Chromatography Model

James Klimavicz

September 1, 2024

Overview

The purpose of this package is to create realistic-looking chromatogram data, and then analyze the chromatograms to detect peaks, and calculate peak parameters for each peak. The package attempts to capture many qualities of authentic chromatograms. Key features include:

- Baselines noise, with the ability to reproduce some autoregressive behavior in the baseline signal, as expected for a time series.
- Asymmetric peak shape, with peaks being modeled by exponentially modified gaussian functions [1]. Peaks that elute later tend to be broader and shorter, but with similar areas.
- Solvent gradients and choices of solvent affect the baseline absorbance, and result in changes in the peak retention time of each compound. The change in retention time depends on the choice of solvents, percentage of each solvent, and the flow rate.
- Compound data for over 300 compounds, including UV-Vis data, have been pre-populated to produce random samples of various compounds. Samples can be defined with lists of compound names or CAS numbers
- Ability to introduce small random variations in peak elution times and peak heights, both constant across an injection and also within an injection to produce realist variations between chromatograms for otherwise identical samples and methods.

Contents

1 Systems

Currently, a system is implemented to:

- store a system name,
- allow for a system-specific deviation in retention time δt_{sys}, e.g. those due to differences in tubing lengths between column components,
- facilitate injections onto the column, and
- allow for placing a new column with a different serial number into the system.

The `System` also stores any `**kwargs` provided to the class. Nothing is currently implemented for this, but this may be useful to have as a reference for, e.g., an output file.

1.1 Columns

The column itself is modeled by the `Column` class, which keep track of injections, n_{inj}. Upon instantiation, the column is given attributes for column-specific differences for peak asymmetry δA_{def}, broadening δB_{def}, and retention time δt_{def}.

It also contains an attribute for when a column may begin to fail, n_{fail}. After this number of injections is reached, each injection carries a probability of triggering a failure, equal to $0.00015 \cdot (n_{\text{inj}} - n_{\text{fail}})$. After failing, each injection causes slight increases in asymmetry δA_{fail}, broadening δB_{fail}, and retention times δt_{fail}, using the following formulas:

$$\begin{aligned}
\rho &\in \text{Uniform}(0.95, 1.05) \\
d &= n_{\text{inj}} - n_{\text{fail}} + 1 \\
\xi &= 0.5\rho \cdot \ln\left(1 + 10^{-7} \cdot \left(\frac{x}{2} - 1\right)^3\right)
\end{aligned}$$

$$\delta A_{\text{fail}} = 20 \cdot \xi \cdot (A - 1) \tag{1}$$

$$\delta B_{\text{fail}} = \frac{\xi \cdot W}{20} \tag{2}$$

$$\delta t_{\text{fail}} = 1 + 0.01\xi \tag{3}$$

When it then comes to a specific injection, the column parameters $\delta A_{\text{col}} = \delta A_{\text{def}} + \delta A_{\text{fail}}$, $\delta B_{\text{col}} = \delta B_{\text{def}} + \delta B_{\text{fail}}$, and $\delta t_{\text{col}} = \delta t_{\text{def}} + \delta t_{\text{fail}}$ are surfaced to adjust the appearance of the chromatogram.

2 Methods

There are two types of methods: instrument methods, which determine how data is created (mimicking how data is collected in typical instruments), and processing methods, which determine how the chromatogram signal is processed.

2.1 Instrument Methods

Within the instrument methods are import parameters such as:

- solvents used
- solvent gradients and flow gradients used
- detection parameters, such as UV-Vis wavelengths.

2.1.1 Solvents

2.1.2 Solvent Gradients

2.2 Processing Methods

3 Compounds

The `Compound` class

3.1 Compound Library

3.2 UV Spectra

3.3 Solvents

3.4 Solvent Library

3.5 Calculation of Retention Times

The calculation of retention time depends on many factors, including:

- compound properties,
- solvent composition and properties,
- solvent flow rate, and
- column properties (not yet implemented).

Solvent attributes included in a `Solvent` class object include hydrogen bond acidity A, hydrogen bond basicity B, solvent polarity P, solvent dipolarity π, and solvent dielectric ε.

A given `Compound` has attributes for a default retention column volume v_0, an estimated partition coefficient $\log P$, total polar surface area σ, molecular weight MW, an estimated water solubility coefficient $\log S$, and number of hydrogen bond acceptors and donors, H_A and H_D. These properties were estimated using SwissADME [2].

