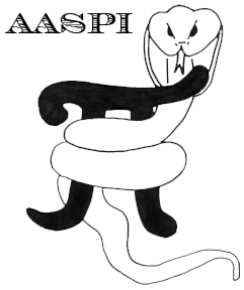


VOLUMETRIC INDEPENDENT COMPONENT ANALYSIS FOR 3D SEISMIC FACIES ANALYSIS – PROGRAM ica3d

Contents

Overview

Independent Component Analysis (ICA) uses higher order statistics to separate a multivariate data into independent components, finding a linear representation of non-Gaussian data (Hyvärinen and Oja, 2000). This representation allows the extraction of more valuable information than Principal Component Analysis in which the data is decomposed in linear uncorrelated components (Lubo-Robles, 2018).

To illustrate ICA, the popular cocktail-party problem is commonly used. If two people are speaking at the same time in a room where two microphones are recording their voices, the recorded signals are a weighted linear combination of the individual signals spoken by the two people. Although the goal is to estimate the original individual signals using the recorded signals, this cannot be accomplished because the relationship between the recorded and the individual signals is unknown. However, following Hyvärinen and Oja, (2000), it is possible to compute the individual signals under the assumption in which they are statistically independent.

The goal is to obtain an unmixing matrix **W**, which multiplied with the recorded signals, computes the individual signals or independent components. The ica3d algorithm is based on the FastICA algorithm developed by Hyvärinen and Oja, (2000), but with modifications in order to apply it using volumetric seismic attributes. Also, some preprocessing steps are applied to the input data in order to make the estimation of the independent components better conditioned (Hyvärinen and Oja, 2000).

Data preparation

Computing the Mean, Standard Deviation and the Correlation Matrix

The mean of any attribute a_j is simply the sum of its values at M voxels divided by M:

$$\mu_j = \frac{1}{M}\sum_{m=1}^{M} a_{jm} \,. \tag{1}$$

The standard deviation of each attribute a_j is computed as:

$$\sigma_j = \sqrt{\mathrm{E}[\,(a_j - \mu_j)^2]} \tag{2}$$

where, E is the average or expected value operator.

Contrary to human voices and other ICA applications, each seismic attribute may have a different unit of measurement and range of values. The seismic envelope may range between 0 and +10000, while curvature may have value that range between -1 and +1 km^{-1}. Therefore, Z-score normalization is applied to the data:

$a_j^{(\mathrm{norm})}=(a_j - \mu_j)/\sigma_j.$ (3)

The correlation matrix, **C**, is constructed by comparing each sample vector to itself and all its neighbors and can be computed from *K* attribute volumes as:

$$C_{kl} = \frac{1}{M}\sum_{m=1}^{M} a_{mk}^{(\mathrm{norm})}(t_m, x_m, y_m) a_{ml}^{(\mathrm{norm})}(t_m, x_m, y_m) \quad (4)$$

where *M* is number of voxels in the volume to be analyzed.

Eigenvectors and Eigenvalues

$$\sum_{j=1}^{J} A_{ij} v_j^{(k)} = \lambda_k v_i^{(k)}, \quad (5)$$

or in matrix form

$$\overline{\mathbf{C}}\mathbf{v}^{(k)} = \lambda_k \mathbf{v}^{(k)}, \quad (6)$$

where $\overline{\mathbf{C}}$ is a *J* by *J* square matrix, λ_k is the k^{th} eigenvalue and $\mathbf{v}^{(k)}$ is its corresponding eigenvector. In general, there are *J* eigenvalue-eigenvector pairs, though not all of them need to be different, and where some of the eigenvalues λ_k may equal 0, indicating a rank-deficient matrix. By convention, the eigenvectors are normalized to be unit vectors

$$\sum_{j=1}^{J} (v_j^{(k)})^2 \equiv 1, \quad (7)$$

while the eigenvalue-eigenvector pairs are sorted from largest to smallest

$$|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_J|, \quad (8)$$

Principal components and whitening

The k^{th} principal component, $\mathbf{p}^{(k)}$, at the m^{th} voxel (t_m,,x_m,ym_n) is a scalar value that represents the projection of an *J*-dimensional sample vector, **a**, against the k^{th} unit length, *J*-dimensional eigenvector, $\mathbf{v}^{(k)}$,:

$$P_j^{(k)}(t_m, x_{m,} y_m) = \sum_{j=1}^{J} a_j^{(\mathrm{norm})}(t_m, x_{m,} y_m) v_j^{(k)}. \quad (9)$$

