**Chroma: tutorials and descriptions**

*Correspondence: Julian Traphagan (*[*jtraph1@lsu.edu*](mailto:jtraph1@lsu.edu)*)*

Description: This document contains details on functions available in Chroma. Follow the **Vignette** section with the corresponding MATLAB file (vignette.m) to run basic example operations within the package. Function and input argument descriptions can be found here or by using the *help* command in MATLAB. Contact the correspondence email with questions or suggestions for new features.

**Overview of Chroma**

Hydrocarbon indices are frequently utilized in organic geochemistry as proxies for biological source, maturity, and degradation of organic compounds preserved in sediments. These molecular ratios provide information and constraints for paleoenvironmental and palaeoecological analyses and are thus powerful tools for interpretation and contextualization of the sedimentary record. Hydrocarbon indices are determined by identifying *n*-alkane chain lengths in chromatogram data and integrating them for compound abundance.

Chroma is a package of MATLAB functions for biomarker data processing and calculations in gas-chromatography analysis. The purpose of this package is to provide a group of efficient, accurate, and flexible tools for determining component-specific molecular abundances and hydrocarbon. The base calculations are performed in the title *chroma* function, which reads in a chromatogram of a sample analysis and automatically integrates targeted peaks which are referenced to a standard. Chroma was originally developed for the processing of chromatograms of *n*-alkane peaks produced by a flame ionization detector (FID) but can be adapted for identifying any time series of discrete peaks and will be updated periodically to include more post-processing calculations for other biomarker compounds.

List of available functions in Chroma:

1. *chroma*
2. *chromall*
3. *pkcomp*
4. *indall*
5. *prepfiles*
6. *prph*
7. *cpi*
8. *paq*
9. *acl*
10. *tarhc*
11. *oep*
12. *lh*
13. *lchsch*
14. *wi*
15. *c31c19*
16. *stack*
17. *plotchrom*
18. *compcpi*
19. *compacl*

**Running base programs**

The function *chroma* operates by reading in the time series information of a single sample and single standard of equal vector length. Target peaks (e.g., *n*-alkanes) are identified by locating the peaks in the standard and selecting the corresponding peaks in the sample. Non-target peaks (i.e., noise, contaminants, and other compounds in the sample) are automatically omitted from the sample peak identification if outside a specified time window of the peaks identified in the standard. The areas of the algorithm-selected peaks are determined by trapezoidal integration and stored in the output data table.

Input data must be read into functions in the Chroma package in a specific format. The function *prepfiles* automatically reads in a data file (e.g., file extensions .csv, .xslx, etc.) and restructures the data into a readable format. Data files must have the time information in the first column and the intensity information in the second column (any units are acceptable) to be properly formatted by *prepfiles*. If reading in multiple samples to be analyzed by *chromall*, data from each sample should be included in the data file following this column sequence (time, intensity; time, intensity; time, intensity…) with no space between columns.

The Chroma package includes operations for calculating several hydrocarbon indices either using individual functions for each or simultaneously using the *indall* function. The available indices to be calculated include the following: carbon preference index (CPI), average chain length (ACL), *P*­­aq, terrestrial/aquatic ratio (TAR), odd/even preference (OEP), low molecular weight/high molecular weight ratio (L/H), long chain/short chain ratio (LCH/SCH), weathering index (WI), and *n-*alkane ratio (C31/C19), among others. These calculations are all performed following the specifications of the original reporting literature.

**Peak Integration**

The sample peak integration algorithm is initialized by setting the criteria for peak identification. These include the peak height threshold minimum, start time of the analysis, and the maximum time window around the reference peak to identify corresponding sample peak. If not included as arguments, the function will use default values for each parameter. These parameters should be adjusted according to peak density and sample rate for optimal peak detection and noise filtering. If the standard does not include peaks present in the sample, these peaks will not be detected in the sample. To include these peaks in the calculations, additional peaks can be manually added if the estimated retention time and component number is known.

