isobaricquant

Villen Lab

Jun 08, 2022

CONTENTS:

CHAPTER

ONE

INTRODUCTION

IsobaricQuant, a Java-based software tool for the quantification, visualization, and filtering of isobarically-labeled peptides. IsobaricQuant is a cross-platform standalone tool that can be operated via an intuitive graphical user interface (GUI), or integrated into custom pipelines via command line.

For input, it requires the mzML file of the MS run, a CSV file containing peptide spectral matches (PSM) obtained from a search engine, and a user-supplied text configuration file. Optionally, isotopic impurity correction can be performed using a reporter ion isotopic distribution CSV file. IsobaricQuant supports MS2 and MS3 level reporter ion quantification for iTRAQ up to 8-plex, TMT up to 11-plex and TMTpro up to 16-plex.

An integrated viewer allows visual inspection of isolation window purity in MS1 scans, reporter ion quantification, and fragment ion assignment. For MS3 acquisition methods, it further enables visual assessment of SPS ion selection.

CHAPTER

TWO

GETTING STARTED

2.1 Dependencies

In order to run `IsobaricQuant` Java 17 is needed, first of all make sure you don't have the correct version installed already with the following command:

```
java --version
```

If you don't have Java 17 installed follow the official instructions on how to install Java 17 for your platform:

2.1.1 Microsoft Windows

Installation of the JDK on Microsoft Windows Platforms: Instructions include steps to install the JDK on 64-bit Microsoft Windows operating systems.

2.1.2 Linux

Installation of the JDK on Linux Platforms: Instructions include installation from Archive files and Red Hat Package Manager (RPM).

2.1.3 macOS

Installation of the JDK on macOS: Instructions include steps to install the JDK on the macOS platform.

2.2 Installation

Since IsobaricQuant is a java application, its installation is straightforward, first you need to download the jar file for your platform:

- Windows
- MacOS
- Linux

Once the jar file is downloaded, place it in a folder of your choice, open a terminal and navigate to the jar file location. That's it!

2.3 How to run IsobaricQuant?

After downloading the jar file, `IsobaricQuant` can be used via GUI:

```
java --add-opens java.base/java.lang=ALL-UNNAMED -jar IsobaricQuant.jar
```

Or it can be used via command line:

```
java -jar IsobaricQuant.jar -c <config_file> -mzf <mz_file> -h <hits_file> -o <output_
↪folder>
```

2.3.1 IsobaricQuant CLI mode

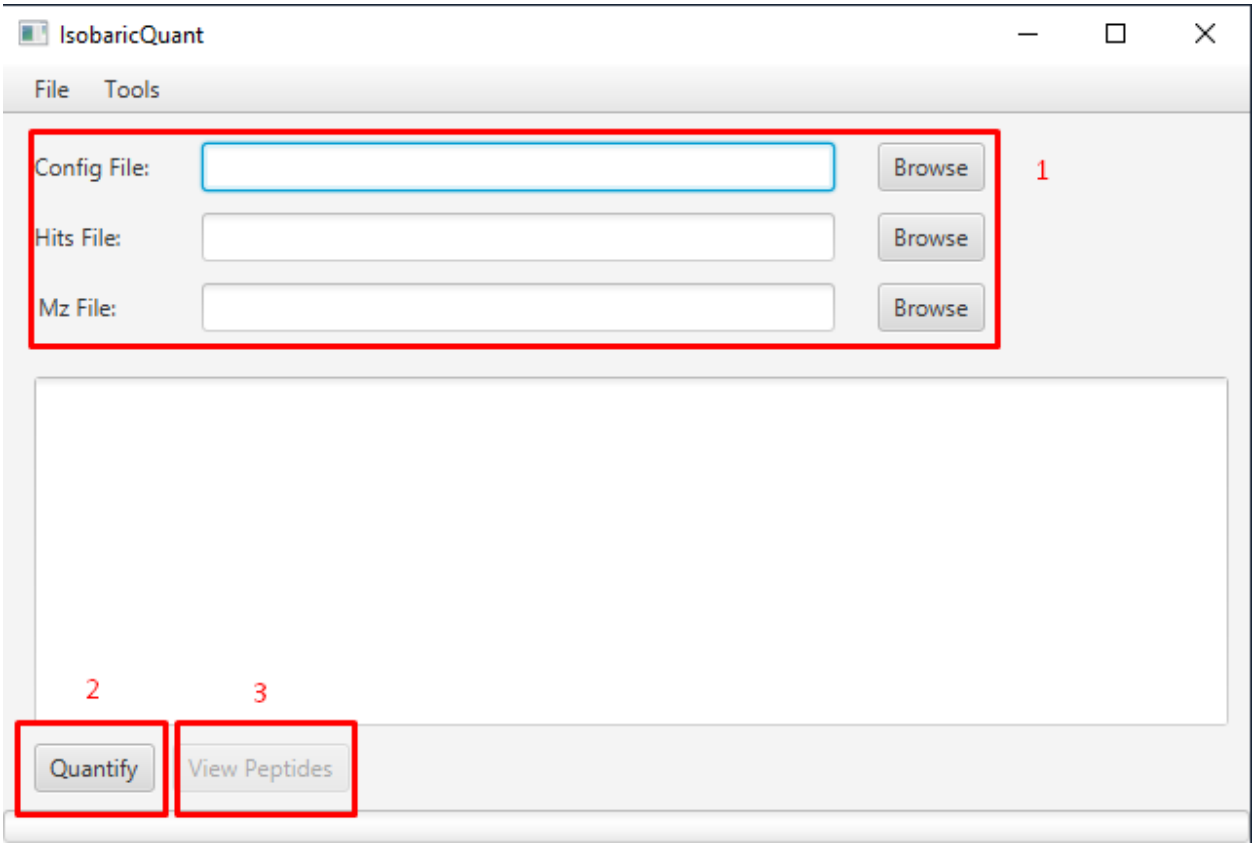
```
java -jar IsobaricQuant.jar -c <config_file> -mzf <mz_file> -h <hits_file> -o <output_
↪folder>
```