To whiten the data, one simply normalizes by the inverse of the square root of the corresponding eigenvalues:

$$\hat{a}_j = \frac{P_j^{(k)}(t_m, x_{m,} y_m)}{(\lambda_j + \varepsilon)^{1/2}}, \quad (10)$$

where $\hat{a}_j$ are the Z-normalized whitened seismic attributes and ε is a fraction of the largest eigenvalue, λ_1 to avoid division by zero.

PCA whitening is used not only to reduce the dimensionality of the data but also to reduce noise. In addition, it guarantees that the data is uncorrelated, i.e., there is no a linear relationship between the input properties, thus simplifying the estimation of the independent component because reduces the number of free

In the ica3d algorithm, to decide how many components are preserved, the percentage of variance retained is analyzed (Stanford, 2018). If there are *n* principal components whose eigenvalues are $\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_n$ where $\lambda_j \geq \lambda_{j+1}$. For *n* attributes, the algorithm keeps those components whose sum just exceeds a defined percentage δ, of the variability of the data, while the remaining variability is considered to be noise:

$$\frac{\sum_{j=1}^{k} \lambda_j}{\sum_{j=1}^{n} \lambda_j} \geq \delta, \quad (11)$$

where in the algorithm a value δ = 0.9 is used to distinguish between data and noise.

Independent components estimation

Following Hyvärinen and Oja (2000) and based on the central limit theorem, is known that the distribution of two independent variables is less Gaussian than the distribution of the sum of the two variables. Thus, if the non-Gaussian behavior of the preprocessed data is maximize, it is possible to compute the unmixing matrix, **W**, in order to obtain the independent components. Also, to measure the non-Gaussianity of the data, a modified version of entropy called negentropy is used (Hyvärinen, 1999; Hyvärinen and Oja, 2000) and it is always nonnegative and equal to zero for a Gaussian distribution.

Assuming a random variable $y=W^T\hat{a}$ with zero mean and unit variance, negentropy *J* (Hyvärinen, 1999) approximates as:

$$J(y) = \{E[G(y)] - E[G(v)]\}^2, \quad (12)$$

where *v* is a centered and whitened Gaussian variable, E is the expected value operator which in practice is replaced by the sample means (Hyvärinen and Oja, 2000) and G is a non-quadratic function:

$$G(y) = -e^{-\left(y^2/2\right)}, \quad (13)$$

which according to Zanardo Honorio, et al., 2014 provides good resolution and delineation of geological features.

Following the FastICA algorithm developed by Hyvärinen and Oja (2000), to compute the independent components the contrast function, G is maximized. Also, the independent components are computed simultaneously and to avoid convergence to the same maxima, the outputs are decorrelated after each iteration (Hyvärinen and Oja, 2000).

In each iteration of the algorithm, each row of the the unmixing matrix, **W**, is updated by