**Data visualization**

There are several functions included in the Chroma package for quick visualization and diagnosis of properties of a sample. Many of these are automatic outputs of the *chromall* function but can be individually performed as well. The *stack* function, for example, adds together and normalizes the chromatograms of multiple samples; the returned stacked chromatogram subdues the noise and emphasizes peaks which are present in all or most of the samples, thus visually identifying the dominant components present in the sample group. This calculation is included in the diagnostics output of the *chromall* algorithm, as well as a histogram of chain lengths, average integrated areas of each component, and the distribution of areas.

The *pkcomp* function is used for comparing the peak heights and areas of two samples or analyses. An example of a potential use for this function is determining the amount of material loss associated with urea adduction, a common procedure for noise reduction in gas chromatographic analysis. Comparing the peaks of a sample before and after urea adduction shows the percentage of material lost during the procedure.

The *plotchrom* function is a tool for easy data visualization, containing several options for plotting your data.

**Vignette**

Use this section with the corresponding vignette.m file to run base scripts in Chroma. Use *help* in MATLAB for any function to see documentation.

1. To run functions in Chroma, input sample and standard files must be in a readable format. The function *prepfiles* reads most delimited or spreadsheet file extensions (e.g., .txt, .csv, .xls, .xlsx) and converts the data and header information into the Chroma format. If the file has sample ID information, the row in the file may be specified by the 'headerrow' argument. The base *chroma* function requires the input of a sample and standard chromatogram converted to this structure. The input data file should have the first column as the time and the second column as the intensity (any units are accepted). See the './Chroma/data/exampleDF.xlsx' for an example of a proper input file format.
2. Once the standard and sample data files are in the Chroma structure, you can run *chroma* to determine the sample peak areas. Required input arguments include the sample (DF), standard (RM), and the components present in the standard (refcomp). If the number of peaks in the standard chromatogram does not match the length of refcomp, *chroma* will return an error. In this case, we used the Indiana University at Bloomington standard B4 containing the *n*-alkane components C16-30. MA is the output table containing the peak times, areas, and heights.
   1. Change 'view' to 'yes' to open a diagnostic figure. Here, you can assess whether the peak detections were successful. If not, additional parameters may need adjusting.