The CLI mode doesn't include a viewer, as soon as the command is executed the quantification starts, once is completed an output folder is generated with all the output files detailed in the previous section. The generated files can be opened using the GUI version at any time. The CLI version will allow the user to include IsobaricQuant in any pipeline

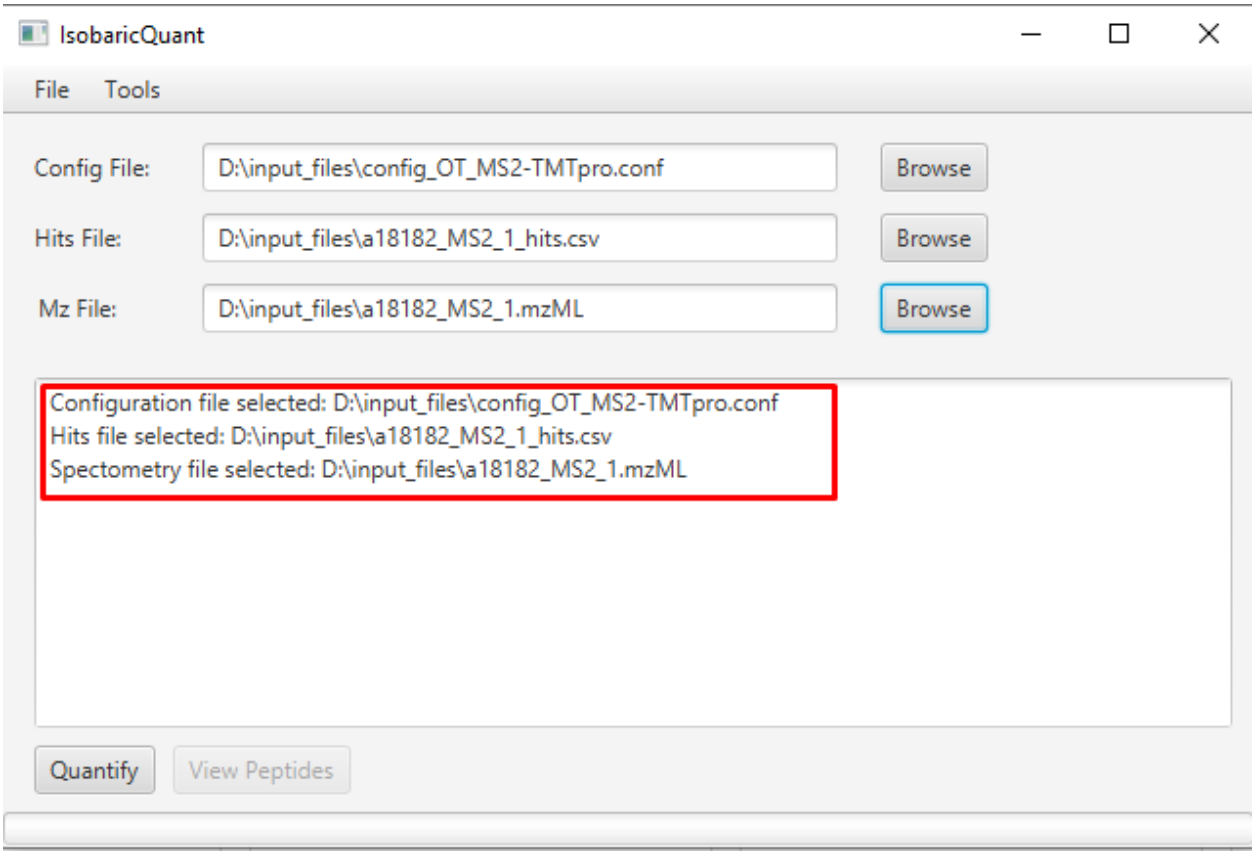
2.3.2 IsobaricQuant GUI mode

On the GUI mode you will find a window with some options:

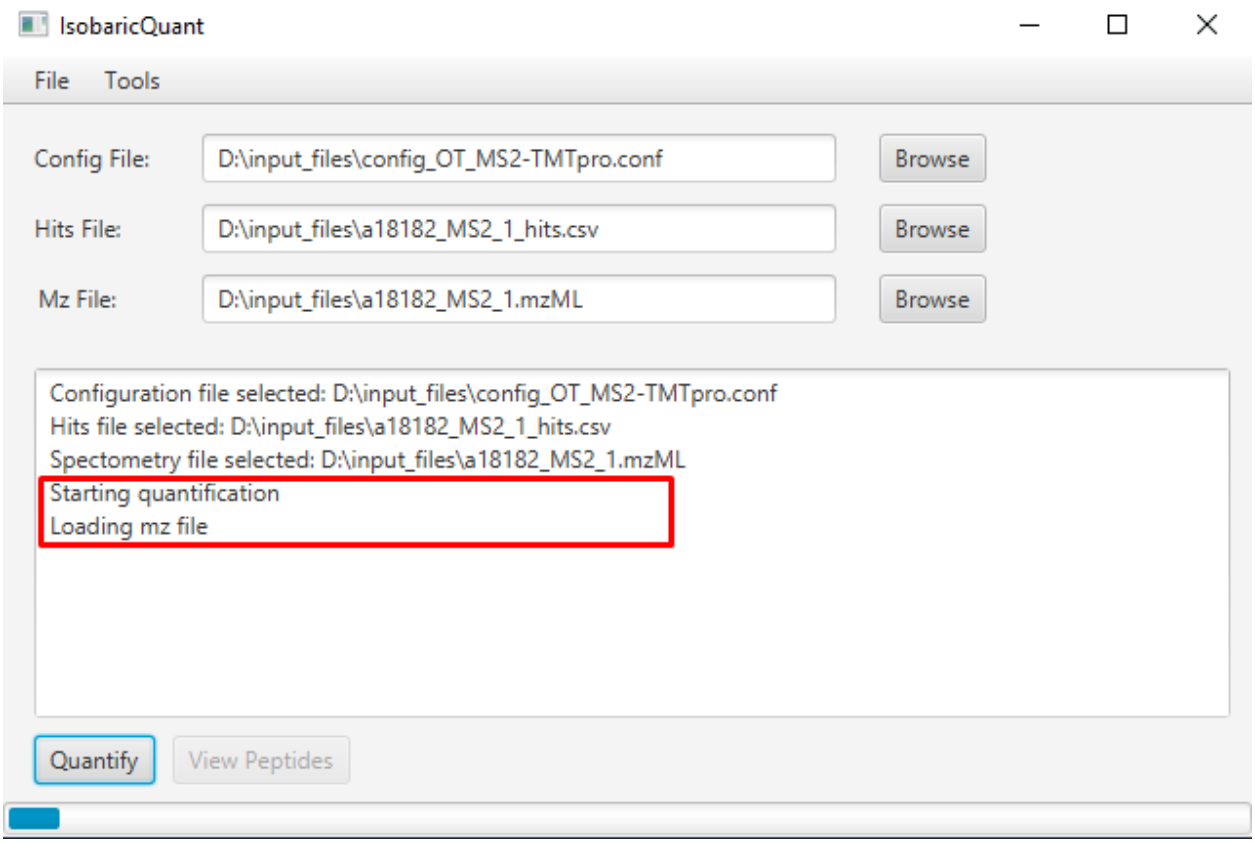
1. Select the input files, for more information about input files refer to the `Input/Output Files` section
2. Once all the files have been selected preceed with the quantification using the Quantify button
3. When the quantification is completed you can proceed to consult the peptide information



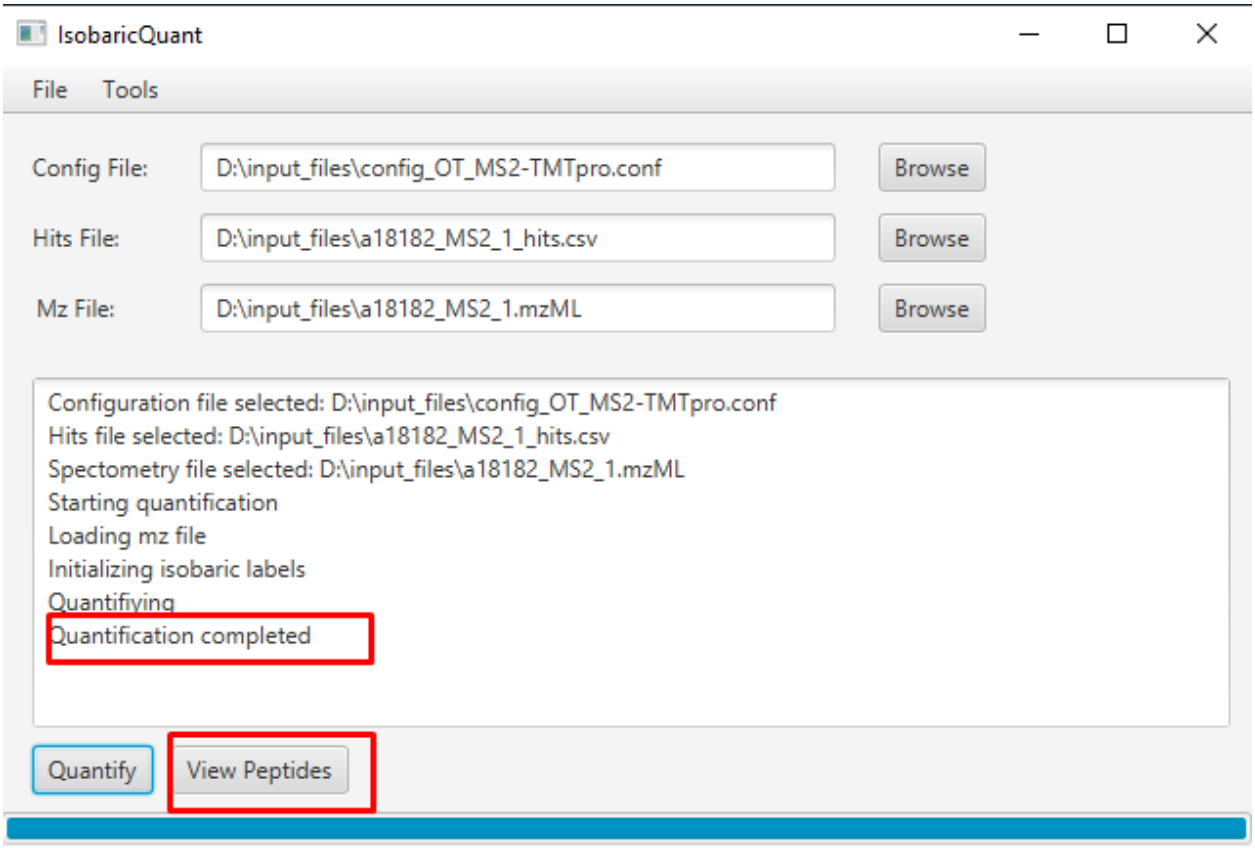
In the log window you can check if the path of the selected files is correct:

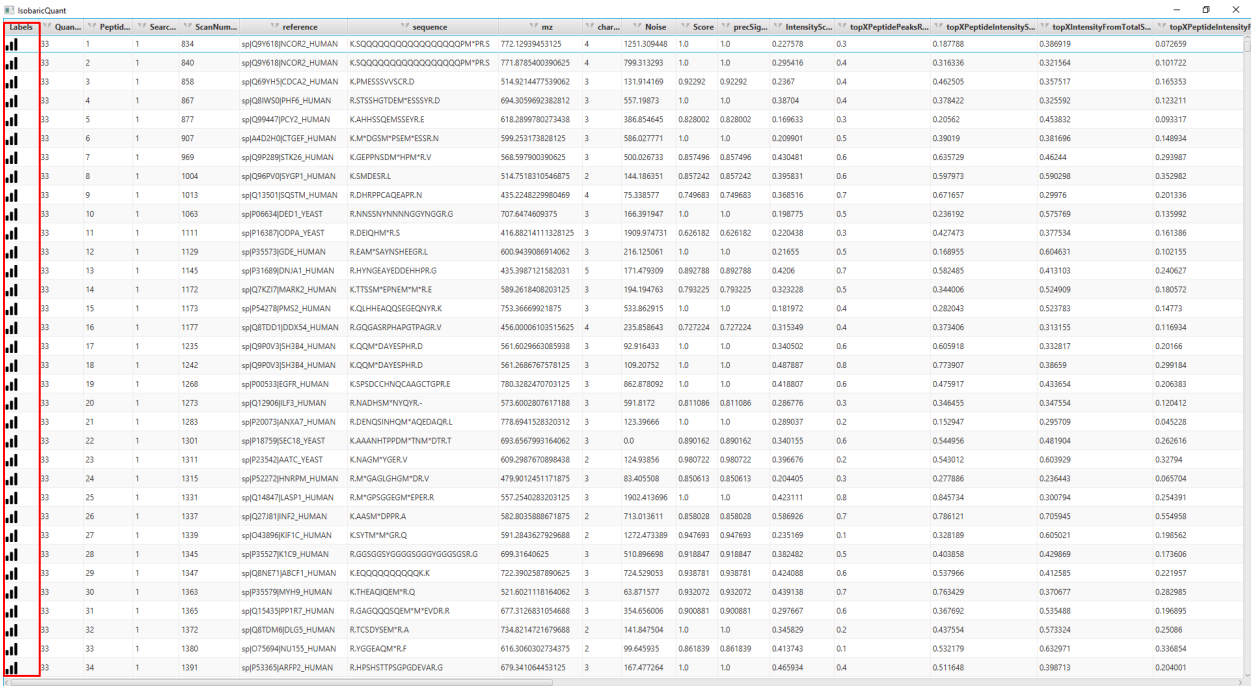


Once you click the Quantify button, the quantification will start, indicating the steps involved on the process:



When the quantification is done, an output folder is created with the quantification information. The GUI includes a viewer, that will allow the user to view the quantification matching information including fragment ions





