Introduction to MSanalyzeNOM

Robert B. Young

6/10/2020

Introduction

This document demonstrates R functions in the **{MSanalyzeNOM}** package for analyzing the mass spectrometry (MS) data of natural organic matter (NOM) samples after formula assignment. It was specifically created to analyze MS data from NOM samples analyzed by Fourier transform ion cyclotron resonance (FTICR) MS at the National High Magnetic Field Laboratory (MagLab) in Tallahassee, Florida, USA. Usually, the analyses were conducted using electrospray ionization in negative ion mode (NegESI), which is the default ion type in the MSanalyzeNOM package. In addition, the MSanalyzeNOM package generally assumes that the detected ions are singly-charged, based on literature describing the ESI-FTICR MS analysis of dissolved NOM (e.g., Stenson et al. 2002).

Reading MS Data from the MagLab

The MSanalyzeNOM package uses the *get_sample_data()* function to read Excel files exported by PetroOrg© software, which was created at the MagLab for mass calibration, molecular formula assignment, and other purposes. The Excel file contains one or more sheets with summary data, a sheet with mass spectrometry data for detected ions that could not be assigned molecular formulas ("no hits"), and numerous sheets containing mass spectrometry data for the assigned molecular formulas, which have been grouped by heteroatom class (the number and identity of heteroatoms contained in the assigned molecular formulas). For example, $C_6H_{10}O_5$, $C_7H_{12}O_5$, and $C_8H_{14}O_5$ would be part of the "O5" heteroatom class, and $C_{11}H_9NO_6S$ would be part of the "N1 O6 S1" heteroatom class.

The *get_sample_data()* function returns a named list containing file and sample information, selected mass spectrometry parameters, mass spectrometry data related to the assigned molecular formulas, and mass spectrometry data related to detected ions that could not be assigned molecular formulas. Most of other functions in the MSanalyzeNOM package rely on column names produced by the *get_sample_data()* function.

```r
# sample_xcel contains the path and filename of a sample Excel file for a DOM
# sample if no path and filename are given, R's file.choose() function will
# permit you to choose a file
DOM_sample <- get_sample_data(sample_xcel)
```

Types of Sample Info In particular, the named list contains fields for the sample name, ionization method, and list of elements used for formula assignment. These fields are saved for use in later calculations. For example, the sample name can be used in plot titles, and the ionization method can be used to compute neutral masses for assigned formulas. If no ionization method is specified, the default value is "NegESI" because it is the method we most commonly use at the MagLab.

```r
DOM_sample$sample_name
```

```
## [1] "This_is_a_PetroOrg_Excel_file"
```

```
DOM_sample$ion_technique
```

```
## [1] "NegESI"
```

```
DOM_sample$element_list
```

```
## [1] "C" "H" "N" "O" "S"
```

Changing the Sample Name When the sample name is too detailed for a plot title, it can easily be changed.

```
DOM_sample$sample_name <- "Site 1 - Control (TPIA)"
DOM_sample$sample_name
```

```
## [1] "Site 1 - Control (TPIA)"
```

Types of MS Data for the Assigned Formulas The sheet containing the mass spectrometry data for the assigned formulas from PetroOrg© is illustrated below. Much of this information is used to produce new details for data summary and visualization.

```
head(DOM_sample$assigned_formulas)
```

```
## # A tibble: 6 x 11
##   hetero_class chem_formula    mz theor_mz ppm_error rel_abund     C     H     N
##   <chr>        <chr>        <dbl>    <dbl>     <dbl>     <dbl> <dbl> <dbl> <dbl>
## 1 O1           C11H16O1      163.     163.  -0.190       0.226    11    16     0
## 2 O1           C12H18O1      177.     177.  -0.175       0.206    12    18     0
## 3 O1           C13H20O1      191.     191.  -0.00523     0.158    13    20     0
## 4 O1           C14H22O1      205.     205.  -0.00487     0.557    14    22     0
## 5 O1           C15H24O1      219.     219.   0.0411      0.324    15    24     0
## 6 O1           C16H26O1      233.     233.   0.0386      0.137    16    26     0
## # ... with 2 more variables: O <dbl>, S <dbl>
```

Changing the Element List The table of elements was created from the list of elements that was used for formula assignment in PetroOrg©. The default list of elements includes C, H, N, O and S, but a different element list can be specified.

```
sample_data <- get_sample_data(sample_xcel, element_list = c("C", "H", "N", "O"))
sample_data[["element_list"]]
```

```
## [1] "C" "H" "N" "O"
```

Additional Details Additional details for the ***get_sample_data()*** function can be examined by typing *"?get_sample_data"* in the R console.

van Krevelen Diagrams (H/C vs. O/C)

Major biogeochemical classes of chemical compounds, such as lipids and carbohydrates, have characteristic H/C and O/C ratios (Kim et al. 2003). As a result, van Krevelen diagrams, which typically plot H/C vs. O/C ratios, are an important tool for characterizing NOM.