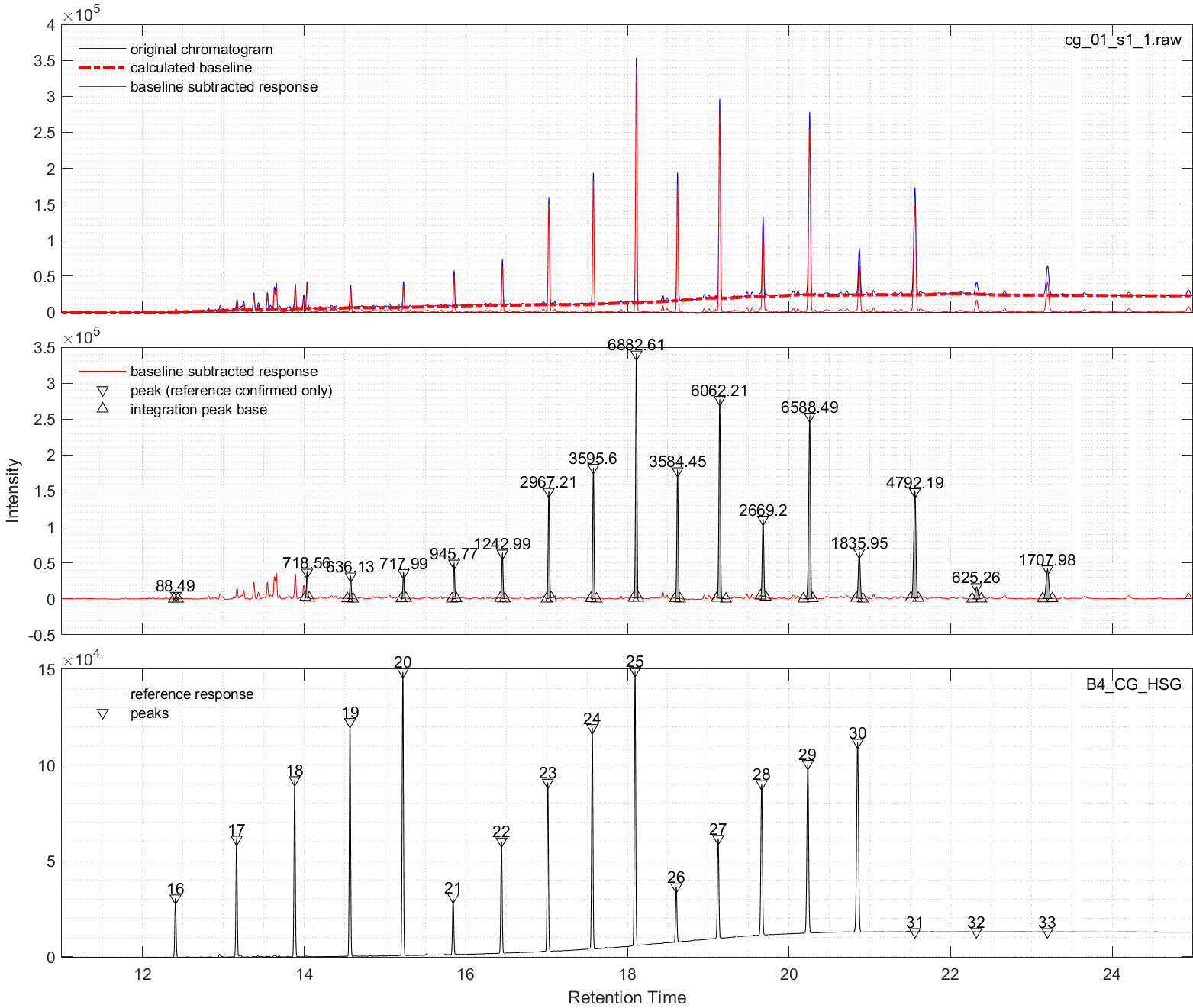


Fig. . Example of output with *chroma* or *chromall*. Syntax: chroma(DF, RM, refcomp, 'view','yes') or chromall(DF, RM, refcomp, 'view','yes')

1. The *chroma* function has several options for plotting and improved peak detections. These are all optional (with built-in default values), except for 'refcomp'
   1. 'refcomp' – a vector of numbers corresponding to components in the standard.
   2. 'pad' – add reference peaks not included in the standard. For example, B4 does not contain the C31-33 components, but if the retention times are known, they can be added to the analysis. The input data should be a two column matrix with time in the first column (in the same units as the data) and the component number in the second column.
   3. 'ds' – specify the reference detection window for peak correlations between the sample and standard. A smaller window provides a stricter criterion for peak correlations but may result in missed peaks if the standard and sample peak times do not align well. A larger window relaxes this criterion but may cause multiple sample peaks to be detected for a single sample, which will result in an error. In general, it is recommended to use the smallest value that is still able to capture all target peaks.
   4. 'view' – option for generating a figure ('yes' or 'no'). The figure is useful for diagnostic purposes. The top panel shows the spline interpolated baseline subtraction. The middle panel shows the integrated peak areas (gray). The bottom panel shows the standard and component numbers.
   5. 'smthreshold' – minimum peak threshold for sample chromatogram (units of intensity). Adjust this to filter out noisy peaks.
   6. 'rmthreshold' – minimum peak threshold for standard chromatogram (units of intensity). Make sure all standard peaks are above this threshold.
   7. 'xrange' – plotting range for x-axis (units of time) when 'view' is set to 'yes'.
   8. 'out' – output data format ('mat' or 'tab'). 'mat' will return a matrix and 'tab' will return a table.
   9. 'cutoff' – start time for the analysis. This tells *chroma* to ignore data before the specified time (useful for removing the solvent peak, for example).
2. If several samples need to be analyzed at once, the *chromall* function can read in multiple sample chromatograms and a single standard chromatogram to generate output hydrocarbon index data for all samples. The input sample data can be formatted from a single data file using *prepfiles*. Each sample should occupy two columns (for time and intensity) with no spaces between samples (see any of the Excel files in ./data for examples). For large datasets, formatting may take some time. The standard should still only contain one chromatogram. Input arguments for *chromall* are the same as *chroma*, with the addition of 'nfold' and 'prof':
   1. 'nfold' – directory for figure output when 'view' is 'yes'. If not specified, the figures will be put into a directory called 'chroma\_figs' in the current working directory. Output figures include diagnostic peak detection and pristane/phytane detection figures for each sample, a figure containing the calculated indices for the dataset, and a summary figure for distribution and uncertainty analysis.
   2. 'prof' – profile position of input sample chromatograms (i.e., depth, age, etc.). Values should be in the same order as the samples. If not specified, output figures will plot by the order number.