Because the solvent profile changes over time, we note that the composite solvent attributes are functions of time. We will assume that these properties are linearly additive with respect to their component proportions. A method also specifies a flow $\eta(t)$, which may or may not remain constant over time.

Let $f(t)$ be the composite compound solvent interaction at time t. A later method will implement a more scientific model, but for now, the chosen parameters were chosen rationally (albeit not scientifically) to produce somewhat realistic changes in retention times based on solvent, including potential changes in elution order for some combinations of compounds. We first define the coefficients:

$$\alpha = \frac{600}{MW \cdot \left(\sqrt{H_D} - 1\right)}$$
$$\beta = \frac{600}{MW \cdot \left(\sqrt{H_A} - 1\right)}$$
$$\gamma = 2 \cdot (3 - \log P) + \frac{\sigma}{MW}$$
$$\delta = \frac{1 - \log S}{5}$$

We then define $f(t)$ to be (again, based on some rational decision but mainly based on heuristic results):

$$f(t) = \frac{1}{120}\left(\alpha \cdot A(t) + \beta \cdot B(t) + \gamma \cdot P(t) + \delta \cdot \varepsilon(t) - 0.05 * (3 * \alpha + \beta + \gamma + \delta)\right) \tag{4}$$

Because elution time is dependent on the cumulative effects of solvent flow, we must account for both the solvent flow and $f(t)$. First, we have the eluted volume as a function of time, $V(t)$:

$$V(t) = \int_0^t \eta(\tau) d\tau, \tag{5}$$

so

$$dV = \eta(t) dt \tag{6}$$

The cumulative effect of solvent over time is is then given by

$$\int_0^{V(t_R)} f(t) dV = \int_0^{v_R} f(t) \cdot \eta(t) dt \tag{7}$$

The actual retention volume v_R can then be found by solving the equation

$$v_R = v_0 - \int_0^{v_R} f(t) \cdot \eta(t) dt \tag{8}$$

In some cases for highly polar species, the heuristically calculated v_R may be less than one, which is non-nonsensical since it implies the compound will elute before a single column volume has flowed through the column. We therefore set v' to be the maximum of the calculated

We then convert this to an elution time t' using the column volume v_{col} and the flow rate:

$$t_R = t(v_R)/v_{col}, \tag{9}$$

where $t(v)$ is the time at which v cumulative volume has been eluted.

4 Injection

4.1 Chromatogram Creation Steps

When an Injection is created, the following steps happen in order:

1. The `Sequence` object is updated with information about about the injection.
2. The `System` is updated with an adding an injection and incrementing the injection count on the column.

3. The `InjectionMethod` is loaded to:
    (a) Determine the wavelengths and channel names to use for creating chromatograms
    (b) Produce a solvent gradient profile for the chromatogram
    (c) Determine the chromatogram length and sample rate.
4. A baseline is created for each chromatogram based on the wavelength and solvent (section 5.1) profile.
5. The compound retention times and peak shapes are calculated based on:
    (a) Default peak widths and asymmetry specified in the user parameters
    (b) System-specific differences in retention time δt_{sys} (see section 1)
    (c) Column-specific differences in retention time δA_{col}, δB_{col}, δt_{col} (section 1.1)
    (d) Solvent profile, flow rate, and column volume (section 3.5)
6. Peak heights are adjusted based on compound UV-Vis absorbances at the wavelengths specified in the instrument method and specified amounts in the sample.
7. Adjusted peaks are then added to the baselines.

4.2 Peak Detection and Quantification Steps

When a chromatogram is is chosen for peak detection, the following steps occur:

1. An adaptive Savitzky-Golay smoothing step is first performed to generate a smoothed signal and smooth second derivative, as described in section 6.1
2. A baseline is determined as described in section 6.2, and subtracted from the smoothed signal
3. The peak detection algorithm described in section 6.3 is used to find peaks
4. Peak parameters for each peak are calculated (section 6.4)

5 Chromatogram

5.1 Baseline

The chromatogram baseline is modeled as

$$y(t) = b(t) + \mathcal{X}_t, \tag{10}$$

where $b(t)$ is a background signal dependent on the solvent profile as outlined in section 2, and $\mathcal{X}_t$ is a one-parameter autoregressive process $\mathcal{X}_t = \varphi \mathcal{X}_{t-1} + \varepsilon_t$ with user-provided parameter φ and white noise process $\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$.

