Rapid development of proteomic applications with the AIBench framework

Hugo López-Fernández[1], Miguel Reboiro-Jato[1*], Daniel Glez-Peña[1], José R. Méndez Reboredo[1], Hugo M. Santos[2], Ricardo J. Carreira[2], José L. Capelo-Martínez[3], Florentino Fdez-Riverola[1]

[1]Escuela Superior de Ingeniería Informática, University of Vigo, Edificio Politécnico, Campus Universitario As Lagoas s/n, 32004 Ourense, Spain

[2]REQUIMTE, Departamento de Química, Facultade de Ciências e Tecnologia, Universidade Nova de Lisboa, Lisboa, Portugal

[3]Bioscope Group, Physical Chemistry Department, University of Vigo, Ourense, Spain

Summary

In this paper we present two case studies of Proteomics applications development using the AIBench framework, a Java desktop application framework mainly focused in scientific software development. The applications presented in this work are Decision Peptide-Driven, for rapid and accurate protein quantification, and Bacterial Identification, for Tuberculosis biomarker search and diagnosis. Both tools work with mass spectrometry data, specifically with MALDI-TOF spectra, minimizing the time required to process and analyze the experimental data.

1 Introduction and Motivation

Since the 1980s, computational methods have been used for comparative analysis of genome data [1], leading to the Bioinformatics field expansion, where various branches such as Genomics, Proteomics, Computer-Aided Drug Design, Bio Data Bases & Data Mining, Molecular Phylogenetics, Microarray Informatics or System Biology emerged [2]. With the arrival of high-throughput 'omics' data, a new dimension was added to data-driven comparative research [1], where manual analysis became unsuitable, requiring the use of computational methods. Nowadays, due to the proliferation of high-throughput techniques, the need for Bioinformatics tools is increasing, being necessary even in small studies.

In this context, the rapid development of successful feature-rich applications containing advanced functionalities in the field of biomedical and clinical research remains a major demand for smaller institutions due to the lack of both human and financial resources [3]. This fact gets worse if we consider the software development effort required to deliver highly specialized applications usually demanding sophisticated user interfaces. Moreover, developing applications in an interdisciplinary and applied research context also presents a large number of particular requisites ranging from computational requirements to usability. Specific issues include (*i*) sharing of heterogeneous data, (*ii*) integrating third party or previously developed

[*]To whom correspondence should be addressed. Email: mrjato@uvigo.es

algorithms, (*iii*) cross-platform compatibility, (*iv*) ability to repeat workflows but changing a few parameters or input data, (*v*) extensive use of logging messages to monitor the progress of long processes, (*vi*) establishing values for a high and variable number of parameters before running experiments and (*vii*) taking the maximum advantage of multi-threading capabilities in high-demanding tasks, among others.

Considering these requirements, we have developed AIBench [3], an open-source Java application framework for rapid development of translational software in Biomedicine. In this work, we describe the design of two new AIBench-based applications in the field of Proteomics: Decision Peptide-Driven and Bacterial Identification. Both tools analyze MALDI-TOF spectra in order to accurately quantify the amount of protein in a given sample and to discriminate between Tuberculosis strains, respectively.

The paper is structured as follows. Section 2 contains an overview of the AIBench architecture and its key design concepts. Section 3 identifies the scope of the applications developed and presents the shared components. Sections 4 and 5 describe the main functionalities and AIBench components of the Decision Peptide Driven and Bacterial Identification applications, respectively. Finally, Section 5 includes the conclusions and future work.

2 The AIBench Framework

The AIBench platform was particularly conceived to facilitate the development of a wide range of research applications based on general input-processing-output cycles where the framework acts as the glue between each executed task. In order to provide the basis for supporting rapid application development, AIBench manages three key concepts that are present in every AIBench application: operations, data-types and views. The developer only needs to concentrate on how to divide and structure the problem specific code into objects of these three entities.

From an architectonical perspective, AIBench is structured in several layers, as it is shown in Figure 1. The AIBench framework runs over a plug-in engine able to define a straightforward reusable component model where both the framework native components and the application-specific functionalities are divided and packaged into plug-ins. AIBench plug-ins are isolated by default, increasing the modularity and ensuring that accidental coupling is not introduced, but they can also interact by establishing dependencies or extension points. A dependency between plug-ins allows one plug-in to require other plug-ins to be present at runtime and to be allowed to access their classes and/or resources. An extension point declares a place where some plug-in can be extended by another plug-in (extension), usually providing a specific interface implementation.

The Core layer contains two native plug-ins: the Core and the Workbench. The AIBench Core detects and registers the application-specific operations, executes them upon request, keeps the results in the Clipboard structure and stores the session workflow in the History. The graphical user interface aspects are implemented in the Workbench plug-in, which creates the main application window, composes a menu bar with all the implemented operations, generates input dialogs when some operation is requested for execution, instantiates the registered results viewers, etc. All additional services bundled with AIBench belong to the Services layer and are also implemented via independent plug-ins that can be easily removed to meet the application