A graph of a graph

Description automatically generated

Fig. Example of index calculation figure output from *chromall*. Syntax: chromall(DF, RM, refcomp, 'view','yes')

A group of graphs showing different sizes and shapes

Description automatically generated

Fig. . Example of summary figure output from *chromall*. Syntax: chromall(DF, RM, refcomp, 'view','yes')

1. The function *pkcomp* compares the peak heights and areas of two samples. This is useful for identifying material lost during procedures like urea adduction. Input arguments require the sample and standard chromatograms for both samples, as well as the component numbers for each standard. Optional input argument used in *chroma* may be used in *pkcomp*. The plotting option is given by the argument 'plt'. The files example1.xlsx and example2.xlsx contain the chromatograms of HGZ sample 12 before and after urea adduction, respectively. The output table T shows the relative amount of material lost during the urea adduction procedure. Standard intensities were adjusted to share an appropriate 'rmthreshold' value. S1 and S2 correspond to the samples in the first and second arguments of the command.
2. The function *indall* can be used to generate a table of multiple index values from a single sample chromatogram. Use *chroma* to get the peak areas and component numbers, setting the 'out' argument to 'mat'. Then run *indall* from these areas. Individual indices can be calculated using the following functions: *acl*, *c31c19*, *cpi*, *lchsch*, *lh*, *oep*, *prph*, *paq*, *tarhc*, and *wi*.
3. The function *plotchrom* plots the chromatograms for easy visualization. Plotting options (after 'plt') include 'layer', 'stack', 'layout', and '3d'. The 'layer' option shows all available chromatograms in one panel window. The 'stack' option plots the sum of all chromatograms (normalized). The 'layout' option plots each chromatogram in individual panels. The '3d1' option plots all chromatograms in a rotatable (click and drag) 3D figure colored by sample number. The '3d2' option plots the same figure but colored according to intensity. The 'single' option plots a single chromatogram from the dataset, specified by the additional argument 'sn'.

A graph of different colored lines

Description automatically generated

Fig. . Example of output from *plotchrom* with syntax plotchrom(DF,'plt','layer','xrange',[11 25])

A graph of a number of objects

Description automatically generated

Fig. . Example of output from *plotchrom* with syntax plotchrom(DF,'plt','stack,'xrange',[11 25])