5.2 Full Chromatogram

The full chromatogram for each detection wavelength is generated by sequentially adding peaks to a baseline. For each compound in the injection, a compound-specific absorbance is calculated from the `Compound` class, and an exponentially-modified gaussian peak is created based on actual compound retention time (see section 3.5), and compound- and time-specific asymmetry and peak broadening.

6 Analysis

6.1 Adaptive Signal Smoothing

The signal smoothing algorithm was implemented as inspired by Chromeleon's "Auto" smoothing algorithm, as described in [3]. Traditionally, smoothing has been performed using a Savitzky-Golay (SG) filter with a constant-width window across the whole chromatogram. However, choosing too small a window results in insufficient smoothing, especially in areas without any peaks, while large might provide adequate smoothing in flatter areas, but deform peaks by making them shorter and wider. Additionally, larger windows may introduce artifacts like baseline signal ringing into the chromatogram.

The goal of the adaptive smoothing is to use a maximally sized smoothing window in regions without peaks, and reduce the size of the smoothing window around peaks. While the technical note does not describe much in the way of implementation, algorithm 1 provides the outline of the implementation for this module.

The algorithm takes the signal $\mathbf{y}$ and a max window size $n = 2k + 1, k \in \mathbb{Z}^+$ as input. The window sizes range from 3 to n over the odd integers, and so there are k different window sizes. In the following algorithm, $\bar{\mathbf{y}}^k = \texttt{SavGol}(\mathbf{y}, n)$ is chromatogram signal $\mathbf{y}$ smoothed by the polynomial order 2 SG filter with window size $n = 2k+1$, while $\overline{y}_i^k$ is the ith value from the chromatogram smoothed with window size n.

The first width chosen s_{curr} is set as $k-1$. At each point y_i, with s_{curr}, we calculate several variances and means, and use this weighting to calculate the optimal window width. We also limit jumps in window sizes between consecutive points to ensure smoothness and prevent large jumps in value.

Algorithm 1 Adaptive Signal Smoothing

```
Require: $\mathbf{y}, s_{\max}$
  for $j \leftarrow 1, k$ do
    calculate $\overline{\mathbf{y}}^j = \texttt{SavGol}(\mathbf{y}, 2i+1)$
  end for
  function SmoothnessPenalty($\overline{\mathbf{y}}^s, i, s, n$)
    $k \leftarrow$ signal noise
    $a_1 = \text{var}(\overline{\mathbf{y}}^s[i-k : i+k+1])$
    $a_2 = 1 + (k - \text{mean}(\overline{\mathbf{y}}^s[i : i+k+1]))^2$
    $a_3 = 1 + (k - \text{mean}(\overline{\mathbf{y}}^s[i-k : i+1]))^2$
    $a_4 = 1 + \text{mean}((\mathbf{y} - \overline{\mathbf{y}}^s)[i-3 : i+4])^2$
    return $a_1 \cdot a_2 \cdot a_3 \cdot a_4/\sqrt{2n+1}$
  end function
  $s_{\text{curr}} \leftarrow k-1$
  $\tilde{\mathbf{y}} \leftarrow \mathbf{0}$
  for $i \leftarrow 1, \text{len}(\mathbf{y})$ do
    $s_{\min} \leftarrow \max(1, s_{\text{curr}} - 1)$
    $s_{\text{mak}} \leftarrow \min(k, s_{\text{curr}} + 1)$
    for $l \leftarrow s_{\min}, s_{\text{mak}}$ do
      $p_l = $ SmoothnessPenalty($\overline{\mathbf{y}}^l, i, l, n$)
    end for
    $\tilde{\mathbf{y}}_i \leftarrow \overline{\mathbf{y}}_i^{\text{argmin}(p_l)}$
  end for
```

6.2 Baseline Detection and Smoothing

The baseline is detected using an asymmetric least squares algorithm. The algorithm currently implemented is the *Peaked Signal's Asymmetric Least Squares Algorithm* (psalsa) method [4].

The algorithm takes three parameters: $p \in (0,1)$, which is a weight parameter; $s > 0$, which is a multiplicative factor that penalizes changes in the second derivative of the baseline, and $k \geq 1$, which controls the exponential decay of weights in peak regions.