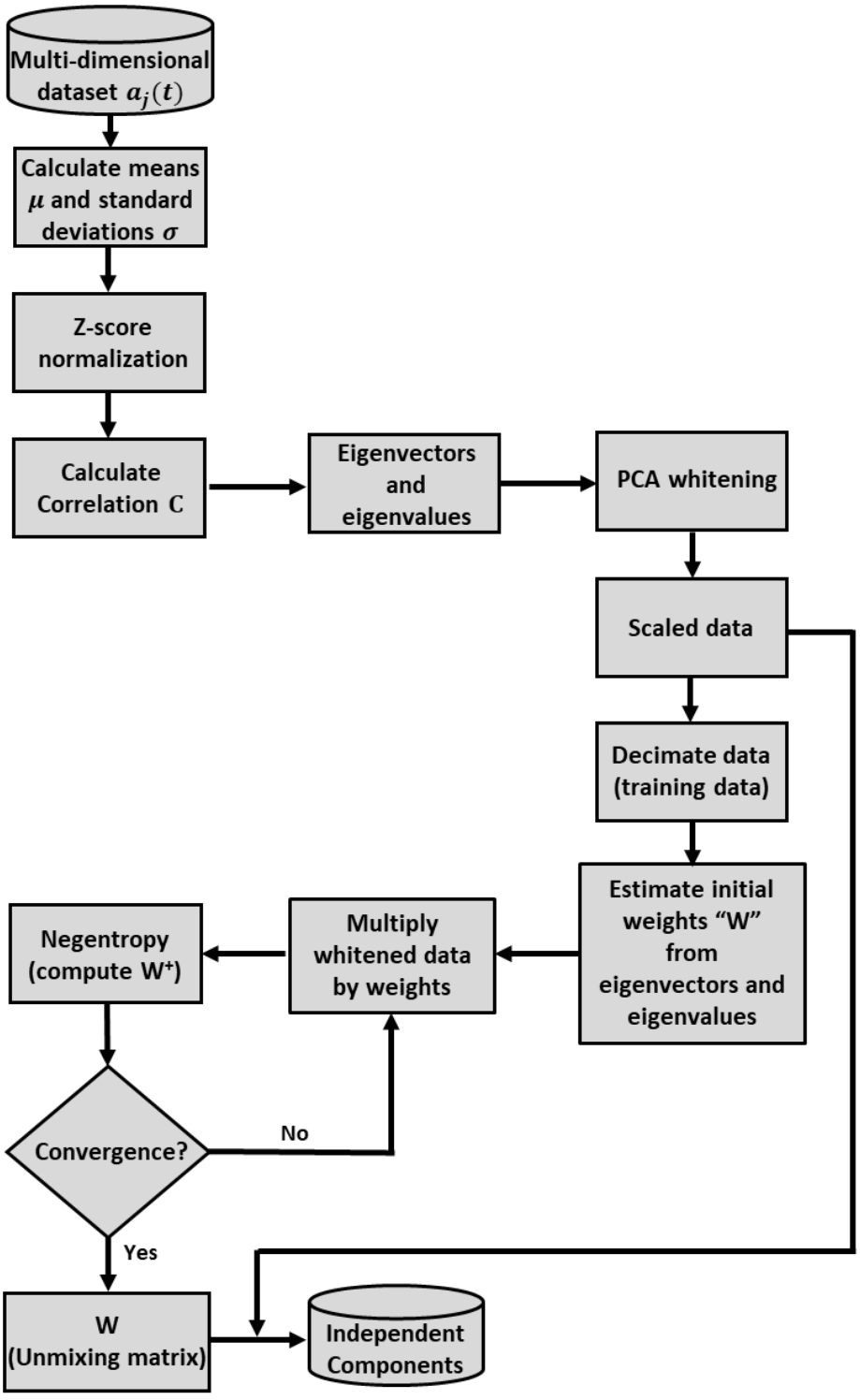
$$\mathrm{W}_j^+ = E\left[\hat{a}\,\frac{\partial G}{\partial \mathrm{a}_k}\left(\mathrm{W}_j^T\hat{a}\right)\right] - E\left[\frac{\partial^2 G}{\partial \mathrm{a}_k \partial \mathrm{a}_l}\left(\mathrm{W}_j^T\hat{a}\right)\right] W_j, \quad (14)$$

and normalized by:

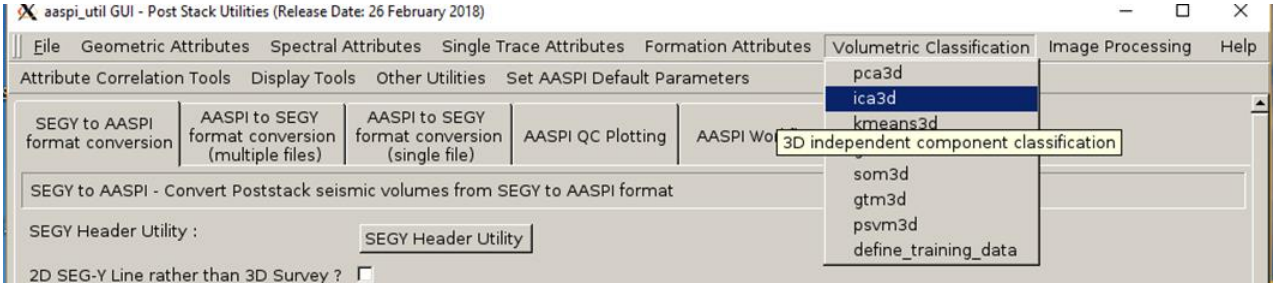
$$\mathrm{W}_j^+ = {\mathrm{W}_j^+}\Big/{||\mathrm{W}_j^+||}, \quad (15)$$

where, $\mathbf{W}^+$ is the updated unmixing matrix, and is decorrelated using Eigenvalue Decomposition (EVD) by

$$\mathrm{W}_{\mathrm{decorr}}^+ = (\mathrm{W}\mathrm{W}^{\mathrm{T}})^{-1/2}\mathrm{W}. \quad (16)$$



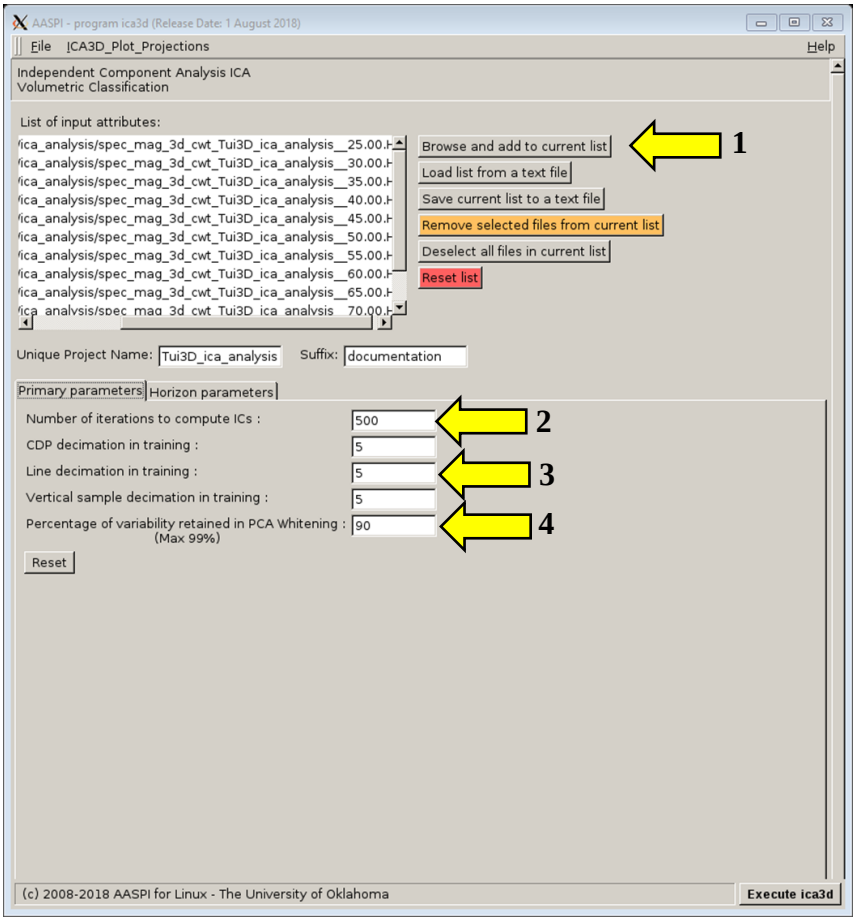
Workflow of the ica3d volumetric classification algorithm (Lubo-Robles, 2018). In order to decrease the computational cost of processing multiple seismic attributes, the data is decimated to construct a smaller training data volume from which the unmixing matrix **W** is computed. The algorithm is based on the FastICA algorithm developed by Hyvärinen and Oja (2000),but with modifications to implement it using seismic attributes.