A graph of a graph

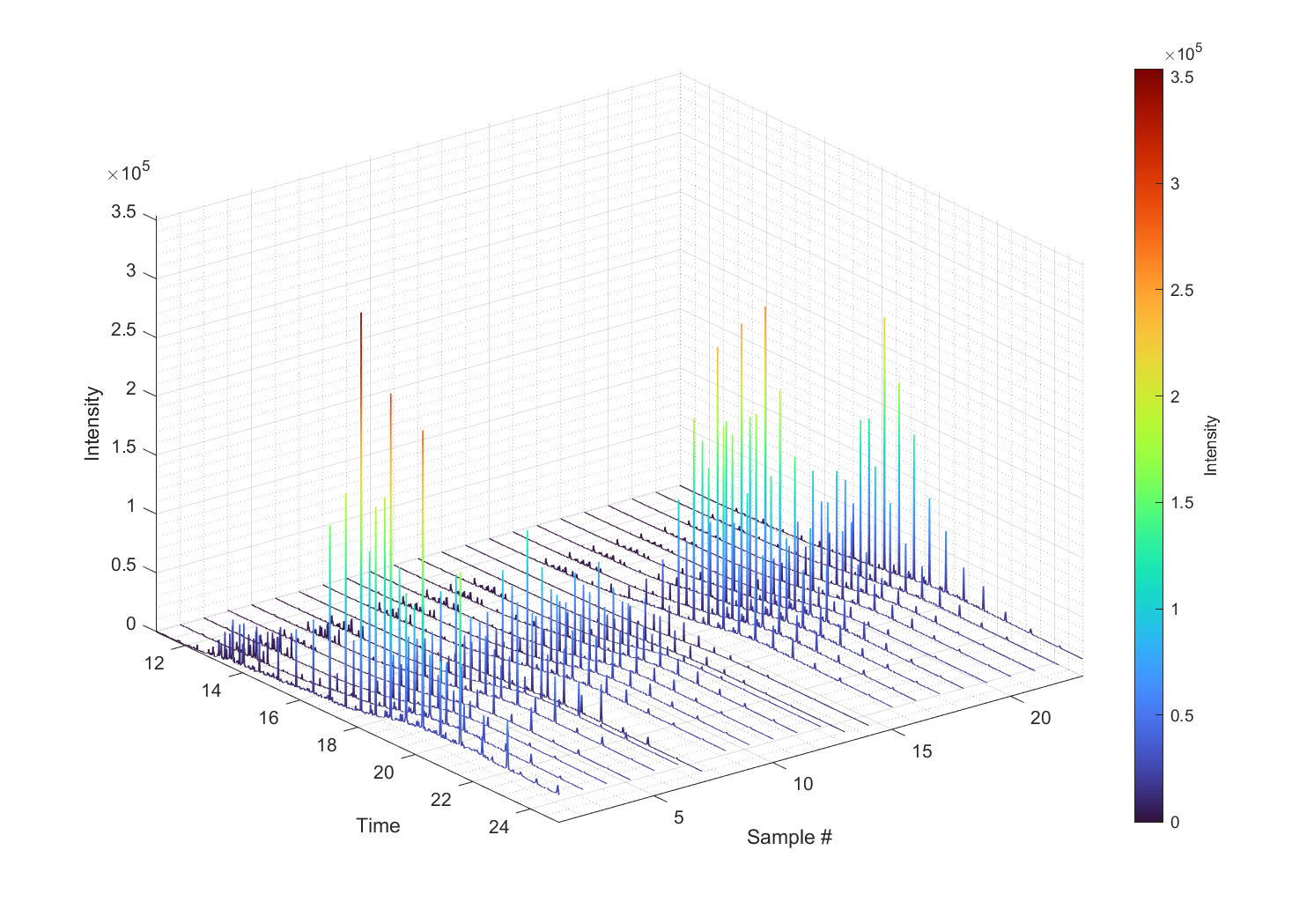
Description automatically generated

Fig. . Example of output from *plotchrom* with syntax plotchrom(DF,'plt','3d1','xrange',[11 25])

Fig. . Example of output from *plotchrom* with syntax plotchrom(DF,'plt','3d2','xrange',[11 25])

A graph with lines in the middle

Description automatically generated

Fig. . Example of output from *plotchrom* with syntax plotchrom(DF,'plt','single','sn',7,'xrange',[11 25]). This plots the 7th chromatogram in DF. If 'sn' is not specified, the first chromatogram in DF will be plotted.

1. Use *prph* to determine the abundance of pristane and phytane. When 'yes', the 'view' argument generates an output figure for assessing the automatic detection.

A graph of a graph of a person

Description automatically generated

Fig. 9. Example of output from *prph*. Syntax [PRPH,PR,PH] = prph(DF,RM,refcomp, 'view','yes','ds',30','rmthreshold', 2.5e4,'xrange',[12 16])

1. Use *compcpi* and *compacl* to compare the output of different methods for calculating CPI and ACL, respectively. Optional detection input arguments are the same as *chroma* and *chromall*. The 'plt' argument options are '1:1', 'profile', and 'none'.