The weights w_i are calculated as:

$$w_i = \begin{cases} p \cdot e^{\frac{-r_i}{k}} & \text{if } r_i > 0 \\ 1 - p & \text{otherwise} \end{cases} \tag{11}$$

for residuals $r_i = y_i - z_i$, where z_i is the baseline fit at point i. The cost function for psalsa is then

$$C(\mathbf{r}, \mathbf{w}, \mathbf{z}) = \sum_i w_i r_i^2 + s \sum_i (\Delta^2 z_i)^2, \tag{12}$$

where Δ^2 second difference operator. Minimization of this cost function gives

$$\mathbf{z} = (\mathbf{W} + s \cdot \mathbf{D}^T\mathbf{D})^{-1}\mathbf{W}\mathbf{y}, \tag{13}$$

where $\mathbf{D}$ is the tridiagonal second difference matrix, and $\mathbf{W} = \text{diag}(\mathbf{w})$. The implemented method perturbs $\mathbf{D}$ such the $\mathbf{D}_{1,1} = \mathbf{D}_{n,n} = 1$.

For tractability, the chromatogram signal and times arrays are down-sampled to prevent massive matrices. The down-sampled points are then interpolated with a cubic spline. The algorithm is outlined in Algorithm 2.

Algorithm 2 Asymmetric Least Squares: psalsa

```
Require: $\mathbf{y}, k, s, p, \text{tol}$
  $\mathbf{z} \leftarrow \text{mean}(\mathbf{y}) \cdot \mathbf{1}$        ▷ initial baseline guess
  $\mathbf{r} \leftarrow \mathbf{y} - \mathbf{z}$                 ▷ initial residuals
  $\mathbf{w} \leftarrow \mathbf{1}$
  prev_loss $\leftarrow \infty$
  while not converged do
    $\mathbf{W} \leftarrow \text{diag}(\mathbf{w})$
    $\mathbf{z} \leftarrow (\mathbf{W} + s \cdot \mathbf{D}^T\mathbf{D})^{-1}\mathbf{W}\mathbf{y}$
    $\mathbf{r} \leftarrow \mathbf{y} - \mathbf{z}$
    loss $\leftarrow C(\mathbf{r}, \mathbf{w}, \mathbf{z})$
    if |loss − prev_loss| < tol then
      converged $\leftarrow$ True
    end if
    prev_loss $\leftarrow$ loss
    update $\mathbf{w}$
  end while
```

6.3 Detection Algorithm

6.3.1 Initial Region Detection

6.3.2 Region Expansion

6.3.3 Region Contraction

6.4 Peak Parameters

6.4.1 Times and Signals

Peak start and end are determined by the peak detection algorithm. They are selected so the peaks may start and end at the same time, but no peaks may overlap.

`start_time` and `end_time`: The start and end of the peak as determined by the detection algorithm in 6.3. In the following peak parameters, start time is typically denoted t_i (for initial time) and end time as t_f (for final time).

`retention_time`: The time at which a peak exhibits maximum height above baseline. To account for noise in the signal, the highest point in the raw signal is found, and a quadratic curve is fit to the series of time/signal points starts starting three points before the maximal signal and ending three after. The time coordinate of the vertex of this quadratic is set to be the retention time, t_R. The signal coordinate of this vertex is set to be the peak height.

`start_baseline`, `end_baseline`, and `retention_baseline`:

`start_signal` and `end_signal`:

6.4.2 Type

6.4.3 Quantification

`area`: The area A of the peak under the raw signal and above the baseline:

$$A = \int_{t_i}^{t_f} f(t) \approx \sum_{k=u}^{k} f(t_k)\Delta t,$$

where $f(t)$ is the baseline-corrected signal. Units are in signal $\cdot$ time; for UV-Vis, the units are mAU $\cdot$ min.

`relative_area`: The percentage of area of the peak relative to the summed area of all peaks; that is, if there are n peaks, peak i has relative area

$$A_{\text{rel}} = 100\% \cdot \frac{A_i}{\sum_{k=1}^{n} A_k}$$

with units of percent.

`capillary_electrophoresis_area`: Equal to A/t_R, the capillary electrophoresis area is the area adjusted for migration (retention) time in capillary electrophoresis to compensate for changes in migration time between runs

`height`: The maximal height h of the above the baseline at the retention time t_R. This height is calculated using a quadratic approximation to the apex of the peak; see the `retention_time` entry in section 6.4.1. Height units are the units of the signal.