Labels	Quan...	Peptid...	Searc...	ScanNum...	reference	sequence	mz	char...	Noise	Score	precSig...	IntensitySc...	topXPeptidePeaksR...	topXPeptideIntensityS...	topXIntensityFromTotalS...	topXPeptideIntensityF
33	1	1	834	sp|Q9Y618|NCOR2_HUMAN	K.SQQQQQQQQQQQQQQQQQPM*PR.S	772.12939453125	4	1251.309448	1.0	1.0	0.227578	0.3	0.187788	0.386919	0.072659
33	2	1	840	sp|Q9Y618|NCOR2_HUMAN	K.SQQQQQQQQQQQQQQQQQPM*PR.S	771.8785400390625	4	799.313293	1.0	1.0	0.295416	0.4	0.316336	0.321564	0.101722
33	3	1	858	sp|Q69YH5|CDCA2_HUMAN	K.PMESSSVVSCR.D	514.9214477539062	3	131.914169	0.92292	0.92292	0.2367	0.4	0.462505	0.357517	0.165353
33	4	1	867	sp|Q8IWS0|PHF6_HUMAN	R.STSSHGTDEM*ESSSYR.D	694.3059692382812	3	557.19873	1.0	1.0	0.38704	0.4	0.378422	0.325592	0.123211
33	5	1	877	sp|Q99447|PCY2_HUMAN	K.AHHSSQEMSSEYR.E	618.2899780273438	3	386.854645	0.828002	0.828002	0.169633	0.3	0.20562	0.453832	0.093317
33	6	1	907	sp|A4D2H0|CTGEF_HUMAN	K.M*DGSM*PSEM*ESSR.N	599.253173828125	3	586.027771	1.0	1.0	0.209901	0.5	0.39019	0.381696	0.148934
33	7	1	969	sp|Q9P289|STK26_HUMAN	K.GEPPNSDM*HPM*R.V	568.597900390625	3	500.026733	0.857496	0.857496	0.430481	0.6	0.635729	0.46244	0.293987
33	8	1	1004	sp|Q96PV0|SYGP1_HUMAN	K.SMDESR.L	514.7518310546875	2	144.186351	0.857242	0.857242	0.395831	0.6	0.597973	0.590298	0.352982
33	9	1	1013	sp|Q13501|SQSTM_HUMAN	R.DHRPPCAQEAPR.N	435.2248229980469	4	75.338577	0.749683	0.749683	0.368516	0.7	0.671657	0.29976	0.201336
33	10	1	1063	sp|P06634|DED1_YEAST	R.NNSSNYNNNNGGYNGGR.G	707.6474609375	3	166.391947	1.0	1.0	0.198775	0.5	0.236192	0.575769	0.135992
33	11	1	1111	sp|P16387|ODPA_YEAST	R.DEIQHM*R.S	416.88214111328125	3	1909.974731	0.626182	0.626182	0.220438	0.3	0.427473	0.377534	0.161386
33	12	1	1129	sp|P35573|GDE_HUMAN	R.EAM*SAYNSHEEGR.L	600.9439086914062	3	216.125061	1.0	1.0	0.21655	0.5	0.168955	0.604631	0.102155
33	13	1	1145	sp|P31689|DNJA1_HUMAN	R.HYNGEAYEDDEHHPR.G	435.3987121582031	5	171.479309	0.892788	0.892788	0.4206	0.7	0.582485	0.413103	0.240627
33	14	1	1172	sp|Q7KZI7|MARK2_HUMAN	K.TTSSM*EPNEM*M*R.E	589.2618408203125	3	194.194763	0.793225	0.793225	0.323228	0.5	0.344006	0.524909	0.180572
33	15	1	1173	sp|P54278|PMS2_HUMAN	K.QLHHEAQQSEGEQNYR.K	753.36669921875	3	533.862915	1.0	1.0	0.181972	0.4	0.282043	0.523783	0.14773
33	16	1	1177	sp|Q8TDD1|DDX54_HUMAN	R.GQGASRPHAPGTPAGR.V	456.00006103515625	4	235.858643	0.727224	0.727224	0.315349	0.4	0.373406	0.313155	0.116934
33	17	1	1235	sp|Q9P0V3|SH3B4_HUMAN	K.QQM*DAYESPHR.D	561.6029663085938	3	92.916433	1.0	1.0	0.340502	0.6	0.605918	0.332817	0.20166
33	18	1	1242	sp|Q9P0V3|SH3B4_HUMAN	K.QQM*DAYESPHR.D	561.2686767578125	3	109.20752	1.0	1.0	0.487887	0.8	0.773907	0.38659	0.299184
33	19	1	1268	sp|P00533|EGFR_HUMAN	K.SPSDCCHNQCAAGCTGPR.E	780.3282470703125	3	862.878092	1.0	1.0	0.418807	0.6	0.475917	0.433654	0.206383
33	20	1	1273	sp|Q12906|ILF3_HUMAN	R.NADHSM*NYQYR.-	573.6002807617188	3	591.8172	0.811086	0.811086	0.286776	0.3	0.346455	0.347554	0.120412
33	21	1	1283	sp|P20073|ANXA7_HUMAN	R.DENQSINHQM*AQEDAQR.L	778.6941528320312	3	123.39666	1.0	1.0	0.289037	0.2	0.152947	0.295709	0.045228
33	22	1	1301	sp|P18759|SEC18_YEAST	K.AAANHTPPDM*TNM*DTR.T	693.6567993164062	3	0.0	0.890162	0.890162	0.340155	0.6	0.544956	0.481904	0.262616
33	23	1	1311	sp|P23542|AATC_YEAST	K.NAGM*YGER.V	609.2987670898438	2	124.93856	0.980722	0.980722	0.396676	0.2	0.543012	0.603929	0.32794
33	24	1	1315	sp|P52272|HNRPM_HUMAN	R.M*GAGLGHGM*DR.V	479.9012451171875	3	83.405508	0.850613	0.850613	0.204405	0.3	0.277886	0.236443	0.065704
33	25	1	1331	sp|Q14847|LASP1_HUMAN	R.M*GPSGGEGM*EPER.R	557.2540283203125	3	1902.413696	1.0	1.0	0.423111	0.8	0.845734	0.300794	0.254391
33	26	1	1337	sp|Q27J81|INF2_HUMAN	K.AASM*DPPR.A	582.8035888671875	2	713.013611	0.858028	0.858028	0.586926	0.7	0.786121	0.705945	0.554958
33	27	1	1339	sp|O43896|KIF1C_HUMAN	K.SYTM*M*GR.Q	591.2843627929688	2	1272.473389	0.947693	0.947693	0.235169	0.1	0.328189	0.605021	0.198562
33	28	1	1345	sp|P35527|K1C9_HUMAN	R.GGSGGSYGGGGSGGGYGGGSGSR.G	699.31640625	3	510.896698	0.918847	0.918847	0.382482	0.5	0.403858	0.429869	0.173606
33	29	1	1347	sp|Q8NE71|ABCF1_HUMAN	K.EQQQQQQQQQK.K	722.3902587890625	3	724.529053	0.938781	0.938781	0.424088	0.6	0.537966	0.412585	0.221957
33	30	1	1363	sp|P35579|MYH9_HUMAN	K.THEAQIQEM*R.Q	521.6021118164062	3	63.871577	0.932072	0.932072	0.439138	0.7	0.763429	0.370677	0.282985
33	31	1	1365	sp|Q15435|PP1R7_HUMAN	R.GAGQQQSQEM*M*EVDR.R	677.3126831054688	3	354.656006	0.900881	0.900881	0.297667	0.6	0.367692	0.535488	0.196895
33	32	1	1372	sp|Q8TDM6|DLG5_HUMAN	R.TCSDYSEM*R.A	734.8214721679688	2	141.847504	1.0	1.0	0.345829	0.2	0.437554	0.573324	0.25086
33	33	1	1380	sp|O75694|NU155_HUMAN	R.YGGEAQM*R.F	616.3060302734375	2	99.645935	0.861839	0.861839	0.413743	0.1	0.532179	0.632971	0.336854
33	34	1	1391	sp|P53365|ARFP2_HUMAN	R.HPSHSTTPSGPGDEVAR.G	679.341064453125	3	167.477264	1.0	1.0	0.465934	0.4	0.511648	0.398713	0.204001