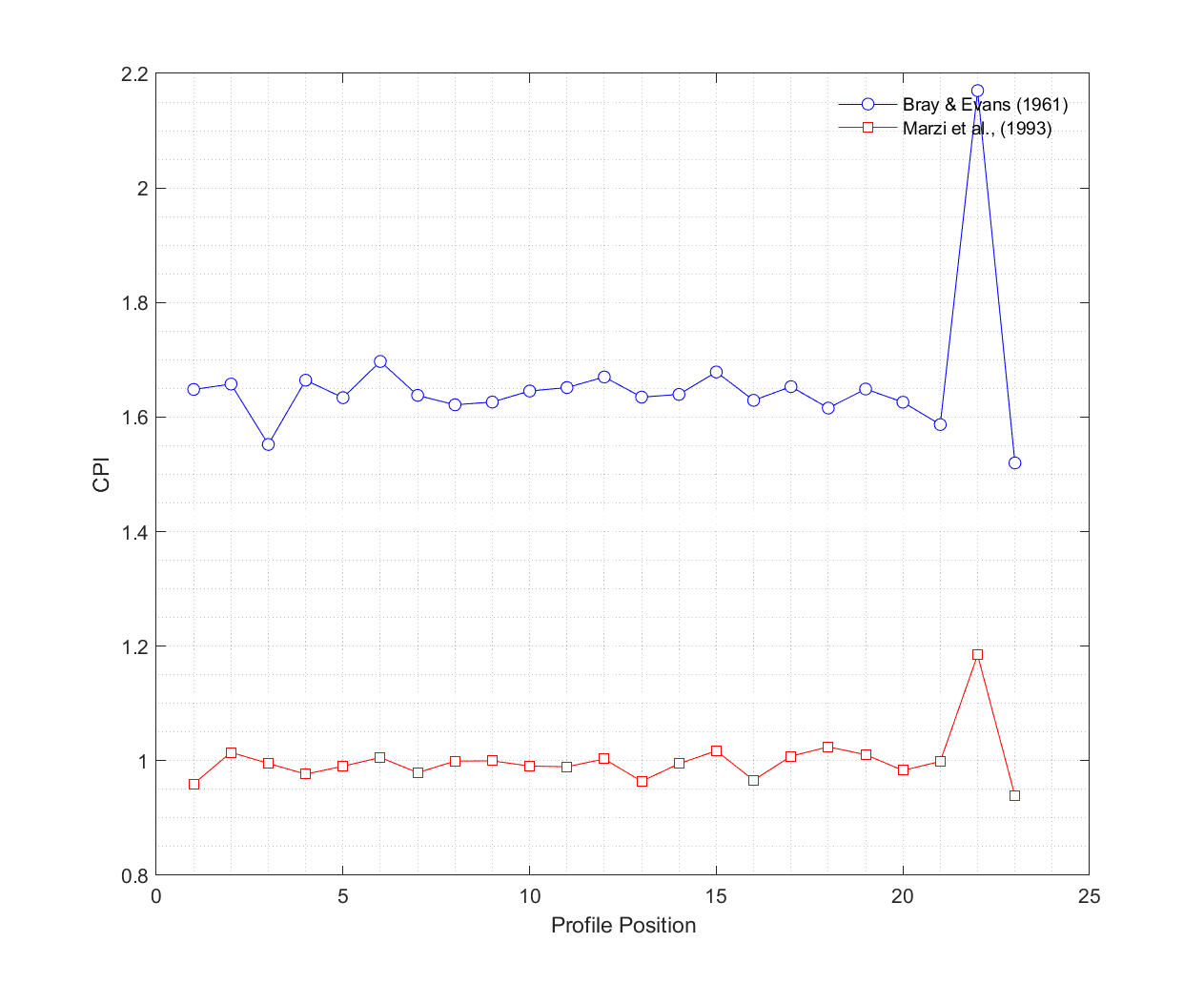


Fig. 10. Example of *compcpi* output figure with syntax [CPI2,CPIBE] = compcpi(DF,RM,refcomp,'profile');