`relative_height` The percentage of the peak's height above baseline relative to the summed heights above baseline for all peaks:

$$h_{\text{rel}} = 100\% \cdot \frac{h_i}{\sum_{k=1}^{n} h_k}$$

with units of percent.

6.4.4 Widths

When possible, peak widths are calculated based on actual signal data, determining at what time a signal crosses a height threshold to the left and right of the retention time. For peaks that are not baseline resolved, it may be necessary to use extended peak calculations to estimate peak widths.

`width_50_<left,right,full>`: The left, right, and full peak width of the peak at 50% height.

`width_10_<left,right,full>`: The left, right, and full peak width of the peak at 10% height.

`width_50_<left,right,full>`: The left, right, and full peak width of the peak at 5% height.

`width_4_sigma_<left,right,full>`: The left, right, and full peak width of the peak at $4\sigma \approx 13.4\%$ height.

`width_5_sigma_<left,right,full>`: The left, right, and full peak width of the peak at $5\sigma \approx 4.4\%$ height.

`width_baseline_<left,right,full>`: The left and right points of inflection of the fitted exponentially-modified gaussian are first calculated. The slopes at these two points are then calculated, and the lines with this slope going through these two points are extended to the baseline. The left, right, and full baseline widths are determined based on where these slope lines intersect the baseline, as well as a vertical line at t_R.

`extended_width_calculation`: an array of the string value names for which extended peak width calculations were needed to calculate, as opposed to using raw signal values.

6.4.5 Statistical Moments

Statistical moments quantify aspects of the peak shape. For the calculation of peak shapes, the function $f(t)$ is the baseline-corrected signal, which is equal to the raw signal after subtraction of the baseline. For the calculation of moments, the absolute value of this function is use to avoid negative moments, which can lead to runtime errors when calculating higher moments. This will result in differences between **`moment 0`** and the **`area`** for smaller peaks when noise levels cause the signal to drop below the baseline.

`moment_0`: Equal to peak area. For peak start time t_i and peak end t_f and baseline-corrected signal $f(t)$,

$$\mu_0 = \int_{t_i}^{t_f} |f(t)|dt \approx \sum_{k=i}^{f} |f(t_k)|\Delta t.$$

Units are equal to the signal value multiplies by time in minutes, e.g. mAU · min.

`moment_1`: Average retention time. For peak start time t_i and peak end t_f and baseline-corrected signal $f(t)$,

$$\mu_2 = \frac{1}{\mu_0} \int_{t_i}^{t_f} t \cdot |f(t)|dt \approx \sum_{k=i}^{f} t_k \cdot |f(t_k)|\Delta t$$

Units are in minutes.

`moment_2`: Retention time variance. For peak start time t_i and peak end t_f and baseline-corrected signal $f(t)$,

$$\begin{aligned}\mu_2 &= \frac{1}{\mu_0} \int_{t_i}^{t_f} (t-\mu_1)^2 \cdot |f(t)|dt \\ &\approx \sum_{k=i}^{f} (t_k-\mu_1)^2 \cdot |f(t_k)|\Delta t\end{aligned}$$

Units are in min^2.

`standard_deviation`: Retention time standard deviation. Standard deviation $\sigma = \sqrt{\mu_2}$. Units are in min.

`moment_3`: Third central moment of retention time. For peak start time t_i and peak end t_f and baseline-corrected signal $f(t)$,

$$\begin{aligned}\mu_3 &= \frac{1}{\mu_0} \int_{t_i}^{t_f} (t-\mu_1)^3 \cdot |f(t)|dt \\ &\approx \frac{1}{\mu_0} \sum_{k=i}^{f} (t_k-\mu_1)^3 \cdot |f(t_k)|\Delta t\end{aligned}$$

Units are in min^3.

`moment_3_normalized`: Third normalized moment of retention time, or statistical skewness (not to be confused with USP skewness). For peak start time t_i and peak end t_f and baseline-corrected signal $f(t)$,

$$\begin{aligned}\tilde{\mu}_3 &= \frac{1}{\mu_0 \cdot \mu_2^{3/2}} \int_{t_i}^{t_f} (t-\mu_1)^3 \cdot |f(t)|dt \\ &\approx \frac{1}{\mu_0 \cdot \mu_2^{3/2}} \sum_{k=i}^{f} (t_k-\mu_1)^3 \cdot |f(t_k)|\Delta t\end{aligned}$$

The normalized moment is unitless.