IsobaricQuant
MS1 Scan 860 | RT: 268.43
Score: 100.00%
Intensity
M/Z
Detected peptide
MS1 peaks
Elution Window
MS2 Scan 867 | RT: 269.68780517578125
[694.306]
MS2 Scan 867 | RT: 269.69
[694.306]
R.STSSHGTDEM*ESSSYR.D(1)
MZ: 1277.494
Intensity: 4185
2b(6)
1b(4)
1y(8)
Noise
Precursors(0)
MS2 peaks
zero
127N
127C
128N
128C
129N
129C
130N
130C
131N
131C
132N
132C
133N
133C
134N
Noise peaks
Noise level

2.4 Input Files

2.4.1 Configuration file


```
{
    "generateFragmentIonsFile": true,
    "ms2NoiseWindowTolerance": 1,
    "searchFragmentWIndowTolerance": 1.5,
    "isoMethod": "TMTpro",
    "confLabels": "all",
    "PPMTolerance": 20,
    "MS1PPMTolerance": 20,
    "MS2PPMTolerance": 20,
    "massType": "HCD_FRAGMENTATION",
    "scanLevel": 2,
    "scoreType": "ISOTOPE_DISTRIBUTION",
    "ms1PrecWindowDaltons": 0.35,
    "neutralLossesFI": "63.98,18.01,17.03",
    "neutralLosses": "63.98,18.01,17.03",
    "topXNum": 10,
    "ms1DepthSearch": 5,
    "recalcIntensities": false,
    "dataSheetName": "TMT11_TL277832",
    "modifications": {
        "varMods": "15.9949146202 M 42.01056468472 n",
        "consMods": "pep_n 304.2071453165, C 57.02146372118, K 304.2071453165",
        "varTermParams": ""
    },
    "mzFilePath": "",
    "hitImpurity": "master_scan",
    "searchMethod": "most_intense"
}
```


Parameter description

isoMethod: Isobaric Methods
Values: iTRAQ4plex,iTRAQ8plex,TMTDuplex,TMT6plex,TMT10plex,TMT11plex,TMTpro
Default: iTRAQ4plex

massType: mass type
Values: HCD fragmentation,Neutral mass
Default: HCD fragmentation

scanLevel: Scan level for isobaric quantification
Values: MS2,MS3
Default: MS2

scoreType: Score calculation

Values:
Isotope distribution: peptide isotopes intensity/total intensity in isotopes window.
Reporters intensity: reporters intensity/total intensity in window.
Reporters found: num. reporters found/num. Reporters
Default: Reporters intensity

confLabels: Labels to be searched in (use comma separated values. ex: 114,115,116,117). Write 'all' to search all of them. For TMT10plex, use label reagent: 126,127N,127C,128N,128C...
Values: input by user
Default: all

recalcIntensities: If activated, it will recalculate intensities using the product data sheet.
Default: not activated

dataSheetName: Product data sheet
Values: TMT10plex_QI218066,TMT11_TL277832
Default: TMT10plex_QI218066

PPMTolerance: PPM Tolerance for MS3
Values: input by user
Default: 10

MS1PPMTolerance: PPM Tolerance for MS1
Values: input by user
Default: 10

MS2PPMTolerance: PPM Tolerance for MS2
Values: input by user
Default: 1000

ms1PrecWindowDaltons: Dalton window that will be used to calculate the precursor signal percentage in the MS1 scan.
Values: input by user
Default: 1

searchFragmentWindowTolerance: MS2 precursor window (Da)
Values: input by user
Default: 2.5

generateFragmentIonsFile: Generates a file with all peptides' fragment ions.
Default: not activated

hitImpurity:
Values: master_scan,weighted_avg
The precursor signal is calculated using the weighted signal or signal percentage
If it's not weighted avg, then it will be `precursor int / intensities` in isolation window (calculated using `ms1PrecWindowDaltons`)

```
signalPerc = precIntensity / allIntensities
```

if it's weighted, then it will calculate the same thing as the above but for the current ms1, its previous ms1 and its posterior ms1. The result is the weighted signal, calculated using the RT distance to the current ms1

```
(dPivotCurrent / totalDistances) * currentSignal + (dPivotPre / totalDistances) *␣
↪preSignal + (dPivotPost / totalDistances) * postSignal
```

Default: master_scan

searchMethod: Search method used to select a TMT reporter ion peak
Values: most_intense, least_intense, lower_ppm_error
Default: most_intense

ms1DepthSearch: Number of MS1 scans depth to search for MS3 scans
Values: input by user
Default: 5

topXNum: Top X most intense ions used to calculate scores
Values: input by user
Default: 10

neutralLosses: Comma separated neutral loss masses (ex: 97.977,97.995). The peak will be searched for these NL.
Values: input by user, e.g. "63.98,18.01,17.03"

neutralLossesFI: Comma separated neutral loss masses (ex: 97.977,97.995). The fragment ions will be searched for these NL.
Values: input by user, e.g. "63.98,18.01,17.03"

2.4.2 Mz File

mzML or mzXML input file. The instrument raw file can be converted from existing tools such as msconvert.

2.4.3 Hits file

Comma separated values of the identified peptides. A conversion of the output file of a search algorithm such as Comet is required.