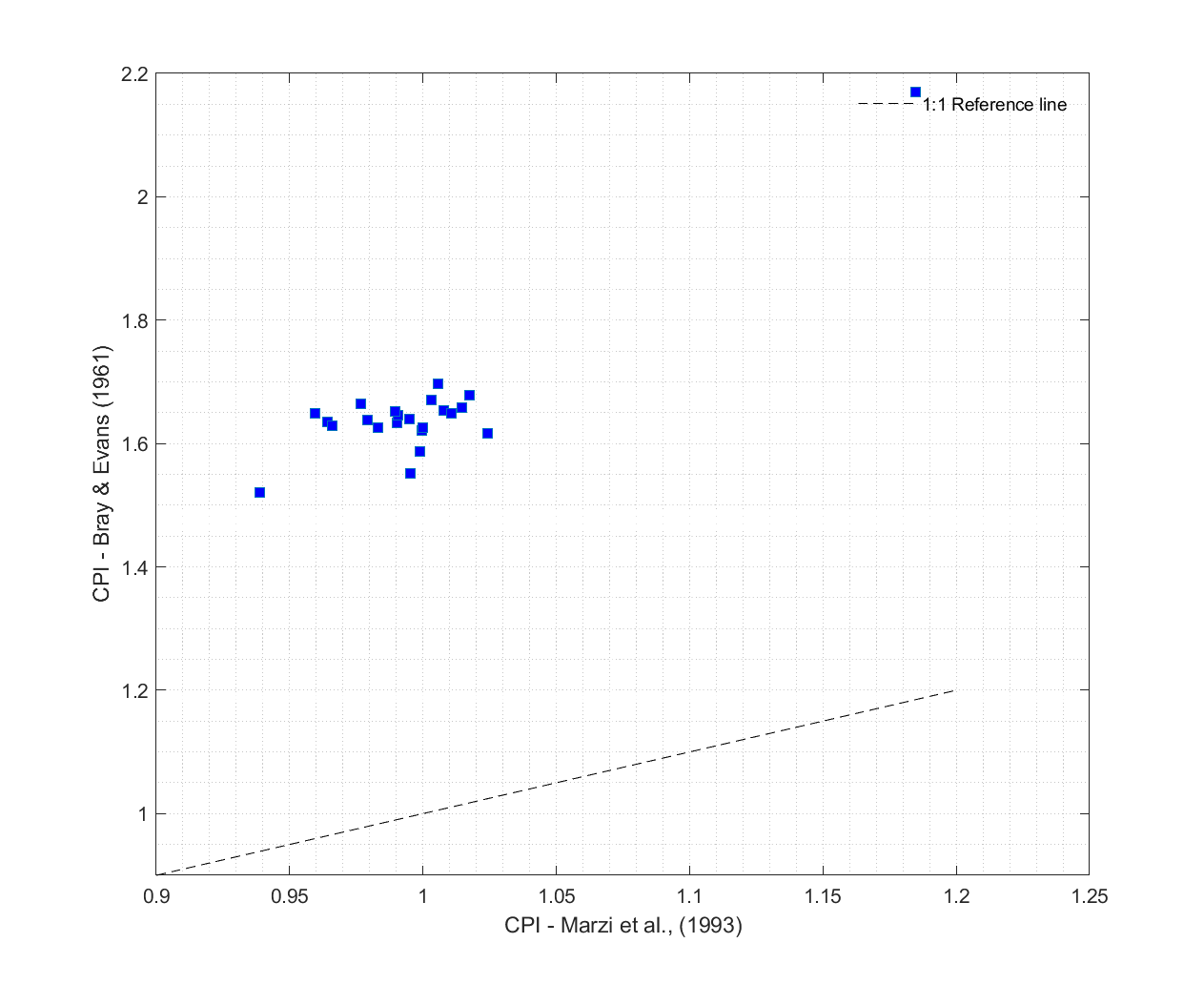


Fig. . Example of *compcpi* output figure with syntax [CPI2,CPIBE] = compcpi(DF,RM,refcomp,'plt','1:1');