`moment_4`: Fourth central moment of retention time. For peak start time t_i and peak end t_f and baseline-corrected signal $f(t)$,

$$\begin{aligned}\mu_4 &= \frac{1}{\mu_0} \int_{t_i}^{t_f} (t-\mu_1)^4 \cdot |f(t)|dt \\ &\approx \frac{1}{\mu_0} \sum_{k=i}^{f} (t_k-\mu_1)^4 \cdot |f(t_k)|\Delta t\end{aligned}$$

Units are in min^4.

`moment_4_normalized`: Fourth central moment of retention time, or kurtosis. For peak start time t_i and peak end t_f and baseline-corrected signal $f(t)$,

$$\begin{aligned}\tilde{\mu}_4 &= \frac{1}{\mu_0 \cdot \mu_2^{2}} \int_{t_i}^{t_f} (t-\mu_1)^4 \cdot |f(t)|dt \\ &\approx \frac{1}{\mu_0 \cdot \mu_2^{2}} \sum_{k=i}^{f} (t_k-\mu_1)^4 \cdot |f(t_k)|\Delta t\end{aligned}$$

The normalized moment is unitless.

6.4.6 Asymmetry

Asymmetry provides a measure for how much a peak is tailing or fronting. This metric may be used to monitor column quality. For the USP/EP and AIA definitions, as well as skewness, ideal peaks have a value near 1, while peaks less than 1 signify a fronting peak, and values greater than 1 signify a tailing peak. Unless otherwise stated in a test or assay, the USP standard requires `asymmetry_USP` to fall between 0.8 and 1.8 if the peak is to be used for quantification.

For statistical asymmetry, ideal peaks are scattered around 0; this value is negative for fronting peaks and positive for tailing peaks.

`asymmetry_USP`: US Pharmacopoeia calculation for asymmetry; this is identical to the European Pharmacopoeia (EP) and Japanese Pharmacopoeia (JP) definition:

$$A = \frac{RW_{5\%} + LW_{5\%}}{2 \cdot LW_{5\%}} = \frac{W_{5\%}}{2 \cdot LW_{5\%}}$$

for 5% left, right and full widths, $LW_{5\%}$, $RW_{5\%}$, and $W_{5\%}$. This value is undefined if these widths cannot be calculated. This value is also sometimes called the **tailing factor** or **asymmetry factor**.

`asymmetry_AIA`: AIA calculation for asymmetry:

$$A = \frac{RW_{10\%}}{LW_{10\%}}$$

for 10% left and right widths, $LW_{10\%}$ and $RW_{10\%}$. This value is undefined if these widths cannot be calculated.

`asymmetry_statistical`: Calculation of asymmetry using moments:

$$A = \frac{\mu_1 - t_R}{\sqrt{\mu_2}}$$

for first and second statistical moments μ_1 and μ_2 (see 6.4.5), and retention time t_R.

`skewness`: Metric similar to asymmetry. Not to be confused with statistical skewness, which is equal to the third normalized moment of retention time, $\tilde{\mu_3}$, as defined in 6.4.5. This measure of skewness is defined as:

$$S = \frac{RW_{10\%} + LW_{10\%}}{2 \cdot LW_{10\%}}$$

for 10% left and right widths, $LW_{10\%}$ and $RW_{10\%}$. This value is undefined if these widths cannot be calculated.

6.4.7 Resolution

Resolution measures how well two peaks are separated, using the width of two peaks and the distance in time between them. Resolution is always defined with regard to a reference peak; a peak cannot reference itself. The module uses the USP, EP/JP, and statistical moments methods each for up to five different reference peaks, where the string value in parenthesis is replaces `<type>` in the resolution field name:

1. previous main peak (`previous_main`): the previous quantified peak
2. next main peak (`next_main`): the next quantified peak
3. previous named peak (`previous_named`): the previous peak with a name that is not equal to `None`, `""`, or `"unknown"`.
4. next named peak (`next_named`): the next peak with a name that is not equal to `None`, `""`, or `"unknown"`.
5. specified reference peak (`reference`): a peak specified by name in the processing method (see section 2.2).