The values are the following:

Search ID: integer value to identify the search where the peptide was detected.
Peptide ID: integer value that identifies the peptide
Sequence: peptide sequence
Reference: protein sequence reference
Charge: peptide charge
Start Scan: peptide MS2 Scan
M/Z: peptide m/z

2.5 Output Files

Once IsobaricQuant finishes the quantification an output folder will be created on the same folder where the IsobaricQuant jar file resides.

The output folder contains 4 csv files with the result of the quantification

2.5.1 isofrag.csv

File containing matching information for the peptide's fragment ions

Peptide ID: peptide id introduced in the input file
Quant ID: quantification id (unused field at 0)
Fragment ion charge
Fragment ion type (a, b…)
Fragment ion position
Fragment ion mz
Fragment ion mz difference
Fragment ion intensity
Fragment ion matched (true or false)

2.5.2 isolab.csv

File containing the quantification data

Peptide ID: peptide id introduced in the input file
Quant ID: quantification id (unused field at 0)
Label ID: string identificator of the label
Label MZ: Matching peak MZ (theoretical if nothing matched)
MZ Variance: mz variance between the theoretical mz and the matched mz (0 otherwise)

Scan: matched scan number
Scan: matched scan number (duplicated)
Number of scans: 1 default value
Unused parameter: 0 default value
Unused parameter: 0 default value
Label intensity: Matching peak intensity (0 otherwise)
MS2 Scan: MS2 scan introduced in the input file
Retention Time: matched scan retention time

2.5.3 isopep.csv

File containing the basic information to be used to match the results with the other files

Quant ID: quantification id (unused field at 0)
Peptide ID: peptide id introduced in the input file
Search ID: search id introduced in the input file
MS2 Scan: MS2 scan introduced in the input file
Noise: Calculated noise in the scan
Score: calculated score as specified in the configuration file (isotope distribution, reporters intensity or reporters found)

2.5.4 isopep_extra.csv

File containing the quantification scores. Please, check the scores description for further information.

Peptide ID: peptide id introduced in the input file
Quant ID: quantification id (unused field at 0)
Precursor signal
PeptideIntensityScore
MS2 TopXPeptidePeaksRatio
MS2 PeptideTopXIntensityScore
MS2 TopXIntensityFromTotalScore
MS2 PeptideTopXIntensityFromTotalScore
MS2 TopPeakIntensityScore
MS2 isTopPeakFromPeptide: 1 (yes), 0 (no)
MS2 isTopPeakFromPeptideNeutralLoss(): 1 (yes), 0 (no)
MS2 TopPeakIntensityTopXScore()
MS2 TopPeakMass
msnTotalSignal
precTPIntRatio
precRepIntRatio
precRepIntRatio
precTotalSignal
totalSignalSPSWind
Scan level: 2 or 3

MS1 scan number: 0 if not found
MS2 scan number
MS3 scan number: 0 if not found or the selected level was 2
MS1 retention time: 0 if not found
MS2 retention time
MS3 retention time: 0 if not found or the selected level was 2
SPS masses: their mz separated by ;

CHAPTER

THREE

TEST FILES

Once IsobaricQuant is ready to run, you can use the following input and output sample files to test the jar and its dependencies are working properly

Input files

Output files

CHAPTER

FOUR

ALGORITHM

For every peptide hit of the input file submitted, we get the corresponding scan (MS2 or MS3 depending on scan level requested) and search for every label the nearest peak to their mz applying a window (+/- PPMTolerance). The peak selection is done using the [searchMethod] indicated by the user.

If the user selected the option to recalculate the intensities based on the datasheet provided, then it will recalculate them using the corresponding equation matrix with the observed intensities. If we haven't obtained enough data to solve the matrix, then it will use the observed intensities instead.

4.1 Noise

Removes the intensities from the peaks of the scan that are within the labels (low and high mz), and the noise is the average of the remaining peaks intensities. The noise is calculated by obtaining all the intensities within the lowest mz label (- PPMTolerance applied) - 1, and the highest one (+ PPMTolerance applied) + 1, and then removing the observed intensities of the labels. The noise will be the average of these intensities.

4.2 Excluded Peaks

For all score calculations in MS2, the following peaks are excluded:

- MS1 non-fragmented precursor M+H
- TMT reporter ions (except for msnTotalSignal)
- TMTpro complementary ion clusters
  - Main series [(M+H) - 163] to [(M+H) - 151]
  - Second series [(M+H) - 180] to [(M+H) - 168]
  - Third series [(M+H) - 134] to [(M+H) - 122]
- TMT (6/10/11-plex) complementary ion clusters
  - Main series [(M+H) - 160] to [(M+H) - 151]
  - Second series [(M+H) - 177] to [(M+H) - 168]

In many scans, we observed TMT complementary ion clusters to be highly abundant, but without much predictive value, especially for SPS score calculations, as they are excluded from SPS ion selection as well.

CHAPTER

FIVE

MSN SCORES

5.1 ScoreType.ISOTOPE_DISTRIBUTION

Gets the precursor for the scan, and calculates a mz window based on ppm tolerance for MS1 and +/- 1. This window is used to calculate the isotopic distribution (mzs). For all peaks in the scan, the intensities are accumulated in 2 different groups: isobIntensities for all the intensities of peaks that are within the isotopic distribution peaks, and nonIsobIntensities for the ones that are not in the distribution.

The score is calculated as:

```
score = isobIntensities/(isobIntensities + nonIsobIntensities)
```

5.2 ScoreType.REPORTERS_INTENSITY

Gets the higher and lower mz from all the labels and for every peak of the scan (MS2 or MS3), accumulates the intensities that are within this range (labels range).