A graph of different colored lines

Description automatically generated

Fig. . Example of *compacl* output figure with syntax [A1,A2,A3] = compacl(DF,RM,refcomp,'plt','profile');

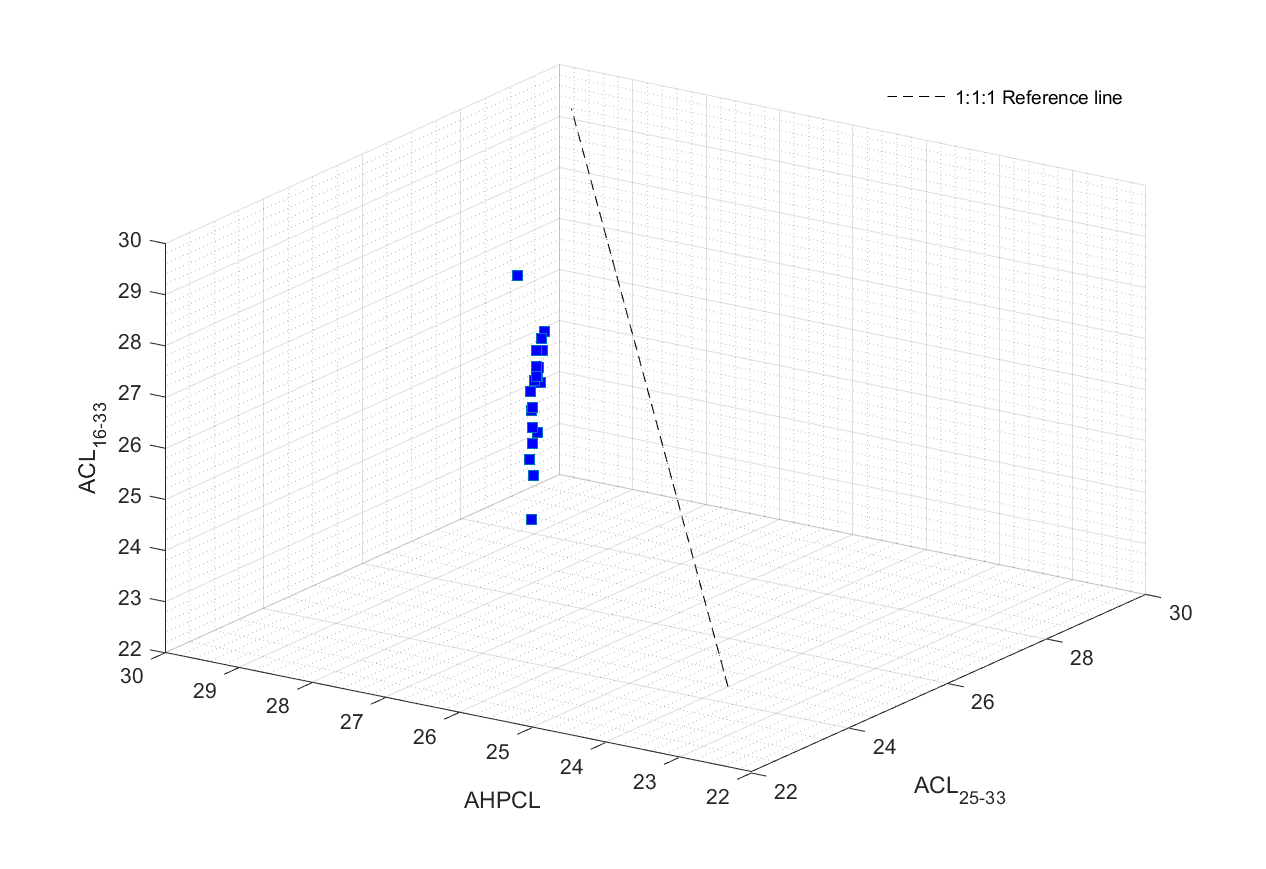


Fig. . Example of *compacl* output figure with syntax [A1,A2,A3] = compacl(DF,RM,refcomp,'plt','1:1');