Calculating the Elemental Ratios The MSanalyzeNOM package uses the ***get_elemental_ratios()*** function to compute elemental ratios. The column name is derived from the numerator and denominator that are supplied. "C" is the default denominator used for van Krevelen diagrams, but different elemental ratios can be used for other purposes. For example, Zark et al. 2017 have developed a method for estimating numbers of carboxyl groups from O/H ratios.

```
DOM_sample$assigned_formulas <- DOM_sample$assigned_formulas %>%
  get_elemental_ratios(num = "H") %>%
  get_elemental_ratios(num = "O") %>%
  get_elemental_ratios(num = "O", denom = "H")
```

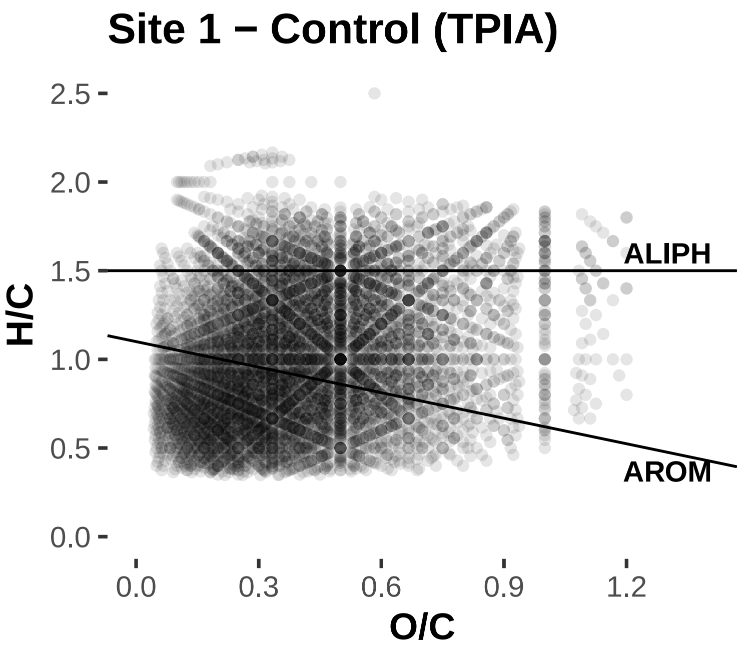
```
## # A tibble: 6 x 4
##   chem_formula  HtoC   OtoC   OtoH
##   <chr>        <dbl>  <dbl>  <dbl>
## 1 C11H16O1      1.45 0.0909 0.0625
## 2 C12H18O1      1.5  0.0833 0.0556
## 3 C13H20O1      1.54 0.0769 0.05
## 4 C14H22O1      1.57 0.0714 0.0455
## 5 C15H24O1      1.6  0.0667 0.0417
## 6 C16H26O1      1.62 0.0625 0.0385
```

Types of van Krevelen Diagrams A traditional van Krevelen diagram plots O/C vs. H/C. In general, high H/C ratios reflect a high degree of saturation, and low H/C ratios reflect the presence of rings and double bonds. Similarly, high O/C ratios reflect the relative presence of hydroxyl, carboxyl and other oxygen functional groups. Accordingly, the area above the horizontal line at $H/C = 1.5$ has been designated as the aliphatic region in accordance with D'Andrilli et al. 2015 and Lv et al. 2017, and the area below the diagonal line from $H/C = 1.1$ has been designated as the aromatic region, using the modified aromaticity index ≥ 0.5 in accordance with Koch & Dittmar 2006.

1. *Flat Diagram (No Information on Abundance)*

The following plot, created by the ***make_VK_flat()*** function, shows the distribution of the assigned formulas as a function of their elemental ratios, but contains no information about the intensities of the detected ions. The darker areas are where the highest number of formulas are plotted.

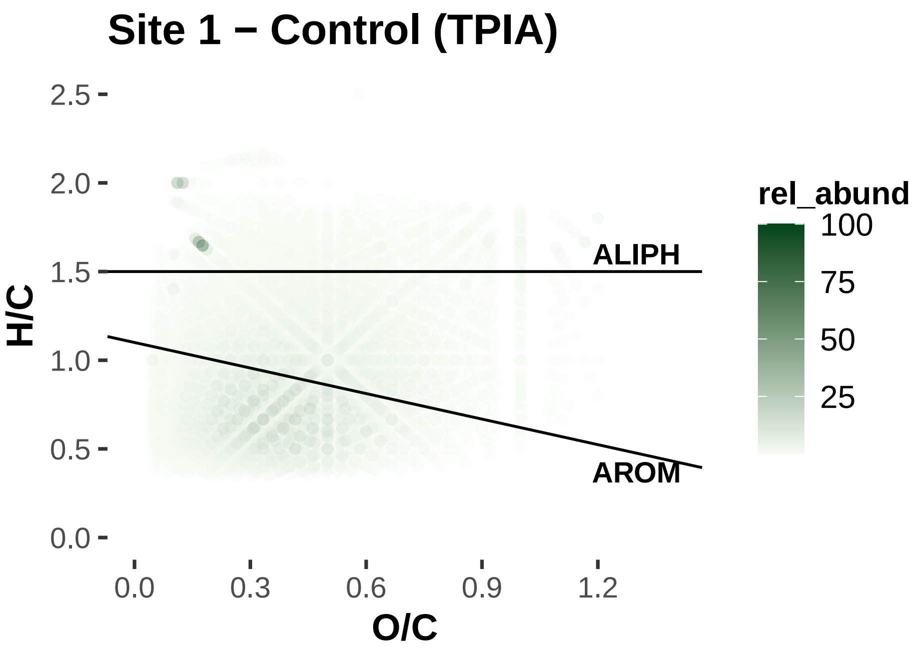
```
plot_VK_flat(DOM_sample$assigned_formulas, plot_title = DOM_sample$sample_name)
```