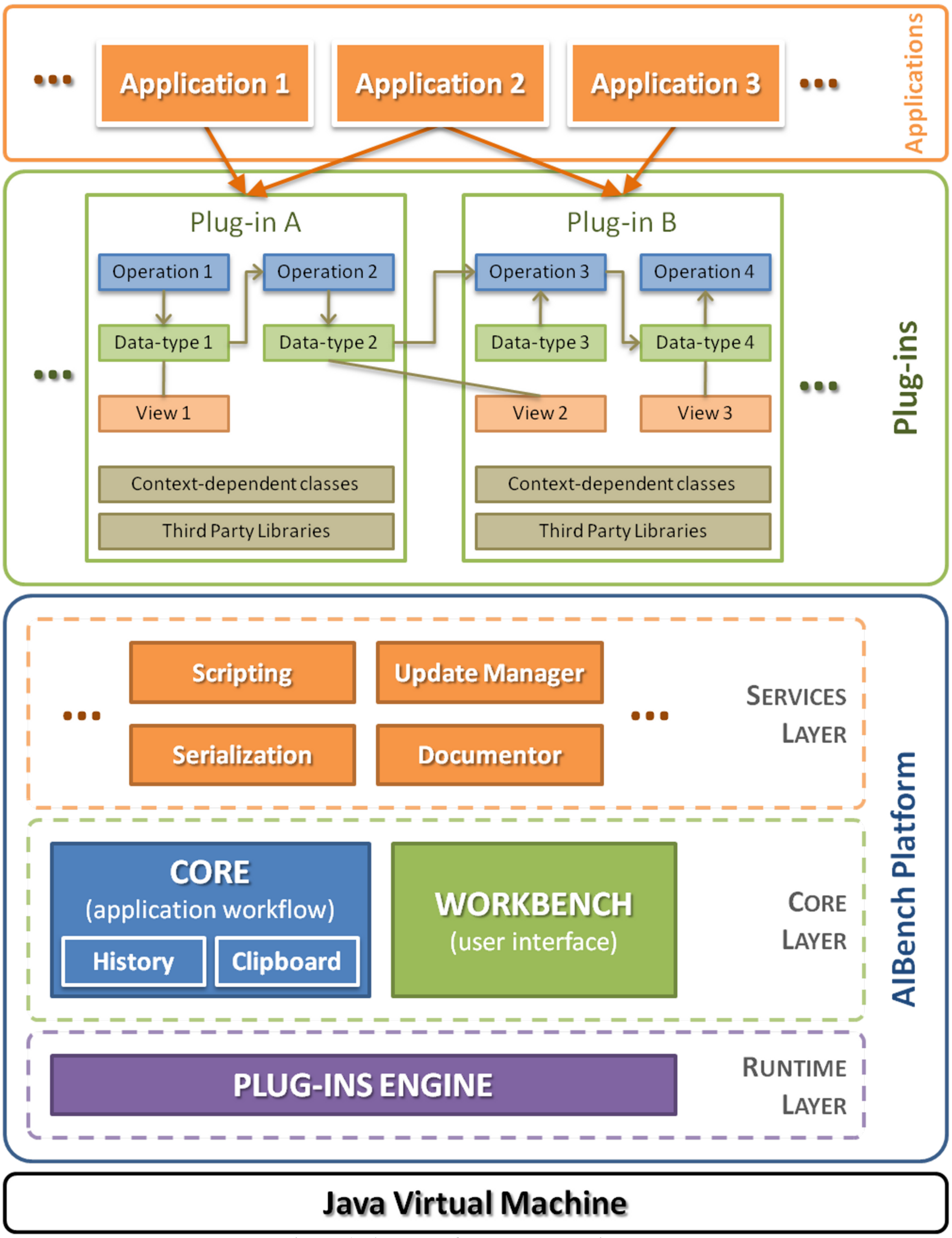


Figure 1: AIBench framework architecture.

specific needs. The Core and Services layers are maintained by the AIBench team and constitute all the code built-in and distributed with the framework, being the starting point of every development.

The Plugin and Application layers are placed on the top of the architecture and contains the application specific code (operations, data-types and views) provided by applications developers (AIBench users). In this sense, when an applications developer starts using the framework, there are no operations, data-types or views available, because these components are problem specific items. However, operations, data-types and views can (and should) be shared among applications related to the same area, especially when they are developed inside the same team. These higher-level components, along with other third-party libraries are also packaged in one or more plug-ins. Finally, from the most abstract point of view, an AIBench application can be seen as a collection of operations, data-types and views, reusable in more than one final application.

3 Building Proteomics Applications

Mass spectrometry (MS) is an analytical technique that generates mass spectra plotting the relative abundance of molecules versus their mass-to-charge ratio (m/z) [4]. This technique is an important tool in Proteomics for biomarker discovery, identification and quantification, as it allows the detection of small molecules (such as peptides) and the measurement of their intensities. Due to the low analysis time and the high data throughput of MS, it is very useful for fast and accurate diagnosing and monitoring different disease states [5]. On the other hand, the volume of data generated in MS experiments makes manual analysis unsuitable, requiring the use of Bioinformatic applications.

In this context, we have used the AIBench framework for the development of two MS data analysis applications with specific objectives, which will be presented in the following sections. These applications are intended to work with MALDI-TOF data, a soft ionization technique used in MS. As the data source for both applications is the same, a first plug-in named 'Peptide Analysis' has been developed containing common data-types, views and operations. Figure 2 shows the main classes of this plug-in.

The SpectrumData class represents a MS spectrum composed by a set of peaks each with mass-to-charge ratio and intensity. ExperimentalData contains a set of SpectrumData coming from the same sample or data analysis. These three classes are specializations of three more generic classes, which can be used to work with other types of experimental data. The LoadSpectrumData and LoadExperimentalData operations can be used to load spectra from CSV files and, although it is not represented in the Figure 2, the ExportToCSV operation is a generic operation to store different data-types into CSV files. Finally, the views in the Figure 2 present the main data-types as charts and tables.