`resolution_ep_<type>`: European Pharmacopoeia calculation for resolution R, where for a reference peak *ref*,

$$R = 1.18 \left| \frac{t_{ref} - t_R}{W_{50\%,ref} + W_{50\%,R}} \right|$$

`resolution_usp_<type>`: US Pharmacopoeia calculation for resolution R, where for a reference peak *ref*,

$$R = 2 \left| \frac{t_{ref} - t_R}{\mathrm{BW}_{ref} + \mathrm{BW}_R} \right|$$

`resolution_statistical_<type>`: Statistical moments calculation for resolution R, where for a reference peak *ref*,

$$R = \left| \frac{t_{ref} - t_R}{2\left(\sqrt{\mu_{2,ref}} + \sqrt{\mu_2}\right)} \right|$$

`signal_to_noise`:

6.4.8 Theoretical Plates

`plates_EP`: European Pharmacopoeia calculation for theoretical plates TP, where for retention time t_R and 50% width $W_{50\%}$,

$$\mathrm{TP} = 5.54 \left(\frac{t_R}{W_{50\%}} \right)^2$$

`plates_USP`: US Pharmacopoeia calculation for theoretical plates TP, where for retention time t_R and baseline width BW,

$$\mathrm{TP} = 16 \left(\frac{t_R}{\mathrm{BW}} \right)^2$$

`plates_statistical`: Statistical Moments calculation for theoretical plates TP, where for retention time t_R and second statistical moment μ_2,

$$\mathrm{TP} = \frac{t_R^2}{\mu_2}$$

7 Samples

7.1 Sample Creation

7.1.1 Product Stability

7.1.2 Reaction Optimization

7.2 Samples JSON

References

[1] Yuri Kalambet et al. "Reconstruction of chromatographic peaks using the exponentially modified Gaussian function". In: *Journal of Chemometrics* 25.7 (2011), pp. 352–356. DOI: 10.1002/cem.1343.

[2] A. Daina, O. Michielin, and V. Zoete. “SwissADME: a free web tool to evaluate pharmacokinetics, drug-likeness and medicinal chemistry friendliness of small molecules”. In: *Sci Rep* 7 (2017), p. 42717. DOI: 10.1038/srep42717.

[3] Peter Zipfell and Darren Barrington-Light. *Technical Note 70698: Intelligent integration using Cobra and SmartPeaks*. Tech. rep. Thermo Fisher Scientific Inc., 2019. URL: https://assets.thermofisher.com/TFS-Assets/CMD/Technical-Notes/TN-70698-CDS-SmartPeaks-Cobra-TN70698-EN.pdf.

[4] Sergio Oller-Moreno et al. “Adaptive Asymmetric Least Squares baseline estimation for analytical instruments”. In: *2014 IEEE 11th International Multi-Conference on Systems, Signals & Devices (SSD14)*. 2014, pp. 1–5. DOI: 10.1109/SSD.2014.6808837.

Appendices

A Future Implementation

Future iterations of this module will include (in no particular order):

- A method to allow for different column types to produce different elution profiles
- Improved scientific model for estimating elution times based on solvent, column, and compound properties
- Additional spectra types
  - 3D UV-Vis
  - CAD/ELSD
  - Fluorescence
  - MS

B Classes

B.1 Injection

B.1.1 Instantiation

```
Injection(
    sample: Sample,
    method: InstrumentMethod,
    processing_method: ProcessingMethod,
    sequence: Sequence,
    system: System,
    user: str | None = "admin",
    injection_time: datetime.datetime | None = None,
)
```

B.1.2 Methods

B.1.3 Dictionary Keywords

B.2 System

B.3 Column

B.4 InstrumentMethod

B.5 ProcessingMethod

B.5.1 Instantiation

B.5.2 Methods

B.5.3 Dictionary Keywords

B.6 Compound

B.7 Sample

B.8 Chromatogram

B.9 PeakFinder

B.10 PeakList

B.11 Peak

C Example Input JSON

C.1 System

```
{
    "name": "Curie",
    "software": "Chromeleon",
    "manufacturer": "ThermoFisher",
    "modules": [
        {
            "name": "Sampler",
            "manufacturer": "Thermo Fisher Scientific Inc.",
            "type": "Autosampler",
            "detector_type": null,
            "part_number": "ACC-3000T",
            "serial_number": "23056882"
        },
        {
            "name": "PumpModule.Pump",
            "manufacturer": "Thermo Fisher Scientific Inc.",
            "type": "Pump",
            "detector_type": null,
            "part_number": "LPG-3400SD",
            "serial_number": "96014241"
        },
        {
            "name": "UV",
            "manufacturer": "Thermo Fisher Scientific Inc.",
            "type": "Detector",
            "detector_type": "UV-Vis",
            "part_number": "DAD-3000RS",
            "serial_number": "54202953",
            "firmware_version": "1.10.0",
            "driver_version": "7.3.2.10759"
        }
    ],
    "column": {}
    },
    "system_retention_time_offset": 0.021
}
```