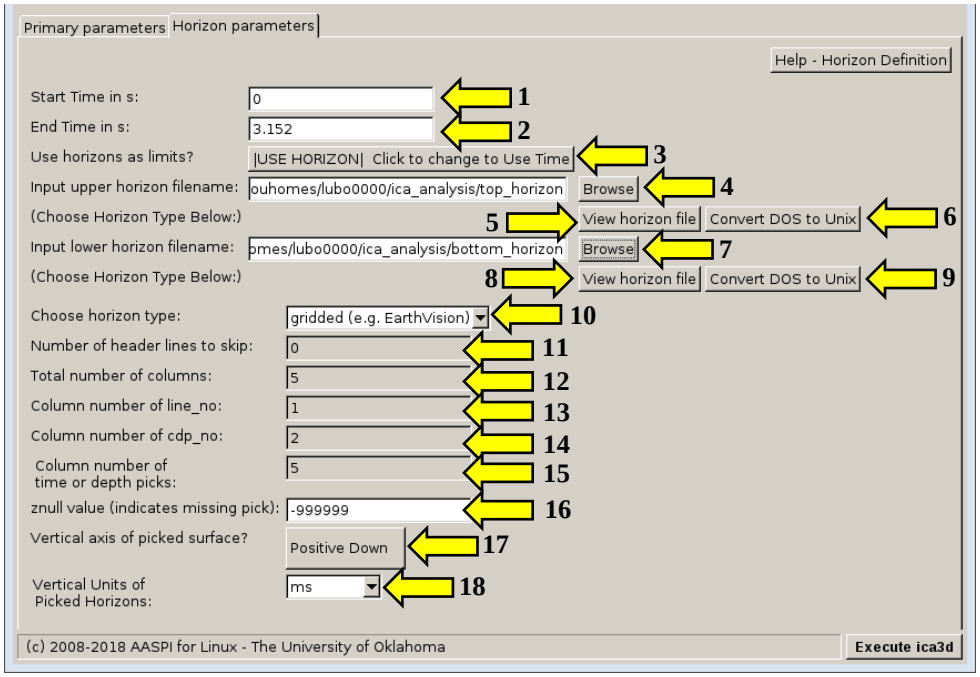
The program **ica3d** is launched from the *Volumetric Classification* in the main **aaspi_util** GUI

Computing ica3d module

Setting the parameters defining independent components is the first step of analysis. Use the browser to choose the input seismic volumetric attributes file (*Arrow 1*). The minimum number of inputs attributes to apply ica3d is three. The selection of the input attributes depends on the geological feature of interest. For analyzing facies variations, volumetric attributes such as coherency, GLCM attributes, dip magnitude, spectral magnitude components, coherent energy can be considered as input. In order to characterize geo-mechanical variation in shale plays volumes such as inversion volumes, lambda-rho, mu-rho, intercept or gradient AVO volumes are useful because they help to identify and characterize the rock physics. Put the maximum number of iterations (*Arrow 2*). Select the decimation rate of input data used to generate the training data (*Arrow 3*). The decimation rate will depend on the size and number of seismic attributes used by the user. If few and small seismic volumes are used, a lower decimation rate that the one shown in this documentation in recommended in order to guarantee convergence of the algorithm. Finally, select the percentage of variability retained used in PCA whitening (*Arrow 4*). The algorithm automatically outputs the number of independent components based on the percentage selected by the user.



