Recipe 3.1. Creating a MS peak list

Table of contents

1. Problem

We want to build a new MS peak list.

2. Solution

A simple peak list needs a mass-to-charge ratio peak'values.

```
// the m/z values
double[] mzs = new double[] { /*** list of sorted doubles ***/ };

// the builder needs mzs to create a peak list
PeakList pl =
        new PeakListImpl.Builder(mzs).build();
```

The builder proposes a set of optional parameters via supplementary methods:

```
// here is an experimental peak list
PeakList plExperimental =
        new PeakListImpl.Builder(mzs).intensities(intensities).build();

// here is a peak list with annotations
PeakList plAnnotated =
        new PeakListImpl.Builder(mzs).annotations(annotations).build();

// by default, null intensity peaks are kept, you can enable a few options
double[] mzs = new double[] {1.1, 2.2, 3.3, 4.2, 4.4, 5.5, 6.6, 7.7, 8.8};
double[] intensities = new double[] {4, 6, 0, 10, 1, 3, 0, 7, 9};

// 1. removal of all peaks where intensity is null
pl = new PeakListImpl.Builder(masses).intensities(intensities)
        .discardNonIntensePeaks().build();

Assert.assertEquals(7, pl.size());

// 2a. reset null intensities (set with min intensity/100)
pl = new PeakListImpl.Builder(masses).intensities(intensities)
        .resetNonIntensePeaks().build();

Assert.assertEquals(0.01, pl.getIntensityAt(2), 0.01);

// 2b. reset null intensities (set with min intensity/100)
pl = new PeakListImpl.Builder(masses).intensities(intensities)
        .resetNonIntensePeaks(1000).build();

Assert.assertEquals(0.001, pl.getIntensityAt(2), 0.001);

// sub peak lists
// a 1st way
PeakList subPl =
```

```
        new PeakListImpl.Builder(pl).indices(Arrays.asList(2, 3)).build();

// a second way
subPl =
        new PeakListImpl.Builder(pl).from(2).to(4).build();
```

Once peak lists have been created, you may have to get some informations. `JPLMSPeakLists` provides a set of utilitary methods for peak lists:

```
JPLMSPeakLists peakListUtil = JPLMSPeakLists.getInstance();

// get the sorted indices of intensities (ascending)
peakListUtil.getSortedIndexIntensityUp(plExperimental);

// get the intensity peak max index
peakListUtil.getMaxIntensityIndex(plExperimental);

// merge peaks in a new peak list
peakListUtil.merge(plExperimental, pl);

// get an interval over peak indices in an interval of mzs
Interval interval = peakListUtil.getIndexRange(pl, 100, 1000);
```

3. Discussion

4. See Also