The following sections describe the case studies of the two Proteomics applications developed with the AIBench framework and based on the 'Peptide Analysis' plug-in previously introduced.

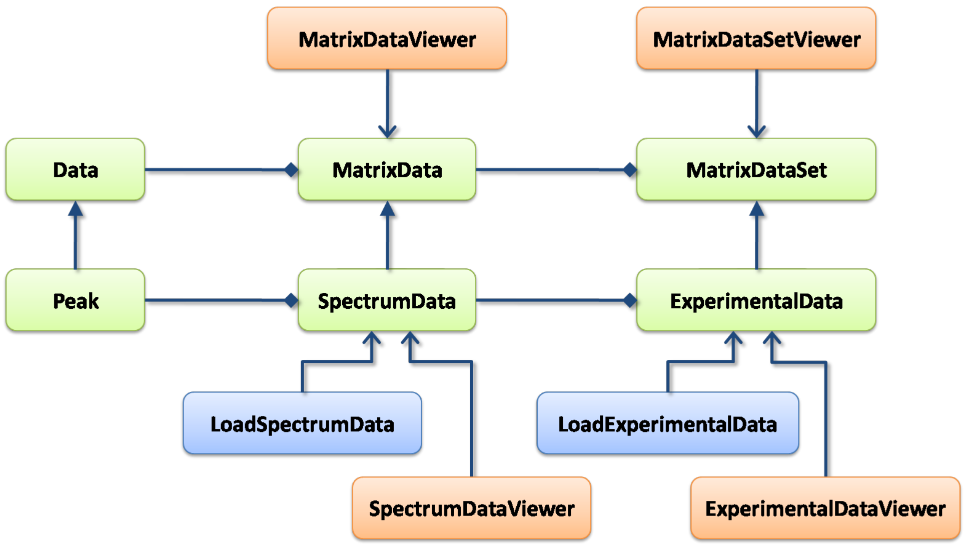


Figure 2: Main data-types (green), operations (blue) and views (orange) of the 'Peptide Analysis' plug-in.

4 Case Study 1: Decision Peptide-Driven

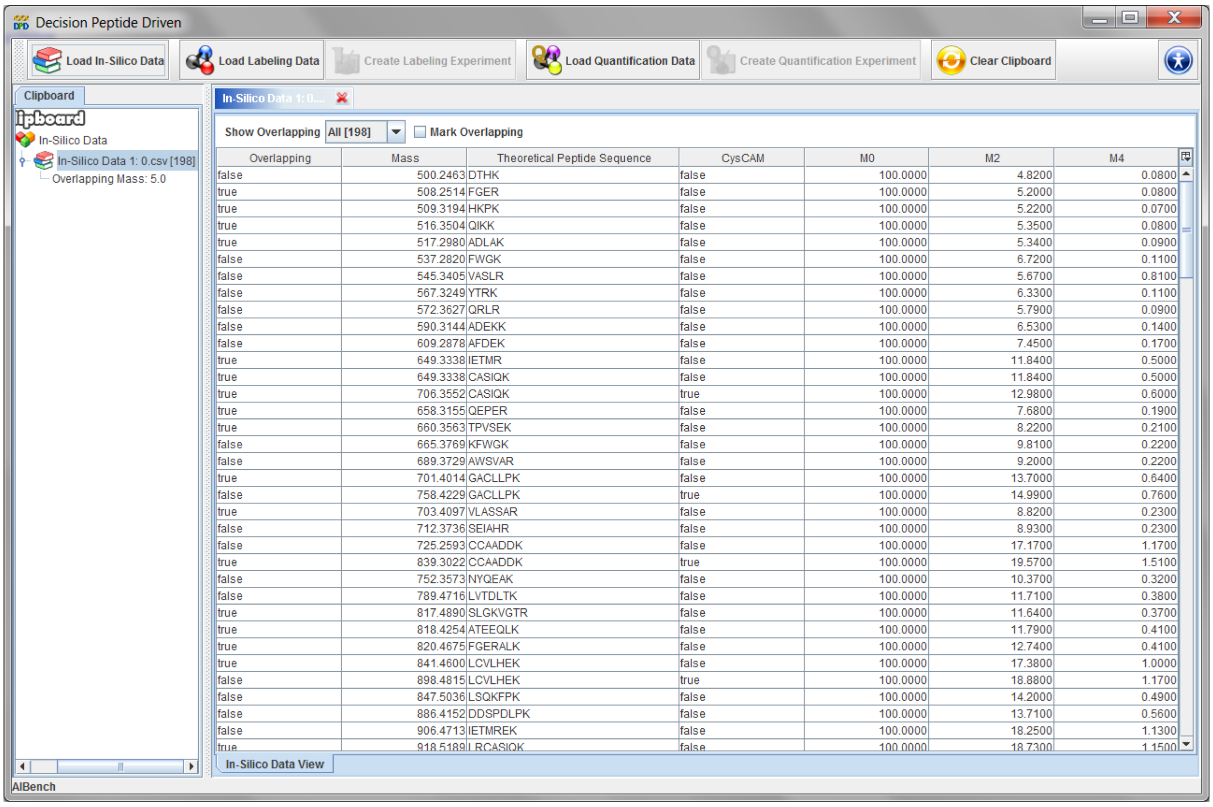
The identification and quantification of the protein content of biological samples plays a crucial role in biological and biomedical research [6]. The Decision Peptide-Driven tool[1] (DPD) [7] implements a software application for assisting the user in a protocol [8] for accurate protein quantification by using MALDI-TOF mass spectrometry. Main application views showing data and experiment information are shown in Figures 3 and 4, respectively.