Then a user need to define the operation window in the **Operation Window** tab shown below. A user can either use a fixed time window, or a window defined by two horizons.



Horizon definition

The horizon definition panel will look the same for almost all AASPI GUIs:

1. Start time (upper boundary) of the analysis window.
2. End time (lower boundary of the analysis window.
3. Toggle that allows one to do the analysis between the top and bottom time slices described in 1 and 2 above, or alternatively between two imported horizons. If *USE HORIZON* is selected, all horizon related options will be enabled. If the horizons extend beyond the window limits defined in 1 and 2, the analysis window will be clipped.
4. Browse button to select the name of the upper (shallower) horizon.
5. Button that displays the horizon contents (see Figure 1).
6. Button to convert horizons from Windows to Linux format. If the files are generated from Windows based software (e.g. Petrel), they will have the annoying carriage return (^M) at the end of each line (Shown in Figure 1). Use these two buttons to delete those carriage returns. Note: This function depends on your Linux environment. If you do not have the program **dos2unix** it may not work. In these situations, the files may have been automatically converted to Linux and thus be properly read in.
7. Browse button to select the name of the lower (deeper) horizon.
8. Button that displays the horizon contents (see Figure 1).
9. Button to convert horizons from Windows to Linux format. (see 6 above).
10. Toggle that selects the horizon format. Currently *gridded* (e.g. EarthVision in Petrel) and *interpolated* (ASCII free format, e.g. SeisX) formats are supported. The gridded horizon are nodes of B-splines used in mapping and have no direct correlation to the seismic data survey. For example, gridded horizons may be computed simply from well tops. The x and y locations are aligned along north and east axes. In contrast interpolated horizons have are defined by *line_no, cdp_no (crossline_no)* and *time* triplets for each trace location. Examples of both format are shown in Figure 1. If *interpolated* is selected, the user needs to manually define each column in the file.
11. Number of header lines to skip in the *interpolated* horizon files.
12. Total number of columns in the *interpolated* horizon files.
13. Enter the column number containing the *line_no (inline_no)* of the interpolated data triplet.
14. Enter the column number containing the *cdp_no (crossline_no)* of the interpolated data triplet.
15. Enter the column number containing the *time* or *depth* value of the interpolated data triplet.
16. *Znull* value (indicate missing picks) in the horizon files.
17. Toggle to choose between positive down and negative down for the horizon files (e.g. Petrel uses negative down).