Using the isobaric intensities of the labels, the score is calculated as:

```
score = isobIntensities/allIntensities
```

5.3 ScoreType.REPORTERS_FOUND

Counts how many of the isobaric labels are within the scan peaks (using MS2 or MS3 tolerance), and calculates the score as:

```
cnt/isoLabels.size()
```

5.4 Precursor signal percentage

Intensity of the precursor percentage from the total intensity in a given window (MS1). Gets the precursor intensity using the mz +/- MS1PPMTolerance. We limit the mz window to search for intensities by using the lowest precursor mz - ms1 window in Da and the highest + window in Da.

Accumulates all peaks intensities within the specified mz window of the MS1 scan.
Calculates the signal as:

```
signalPerc = precIntensity/allIntensities
```

5.5 Precursor weighted signal

Using the previous MS1 scan to the precursor's, the precursor's MS1, and the posterior one. We calculate the retention time of each scan and then the total distance between the scans as: Precursor's retention time + Distance between previous MS1 and precursor's + distance between precursor's and posterior MS1. For each of these 3 scans we calculate the precursor signal percentage so that the weighted score can be calculated as:

```
(dPivotCurrent / totalDistances) * currentSignal + (dPivotPre / totalDistances) *␣
↪preSignal + (dPivotPost / totalDistances) * postSignal
```

CHAPTER

SIX

MS2 SCORES

All MS2 scores are calculated by matching the BY fragment ions of the peptide to their most similar peak. (BY Fragment ions for the peptide, BY fragment ions for the peptide with neutral loss applied (calc all fragment ions for peptide mass - neutral mass loss [neutralLossesForFragmentIons]), and neutral losses specified [neutralLosses]) From the peak list we are excluding the MS1 precursor, the reporter ions, and the complement ions.

top X: top x peaks with highest intensity.

6.1 Peptide Intensity Score

From Scan peaks, sum of intensities of fragment ions divided by total.

```
score = totalFragmIonsIntensity/totalIntensity
```

Top X Peptides Peaks Ratio Top X defined by user Number of fragment ions in top X divided by top X topX-FIons/topXNum

6.2 Peptide Top X Intensity Score

Sum of intensities of fragment ions in top X divided by the total sum of intensities of top X.

```
score = topXFragmIonsIntensity/topXIntensity
```

6.3 Top X Intensity From Total Score

Sum of intensities in top X divided by the sum of all peaks' intensities.

```
score = topXIntensity/totalIntensity
```

6.4 Peptide Top X Intensity From Total Score

Sum of intensities of fragment ions in top X divided by the sum of all peaks' intensities.

```
score = topXFragmIonsIntensity/totalIntensity
```

6.5 Top Peak Intensity Score

Higher intensity divided by the sum of all peaks' intensities.

```
score = topPeakIntensity/totalIntensity
```

6.6 Top Peak Intensity Top X Score

Highest intensity divided by the total sum of intensities of top X.

```
score = topPeakIntensity/topXIntensity
```

6.7 Top Peak From Peptide

Indicates if the top peak (most intense) is a fragment ion

6.8 Top Peak From Peptide Neutral Loss

Indicates if the top peak (most intense) is a neutral loss

6.9 Top Peak Mass

Mass of the top peak (most intense)

6.10 msnTotalSignal

Sum of all the isobaric quant intensities (found labels)

6.11 precTotalSignal

Sum of precursor intensities found in MS2 (low and high mz is calculated using MS2PPMTolerance)

6.12 totalSignalSPSWind

Sum of precursors intensities found in MS2 within a window of mz, this window is calculated as follows:

```
mz +/- ms2PrecursorWindowTolerance(mz)
```

6.13 precTPIntRatio

```
score = precTotalSignal/totalSignalSPSWind
```

6.14 precRepIntRatio

```
matchingBYInt = matching B and Y fragment ions (from the SPS list) intensities
score = matchingBYInt / totalSignalSPSWind
```

6.15 precTPNumRatio

numPeaksSPSWind = number of peaks found in MS2 within a window of mz, this window is calculated as follows:

```
mz +/- ms2PrecursorWindowTolerance(mz)
score = number of precursors found / numPeaksSPSWind
```

CHAPTER

SEVEN

MS3 SCORES

7.1 Definitions

7.1.1 Precursors

List of precursors provided by the scan minus their parent scan precursors MS3 <- we are removing the precursors that are within the MS2 and MS1 MS2 <- we are removing the precursors that are within the MS1

7.1.2 Precursors intensities

We are obtaining the intensities of the MS3 precursors in the MS2 (peak selection using MS2PPMTolerance window)

7.1.3 Peptide Fragment ions

B and Y fragment ions for the peptide (search hit) matching MS2 peaks.

7.1.4 Precursors matching BY intensities

Using the fragment ions calculated for the peptide (search hit). Sum of all fragment ions intensities that their mz is found within any of the precursors windows (using `MS2PPMTolerance`)

7.1.5 Precursors window intensities

For every MS3 precursor we are obtaining all intensities within the window defined by
`lowMz = (Precursor mz - tolerance)` with `MS2PPMTolerance`
`highMz = (Precursor mz + tolerance)` with `MS2PPMTolerance`
Where tolerance is the conversion from `ms2PrecursorWindowTolerance` mass to `m/z` using the peptide charge (**search hit!**)

7.2 MSN Total signal

Sum of all label intensities found in the scan (level defined in the conf file) The peak is found within a window (PPM-Tolerance)

7.3 Precursor total signal

Sum of precursors intensities

7.4 Total signal SPS window

Sum of precursors window intensities

7.5 precTPIntRatio

Precursor total signal divided by total signal sps window

7.6 precRepIntRatio

Precursors matching BY intensities divided by total signal SPS window

7.7 precTPNumRatio

Number of MS3 precursors found in MS2 divided by number of peaks found in their windows.