C.2 Column

```
{
    "inner_diameter": {
        "value": 10,
        "unit": "mm"
    },
    "length": {
        "value": 150,
        "unit": "mm"
    },
    "type": "C18",
    "serial_number": "1995032",
    "injection_count": 0,
    "particle_size": {
        "value": 5,
        "unit": "um"
    }
}
```

C.3 Instrument Method

```
{
    "name": "column_quality_check",
    "run_time": 8,
    "mobile_phases": [
        {
            "name": "water",
            "id": "A"
        },
        {
            "name": "THF",
            "id": "B"
        },
        {
            "name": "methanol",
            "id": "C"
        },
        {
            "name": "acetonitrile",
            "id": "D"
        }
    ],
    "mobile_phase_gradient_steps": [
        {
            "time": 0.0,
            "flow": 5.0,
            "percent_a": 100.0,
            "percent_b": 0.0,
            "percent_c": 0.0,
            "percent_d": 0.0,
            "curve": 5
        },
        {
            "time": 8.0,
            "flow": 5.0,
            "percent_a": 85.0,
            "percent_b": 15.0,
            "percent_c": 0.0,
            "percent_d": 0.0,
            "curve": 5
        }
    ],
    "detection": {
        "uv_vis_parameters": [
            {
                "name": "UV_VIS_1",
                "wavelength": 250,
                "bandwidth": 4
            },
            {
                "name": "UV_VIS_2",
                "wavelength": 270,
                "bandwidth": 4
            }
        ]
    },
    "creation": {
        "computer": "CHROMELEON2",
        "comment": null,
        "time": "2020-08-31T12:18:06.01Z",
        "user": "cmadmin",
        "data_vault_name": "CHROMELEON2/
 ChromeleonLocal"
    },
    "last_update": {
        "computer": "CHROMELEON2",
        "comment": null,
        "time": "2020-08-31T12:24:33.75Z",
        "user": "cmadmin",
        "data_vault_name": "CHROMELEON2/
 ChromeleonLocal"
    },
    "sample_introduction": {
        "dilution_factor": 1.0,
        "injection_volume": 20.0,
        "weight": 1.0,
        "internal_standard_amount": 1.0,
        "autodilution_ratio": 0.0
    }
}
```

C.4 Processing Method

```
{
    "name": "column performance test",
    "detection_parameters": {
        "background_noise_range": {
            "minimum": 0,
            "maximum": 250
        },
        "noise_threshold_multiplier": 10,
        "peak_limit": 5,
        "minimum_height": {
            "type": "baseline_noise_multiplier",
            "value": 10
        },
        "minimum_area": 0.3
    },
    "resolution_reference": "TS-8391",
    "peak_identification": [
        {
            "min_time": 2.0,
            "max_time": 2.5,
            "method": "largest",
            "name": "TS-8391"
        },
        {
            "min_time": 3.6,
            "max_time": 4.1,
            "method": "largest",
            "name": "klimavizinib",
            "calibration": {
                "type": "linear",
                "amount_unit": "umol/mL",
                "points": [
                    {
                        "area": 101.2,
                        "amount": 2.5
                    },
                    {
                        "area": 99.2,
                        "amount": 2.5
                    },
                    {
                        "area": 1002,
                        "amount": 25
                    },
                    {
                        "area": 987.4,
                        "amount": 25
                    },
                    {
                        "area": 10,
                        "amount": 0.25
                    },
                    {
                        "area": 9.3,
                        "amount": 0.25
                    },
                    {
                        "area": 9.2,
                        "amount": 0.25
                    },
                    {
                        "area": 0.913,
                        "amount": 0.025
                    },
                    {
                        "area": 1.01,
                        "amount": 0.025
                    },
                    {
                        "area": 4.989,
                        "amount": 0.125
                    },
                    {
                        "area": 5.12,
                        "amount": 0.125
                    }
                ]
            }
        },
        {
            "min_time": 5.9,
            "max_time": 6.4,
            "method": "largest",
            "name": "caffeine"
        }
    ]
}
```

D Adding to the Compound Library

The compound library is stored in the