Using the DPD software, user can compare the MALDI results of the direct and inverse ^{18}O-labeling experiments and quickly identify those peptides with paralleled loses in different sets of a typical proteomic workflow. Those peptides are used for subsequent accurate protein quantification. The interpretation of the MALDI data from direct and inverse labeling experiments is time-consuming requiring a significant amount of time to do all comparisons manually.

The DPD software shortens and simplifies the search for the peptides that must be used for quantification from a week to just some minutes. To do so, it takes as input several MALDI spectra and aids the researcher in an automatic mode (*i*) to compare data from direct and inverse ^{18}O-labeling experiments, calculating the corresponding ratios to determine those peptides with paralleled losses throughout different sets of experiments; and (*ii*) allow to use those peptides as internal standards for subsequent accurate protein quantification using ^{18}O-labeling.

When working with the DPD software, users can follow two main workflows: labeling and quantification. Figure 5 depicts the data-types (green) and operations (blue) involved in the labeling workflow and their relationships. Along with direct and reverse data, *in silico* data is needed to perform the peptide detection in the spectra. Once the data has been loaded, the reproducible peptides can be obtained by executing a labeling experiment. This is a macro-operation composed by a series of operations: (*i*) detect peptides, indentifies the peptides present in the direct and inverse data using the *in silico* data as reference, (*ii*) filter peptides, removes the low-quality peptides, (*iii*) intersec peptides, joins the direct and inverse peptides, and (*iv*) select reproducible, evaluates the quality of the final peptides. The final output of the labeling work-