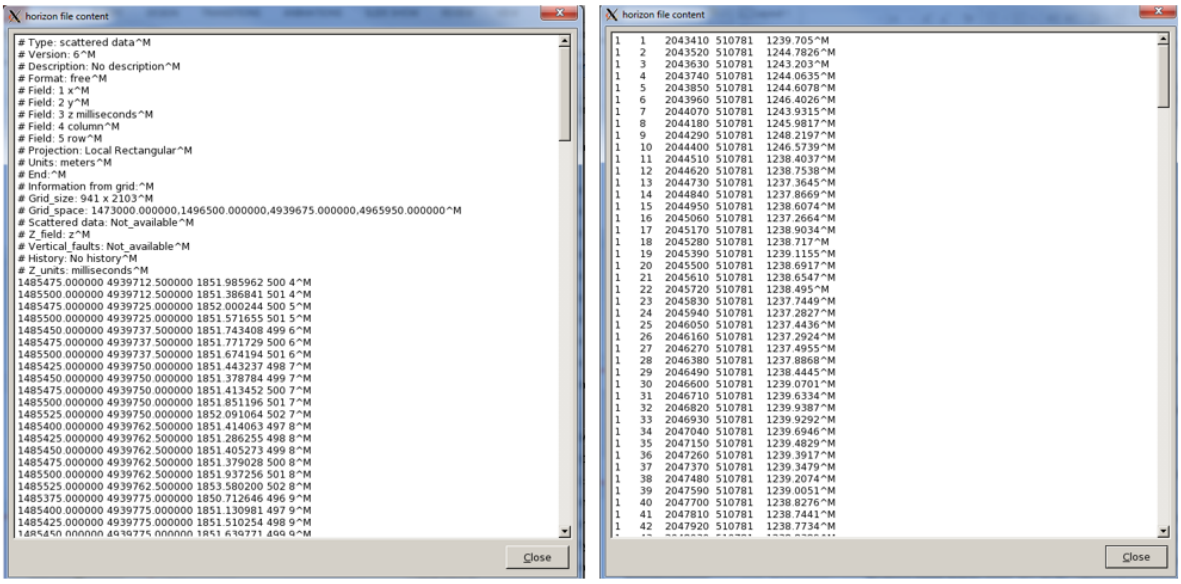
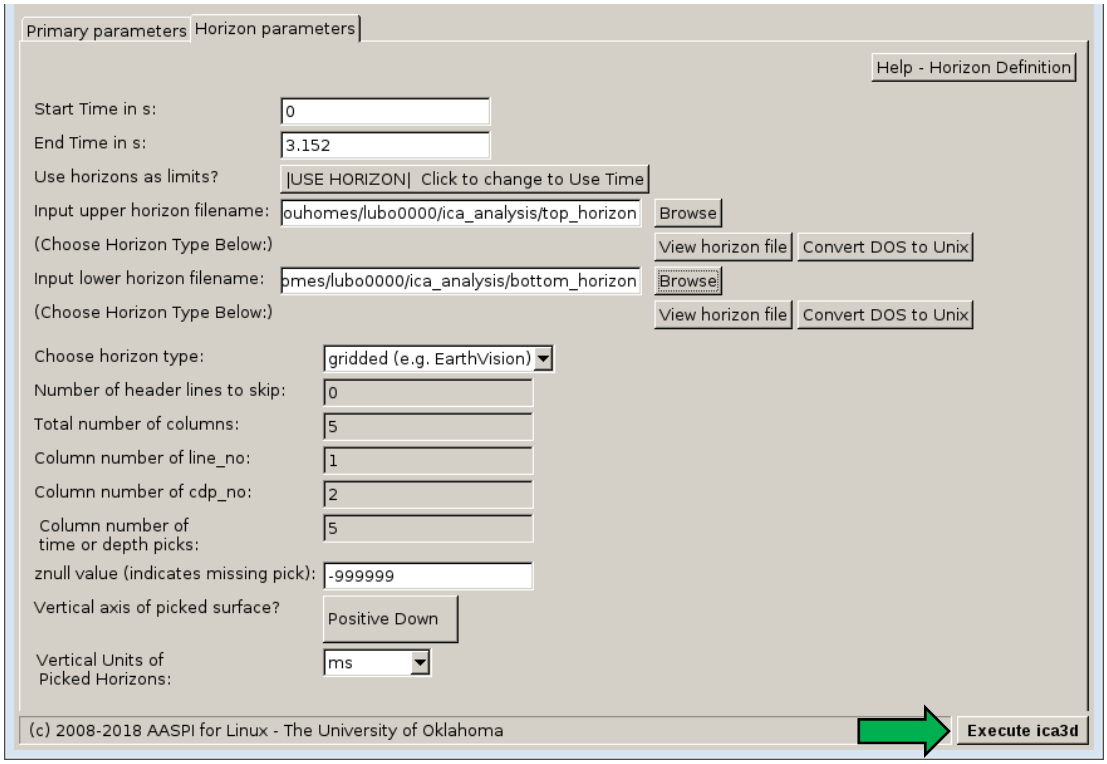


Figure 1. (left) A gridded horizon file (EarthVision format). (right) An interpolated horizon file with five columns (ASCII free format).