2. *Color Gradient (Poor Resolution of Abundance)*

The numbers of detected ions and assigned formulas can be influenced by ion suppression, making it difficult to compare van Krevelen diagrams from different samples or analyses. In fact, the number of assigned formulas is sensitive to ion suppression because the vast majority of detected ions occur at relatively low abundances. Similarly, coloring a van Krevelen diagram with the assigned formulas' relative abundances doesn't necessarily improve resolution.

```
plot_VK_gradient(DOM_sample$assigned_formulas, var = "rel_abund",
                 plot_title = DOM_sample$sample_name)
```



3. *Grouped by %Contribution to Abundance (Better Resolution of Abundance)*

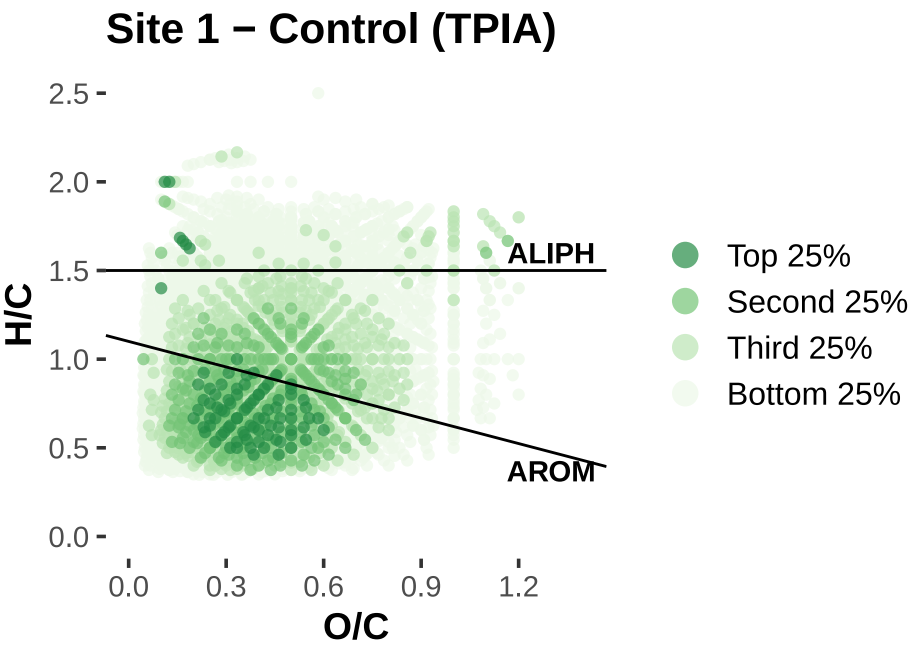
One alternative for increasing resolution is to group assigned formulas by their percent contribution to total assigned abundance, and then plot the grouped formulas in a van Krevelen diagram. For example, all of the formulas that account for the top 25% of total abundance can be grouped together, and so on, until four distinct groups have been formed: “top 25%”, “second 25%”, “third 25%”, and “bottom 25%”.

```
DOM_sample$assigned_formulas <- DOM_sample$assigned_formulas %>%
  get_perc_abund() %>%
  get_25perc_groups()
```

In the current example, typical of an NOM sample, the vast majority of assigned formulas comprise the bottom 25%, and a relatively small number of formulas comprise the top 25%. The end result is that the van Krevelen diagram better shows where the most abundant assigned formulas congregate.

```
## # A tibble: 4 x 2
##   `25perc_group` num_formulas
##   <ord>                 <int>
## 1 Top 25%                  89
## 2 Second 25%              275
## 3 Third 25%               818
## 4 Bottom 25%             4937
```

```
plot_VK_groups(DOM_sample$assigned_formulas, plot_title = DOM_sample$sample_name)
```


Site 1 − Control (TPIA)
H/C
O/C
ALIPH
AROM
● Top 25%
● Second 25%
● Third 25%
● Bottom 25%