eur Radiol*. 2017;27:4538-4543.
2. Shafiq-Ul-Hassan M, Zhang GG, Latifi K, Ullah G, Hunt DC, Balagurunathan Y, Abdalah MA, Schabath MB, Goldgof DG, Mackin D, Court LE, Gillies RJ and Moros EG. Intrinsic dependencies of CT radiomic features on voxel size and number of gray levels. *Med Phys*. 2017;44:1050-1062.
3. Haralick RM, Shanmugam K and Dinstein I. Textural Features for Image Classification. *IEEE Transactions on Systems, Man, and Cybernetics*. 1973;SMC-3:610-621.
4. Galloway MM. Texture analysis using gray level run lengths. *Computer Graphics and Image Processing*. 1975;4:172-179.