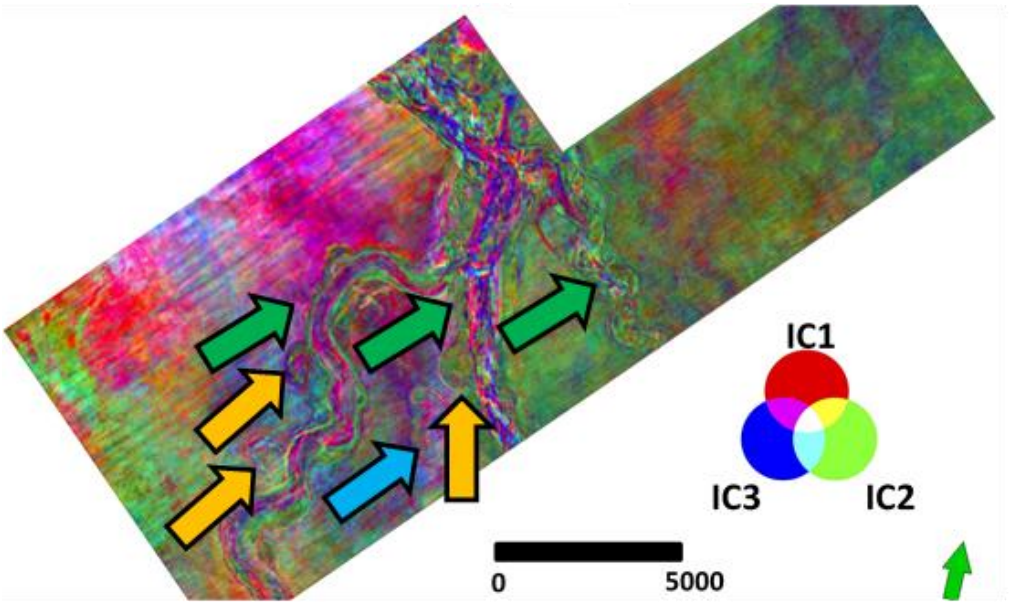
After defining the operation window parameters, press the ***Execute ica3d***.



The generated principle component files are named as:
ica_${unique project name}_${suffix}_j_temp.H

Visualization of the result

To view the resulted independent components, a user can either use **aaspi_plot** to display each component individually, use **aaspi_crossplot** to crossplot two components or use **rgb_cmy_plot** to plot three components using a RGB color scheme. One can also use visualization tools in commercial interpretation packages. It is important to highlight that the order of independent components is undefined, thus they should be sorted by visual inspection based on their geological insight. The figure below corresponds to three independent components plotted against a RGB color scheme using Petrel, along a phantom horizon in a channel complex present in the Moki A Formation, Taranaki Basin, New Zealand. We interpret the green arrows as meandering channels with moderate sinuosity and a tabular shape channel with highly variable internal architecture, orange arrows as oxbows, and the blue arrow as a small abandoned meandering channel.



References

Hyvärinen, A., 1999, Survey of independent component analysis: Neural Computing and Applications, **2**, 94–128.

Hyvärinen, A., and E. Oja, 2000, Independent Component Analysis: Algorithms and Applications: Neural Networks, **13**, nos. 4-5, 411-430.

Lubo-Robles D., 2018, Development of Independent Component Analysis for reservoir geomorphology and unsupervised seismic facies classification in the Taranaki Basin, New Zealand: Master's thesis, University of Oklahoma. http://mcee.ou.edu/aaspi/upload/AASPI_Theses/2018_AASPI_Theses/David_Lubo-Robles_MS_Thesis_Spring_2018.pdf

Stanford, 2018, PCA Whitening:
http://ufldl.stanford.edu/tutorial/unsupervised/PCAWhitening/. Accessed on March 26th, 2018.

Zanardo Honorio, B., A. Sanchetta, E. Pereira, and A. Vidal, 2014, Independent component spectral analysis: Interpretation, **2**, SA21-SA29, doi: 10.1190/INT-2013-0074.1