OHSU Cancer Institute Adopter Site
Task 2.8: mzXML Compatibility Test Activity Logs

Tester: Ted Laderas (laderast@ohsu.edu)

Note: All testing was done on version 1.1. However, a comparison between the source code of version 1.1 and 3.0 show no differences in the mzXML() function tested here.

Test ID M1

Objective: Test mzXML compatibility of Q5 by comparing range of scan size within a mzXML file to matrix output by Q5 mzXML() function.

Procedure:
1. Load mzXML file using load.mzXML function in caMassClass package
2. For each scan in mzXML file, store length of scan
3. Load mzXML file using Q5 mzXML function
4. Note number of columns (scan length of matrix)
5. Report range of scan lengths found in step 2 and compare with number of columns of matrix in step 4

Results: FAIL.

An average error of 7957.388 points points was found between the actual scan length and the Q5 selected scan length.

Table M1.1 below shows results for the individual scans of data.

Table M1.1. Results of Test R1.

actual scanlength	q5 reported scanlength	Q5 error
36548	36548	0
43259	36548	6711
45061	36548	8513
47693	36548	11145
48303	36548	11755
48474	36548	11926
45936	36548	9388
41899	36548	5351
48097	36548	11549
42414	36548	5866
43956	36548	7408
48564	36548	12016
50854	36548	14306
43394	36548	6846
46906	36548	10358
43695	36548	7147

43554	36548	7006
44082	36548	7534
47745	36548	11197
39861	36548	3313
39889	36548	3341
41954	36548	5406
45191	36548	8643
45775	36548	9227
43033	36548	6485
43995	36548	7447
39899	36548	3351
40936	36548	4388
47651	36548	11103
40086	36548	3538
43002	36548	6454
43924	36548	7376
42532	36548	5984
44189	36548	7641
45934	36548	9386
44911	36548	8363
45689	36548	9141
44479	36548	7931
49535	36548	12987
45543	36548	8995
44146	36548	7598
49463	36548	12915
46741	36548	10193
45442	36548	8894
41670	36548	5122
41870	36548	5322
45498	36548	8950
26493	36548	-10055
40889	36548	4341

Test ID M2

Objective: Test mzXML compatibility of Q5 by comparing number of scans in mzXML file to matrix output by Q5 mzXML() function.

Procedure:
1. Load mzXML file using load.mzXML function in caMassClass
2. Count number of scans in mzXML file by using length() function
3. Load mzXML file using Q5 mzXML function
4. Count number of rows (number of scans)
5. Compare number of scans found in step 2 to number of rows in step 4.

Results: PASS.

The reported number of scans (49) was identical to the number of rows (which correspond to scans) in the matrix produced by the mzXML() function.

Test ID M3

Objective: Test mzXML compatibility as part of entire Q5 workflow.

Procedure:
1) Load case and control data using mzXML() function
2) Build 10 random test and train splits with a train/test ratio of 0.75
3) Run Q5 using mzXML loaded data and test train splits
4) Report results

Results: FAIL.

The Q5 workflow will not run because the column length (which corresponds to the scan length) of the case and control matrices are different. This is a serious error condition that could arise if a user is using data that is not interpolated. This also challenges the claim that Q5 can work with raw data. At a bare minimum of processing, interpolating such that the intensities fall on common m/z values is recommended.

Number of columns for case data: 36548
Number of columns for control data: 50379

In addition, running this script generated this error:

```
Error in rbind(deparse.level, ...) : number of columns of matrices must
match (see arg 2)
```

Test ID M4

Objective: Compare output of Q5 algorithm using files loaded with Q5 mzXML() function to output of Q5 algorithm using RProteomics interpolated data.

Procedure:
1) Select 10 random partitions of data file with 75/25 test train splits
2) Load interpolated data into memory and partition using random partitions
3) Run Q5 algorithm on all 10 sets of random partitions
4) Load mzXML data into memory using Q5 functionality – subset data using the same random partitions
5) Run Q5 algorithms on all 10 sets of random partitions
6) Compare statistical output from steps 3 and 5.

Result: UNCLEAR

This test cannot be run because the mzXML compatibility does not load data that has disparate scan lengths (see test M3 above). As this is a common situation with this data, Q5 is unable to work with uninterpolated raw data.