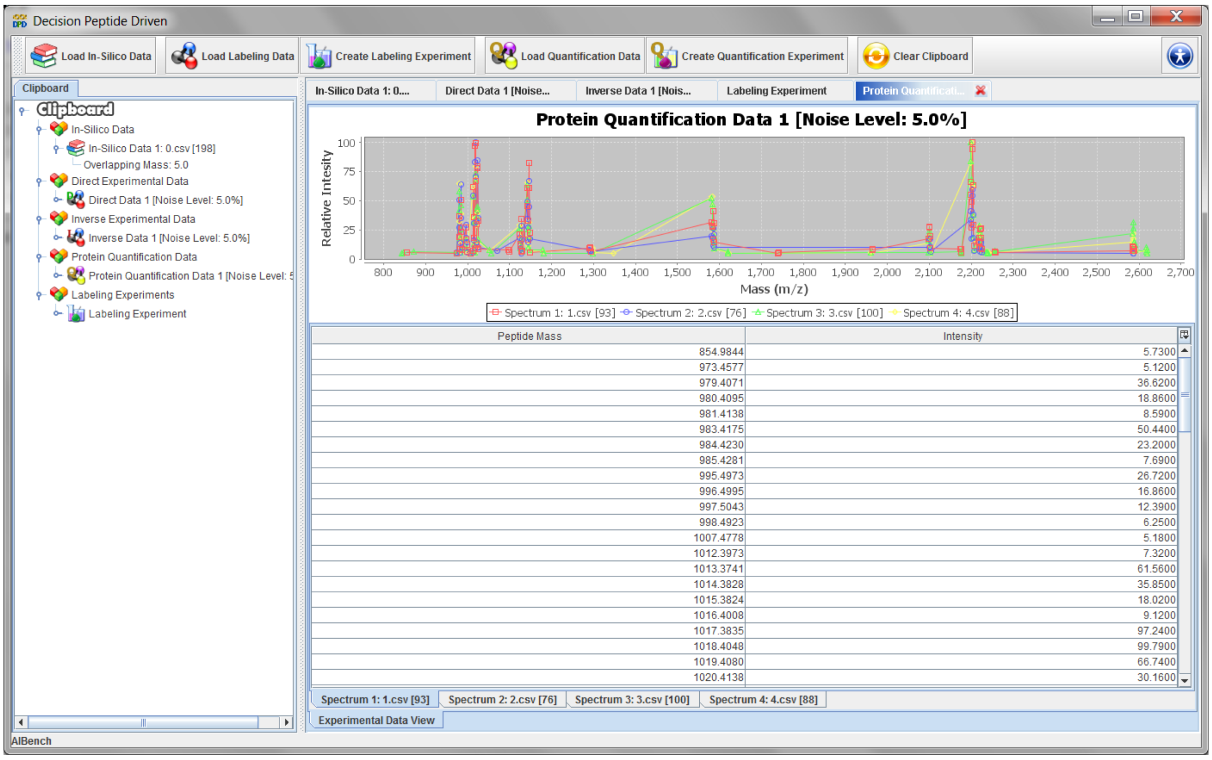
[1]http://sing.ei.uvigo.es/DPD



Show Overlapping All [198] Mark Overlapping

Overlapping	Mass	Theoretical Peptide Sequence	CysCAM	M0	M2	M4
false	500.2463	DTHK	false	100.0000	4.8200	0.0800
true	508.2514	FGER	false	100.0000	5.2000	0.0800
true	509.3194	HKPK	false	100.0000	5.2200	0.0700
true	516.3504	QIKK	false	100.0000	5.3500	0.0800
true	517.2980	ADLAK	false	100.0000	5.3400	0.0900
false	537.2820	FWGK	false	100.0000	6.7200	0.1100
false	545.3405	VASLR	false	100.0000	5.6700	0.8100
false	567.3249	YTRK	false	100.0000	6.3300	0.1100
false	572.3627	QRLR	false	100.0000	5.7900	0.0900
false	590.3144	ADEKK	false	100.0000	6.5300	0.1400
false	609.2878	AFDEK	false	100.0000	7.4500	0.1700
true	649.3338	IETMR	false	100.0000	11.8400	0.5000
true	649.3338	CASIQK	false	100.0000	11.8400	0.5000
true	706.3552	CASIQK	true	100.0000	12.9800	0.6000
true	658.3155	QEPER	false	100.0000	7.6800	0.1900
true	660.3563	TPVSEK	false	100.0000	8.2200	0.2100
false	665.3769	KFWGK	false	100.0000	9.8100	0.2200
false	689.3729	AWSVAR	false	100.0000	9.2000	0.2200
true	701.4014	GACLLPK	false	100.0000	13.7000	0.6400
false	758.4229	GACLLPK	true	100.0000	14.9900	0.7600
true	703.4097	VLASSAR	false	100.0000	8.8200	0.2300
false	712.3736	SEIAHR	false	100.0000	8.9300	0.2300
false	725.2593	CCAADDK	false	100.0000	17.1700	1.1700
true	839.3022	CCAADDK	true	100.0000	19.5700	1.5100
false	752.3573	NYQEAK	false	100.0000	10.3700	0.3200
false	789.4716	LVTDLTK	false	100.0000	11.7100	0.3800
true	817.4890	SLGKVGTR	false	100.0000	11.6400	0.3700
true	818.4254	ATEEQLK	false	100.0000	11.7900	0.4100
true	820.4675	FGERALK	false	100.0000	12.7400	0.4100
true	841.4600	LCVLHEK	false	100.0000	17.3800	1.0000
false	898.4815	LCVLHEK	true	100.0000	18.8800	1.1700
false	847.5036	LSQKFPK	false	100.0000	14.2000	0.4900
false	886.4152	DDSPDLPK	false	100.0000	13.7100	0.5600
false	906.4713	IETMREK	false	100.0000	18.2500	1.1300
true	918.5189	LRCASIQK	false	100.0000	18.7300	1.1500

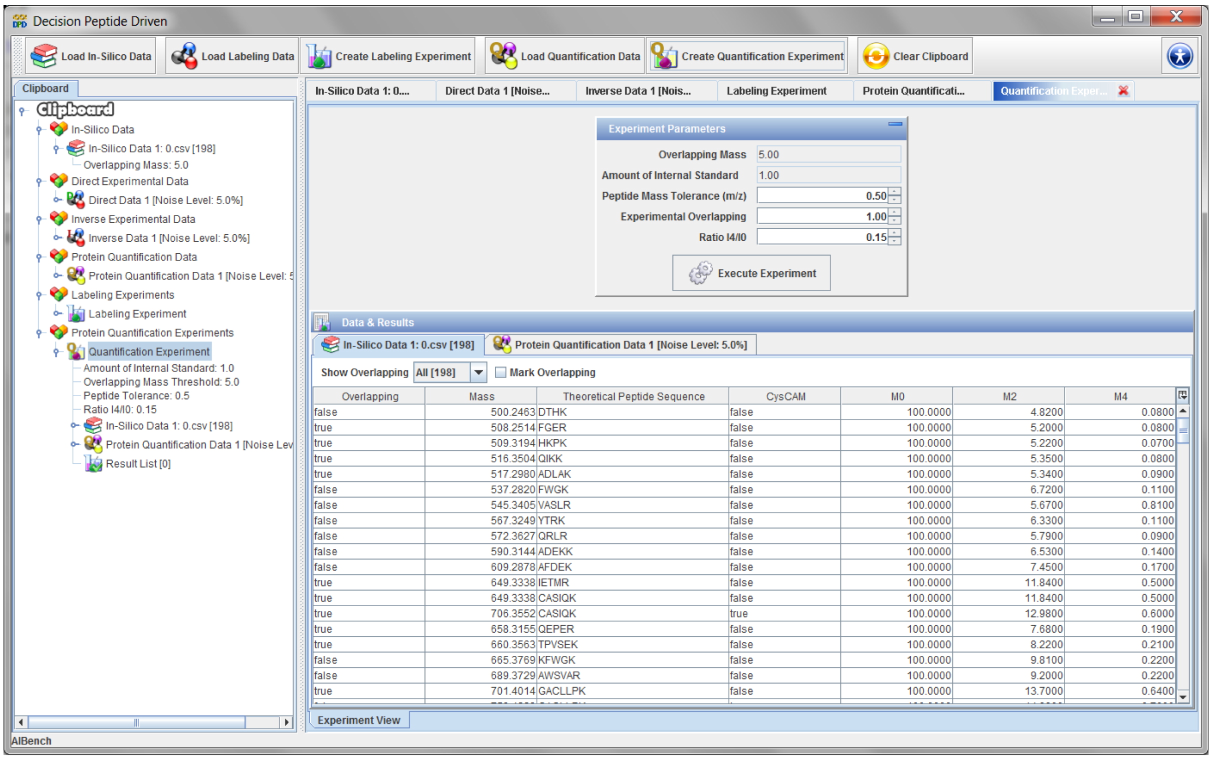
(a)



Peptide Mass	Intensity
854.9844	5.7300
973.4577	5.1200
979.4071	36.6200
980.4095	18.8600
981.4138	8.5900
983.4175	50.4400
984.4230	23.2000
985.4281	7.6900
995.4973	26.7200
996.4995	16.8600
997.5043	12.3900
998.4923	6.2500
1007.4778	5.1800
1012.3973	7.3200
1013.3741	61.5600
1014.3828	35.8500
1015.3824	18.0200
1016.4008	9.1200
1017.3835	97.2400
1018.4048	99.7900
1019.4080	66.7400
1020.4138	30.1600

(b)

Figure 3: Snapshots of (*a*) the *in silico* data view and (*b*) the experimental data view of the DPD software.


Decision Peptide Driven
Load In-Silico Data
Load Labeling Data
Create Labeling Experiment
Load Quantification Data
Create Quantification Experiment
Clear Clipboard
Clipboard
In-Silico Data
In-Silico Data 1: 0.csv [198]
Overlapping Mass: 5.0
Direct Experimental Data
Direct Data 1 [Noise Level: 5.0%]
Inverse Experimental Data
Inverse Data 1 [Noise Level: 5.0%]
Protein Quantification Data
Protein Quantification Data 1 [Noise Level: 5
Labeling Experiments
Labeling Experiment
Protein Quantification Experiments
Quantification Experiment
Amount of Internal Standard: 1.0
Overlapping Mass Threshold: 5.0
Peptide Tolerance: 0.5
Ratio I4/I0: 0.15
In-Silico Data 1: 0.csv [198]
Protein Quantification Data 1 [Noise Lev
Result List [0]
In-Silico Data 1: 0...
Direct Data 1 [Noise...
Inverse Data 1 [Nois...
Labeling Experiment
Protein Quantificati...
Quantification Exper...
Execute Experiment
Data & Results
In-Silico Data 1: 0.csv [198]
Protein Quantification Data 1 [Noise Level: 5.0%]
Experiment View
AIBench

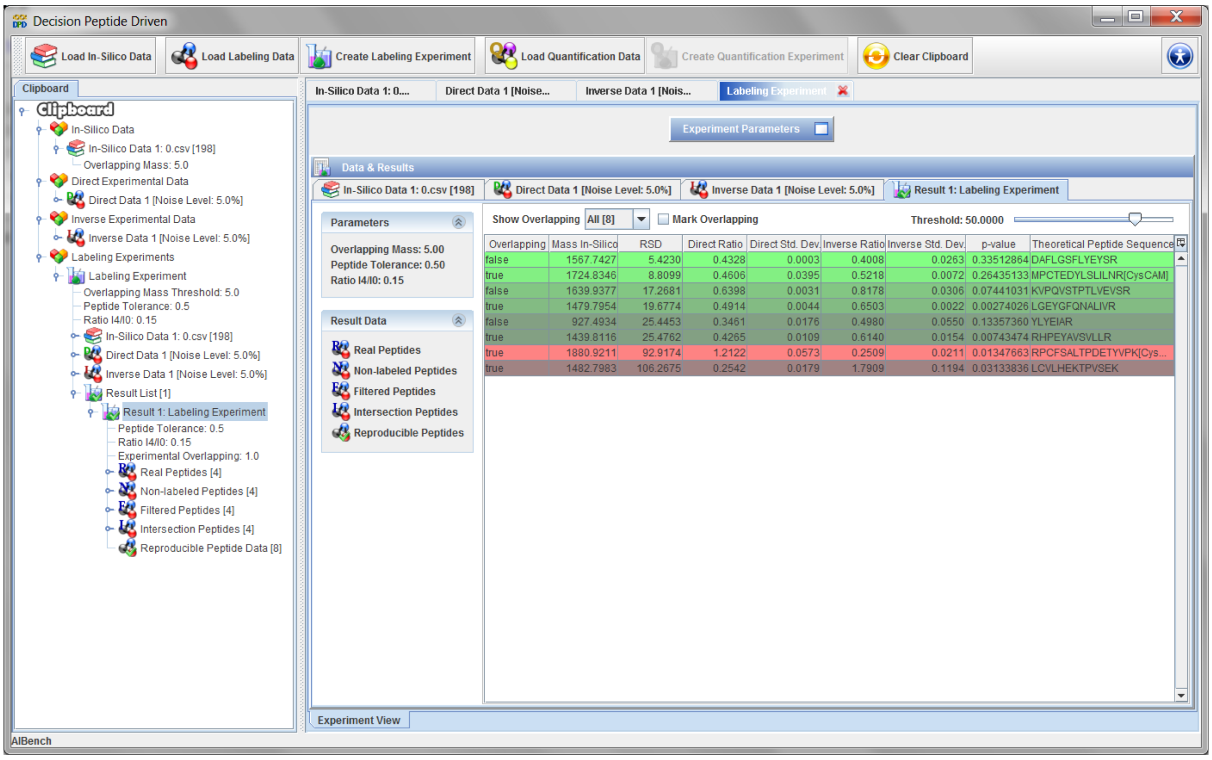
Experiment Parameters

Overlapping Mass	5.00
Amount of Internal Standard	1.00
Peptide Mass Tolerance (m/z)	0.50
Experimental Overlapping	1.00
Ratio I4/I0	0.15

Show Overlapping All [198] Mark Overlapping

Overlapping	Mass	Theoretical Peptide Sequence	CysCAM	M0	M2	M4
false	500.2463	DTHK	false	100.0000	4.8200	0.0800
true	508.2514	FGER	false	100.0000	5.2000	0.0800
true	509.3194	HKPK	false	100.0000	5.2200	0.0700
true	516.3504	QIKK	false	100.0000	5.3500	0.0800
true	517.2980	ADLAK	false	100.0000	5.3400	0.0900
false	537.2820	FWGK	false	100.0000	6.7200	0.1100
false	545.3405	VASLR	false	100.0000	5.6700	0.8100
false	567.3249	YTRK	false	100.0000	6.3300	0.1100
false	572.3627	QRLR	false	100.0000	5.7900	0.0900
false	590.3144	ADEKK	false	100.0000	6.5300	0.1400
false	609.2878	AFDEK	false	100.0000	7.4500	0.1700
true	649.3338	IETMR	false	100.0000	11.8400	0.5000
true	649.3338	CASIQK	false	100.0000	11.8400	0.5000
true	706.3552	CASIQK	true	100.0000	12.9800	0.6000
true	658.3155	QEPER	false	100.0000	7.6800	0.1900
true	660.3563	TPVSEK	false	100.0000	8.2200	0.2100
false	665.3769	KFWGK	false	100.0000	9.8100	0.2200
false	689.3729	AWSVAR	false	100.0000	9.2000	0.2200
true	701.4014	GACLLPK	false	100.0000	13.7000	0.6400

(a)



Parameters

Overlapping Mass: 5.00
Peptide Tolerance: 0.50
Ratio I4/I0: 0.15

Show Overlapping All [8] Mark Overlapping Threshold: 50.0000

Overlapping	Mass In-Silico	RSD	Direct Ratio	Direct Std. Dev.	Inverse Ratio	Inverse Std. Dev.	p-value	Theoretical Peptide Sequence
false	1567.7427	5.4230	0.4328	0.0003	0.4008	0.0263	0.33512864	DAFLGSFLYEYSR
true	1724.8346	8.8099	0.4606	0.0395	0.5218	0.0072	0.26435133	MPCTEDYLSLILNR[CysCAM]
false	1639.9377	17.2681	0.6398	0.0031	0.8178	0.0306	0.07441031	KVPQVSTPTLVEVSR
true	1479.7954	19.6774	0.4914	0.0044	0.6503	0.0022	0.00274026	LGEYGFQNALIVR
false	927.4934	25.4453	0.3461	0.0176	0.4980	0.0550	0.13357360	YLYEIAR
true	1439.8116	25.4762	0.4265	0.0109	0.6140	0.0154	0.00743474	RHPEYAVSVLLR
true	1880.9211	92.9174	1.2122	0.0573	0.2509	0.0211	0.01347663	RPCFSALTPDETYVPK[Cys...
true	1482.7983	106.2675	0.2542	0.0179	1.7909	0.1194	0.03133836	LCVLHEKTPVSEK

(b)

Figure 4: Snapshots of *(a)* the experiment view and *(b)* the experiment results view of the DPD software.

flow is a list of reproducible peptide with a quality value, along with the data generated in the intermediate steps of the experiment.

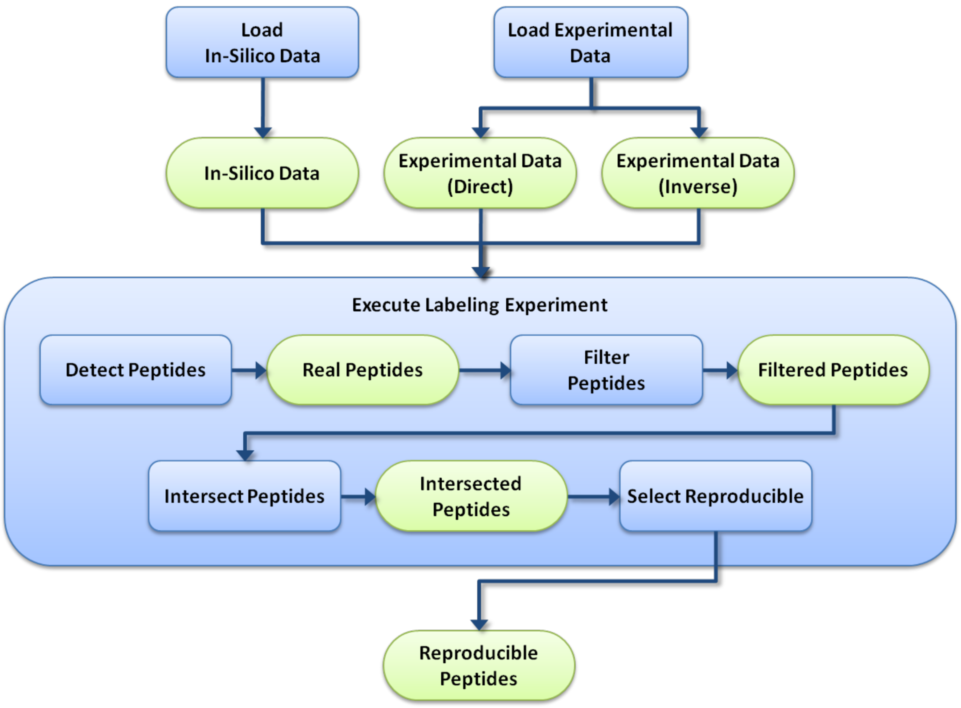


Figure 5: DPD labeling experiment workflow.

The quantification workflow allows the protein quantification in a MALDI-TOF sample. As can be seen in Figure 6, this workflow is very similar to the labeling workflow. The main difference is that, in this case, only one experimenal data is needed. Additionally, the last operation of the quantification experiment is a quantification step where the amount of protein is calculated.

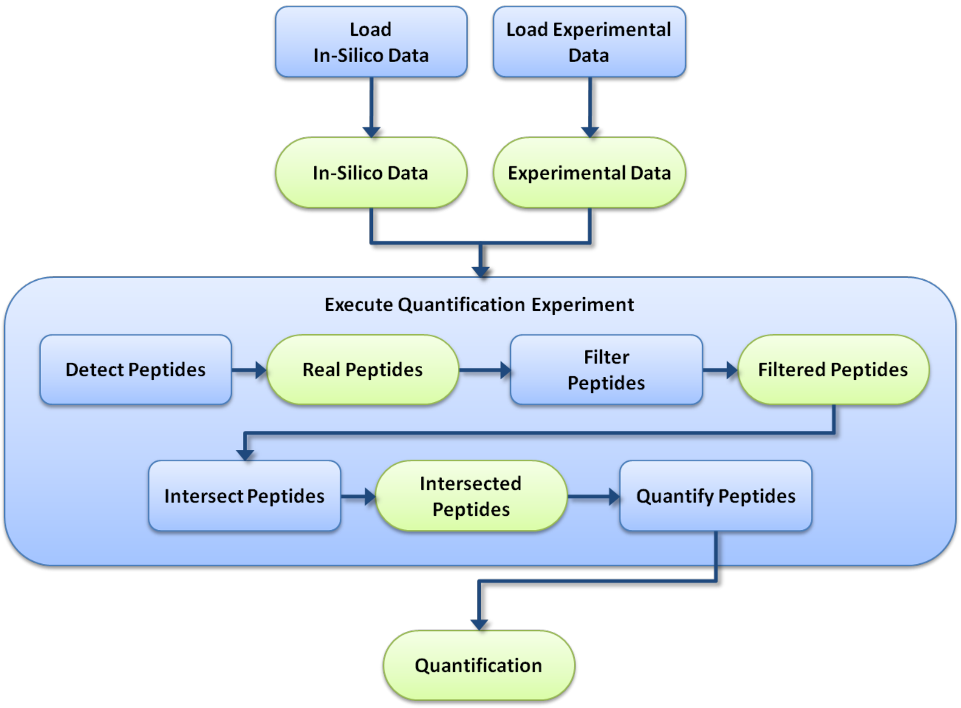


Figure 6: DPD quantification experiment workflow.

Both workflows described above are sequential workflows, as they implement two concrete protocols for MS data analysis specifically designed for the proposed problems. This type of protocols suits perfectly to the input-process-output model on which is based AIBench, thus we could take advantage of the framework capabilities in order to minimize the development time.

The AIBench components used to build the DPD application are distributed in two plug-ins: the 'Peptide Analysis' plug-in, described in the previous section, and the 'Decision Peptide-Driven' plug-in, containing all operations, data-types and views of the software. The second plug-in depends on the first one. The list of components included in the 'Decision Peptide-Driven' plug-in are summarized in Table 1.

Table 1: AIBench components of the DPD software.

Component	Description
Operations	
Create Labeling Experiment	Creates a new labeling experiment.
Create Quantification Experiment	Creates a new quantification experiment.
Execute Experiment	Executes a labeling or quantification experiment.
Export Result	Exports an experiment result as a log file.
Load In Silico Data	Loads *in silico* peptide information from a csv file.
Load Labeling Data	Loads experimental spectra data for labeling experiments from csv files.
Load Quantification Data	Loads experimental spectra data for quantification experiments from csv files.
Data-types	
InSilicoData	*In silico* peptide information.
DirectExperimentalData	Experimental spectra data to be used as direct data in labeling experiments.
InverseExperimentalData	Experimental spectra data to be used as inverse data in labeling experiments.
LabelingExperiment	A labeling experiment. Contains direct and inverse data and the parameter configuration.
QuantificationExperimentalData	Experimental spectra data to be used in quantification experiments.
QuantificationExperiment	A quantification experiment. Contains quantification data and the parameter configuration.
IExperiment	Common interface for labeling and quantification experiments.
IResult	Common interface for experiment results.
Storeable	Items that can be exported to log files.
Views	
Experiment View	Allows the user to execute experiments and view results.
Experimental Data View	Shows chart and table representation of the experimental spectra data.
In Silico Data View	Shows table representation of the *in silico* data.

5 Case Study 2: Bacterial Identification

Tuberculosis (TB) is an infectious disease and the world's leading cause of death. In fact, the World Health Organization estimates that one third of the world's population is infected with Mycobacterium tuberculosis (MTB) [9]. Moreover, the usual methodologies for TB diagnosis can take up to 3 months for the identification of the correct infectious agent or only 1 week with molecular methods like PCR, but these last methods are expensive and are not affordable for many laboratories, especially in undeveloped countries. Recently, Hettick *et al.* reported the discrimination of intact mycobacteria at the strain level using MALDI-TOF-MS and a bio-statistical analysis [10]. However, for security reasons the handling of intact TB, MDR-TB and XDR-TB can only be done in laboratories with a biosafety level of 3, and thus the analysis of intact MTB by mass spectrometry is not possible.

Bacterial Identification tool[2] (BI) is a software application that allows rapid bacterial identification through peptide mass mapping obtained after analysis by MALDI-MS-based approaches.

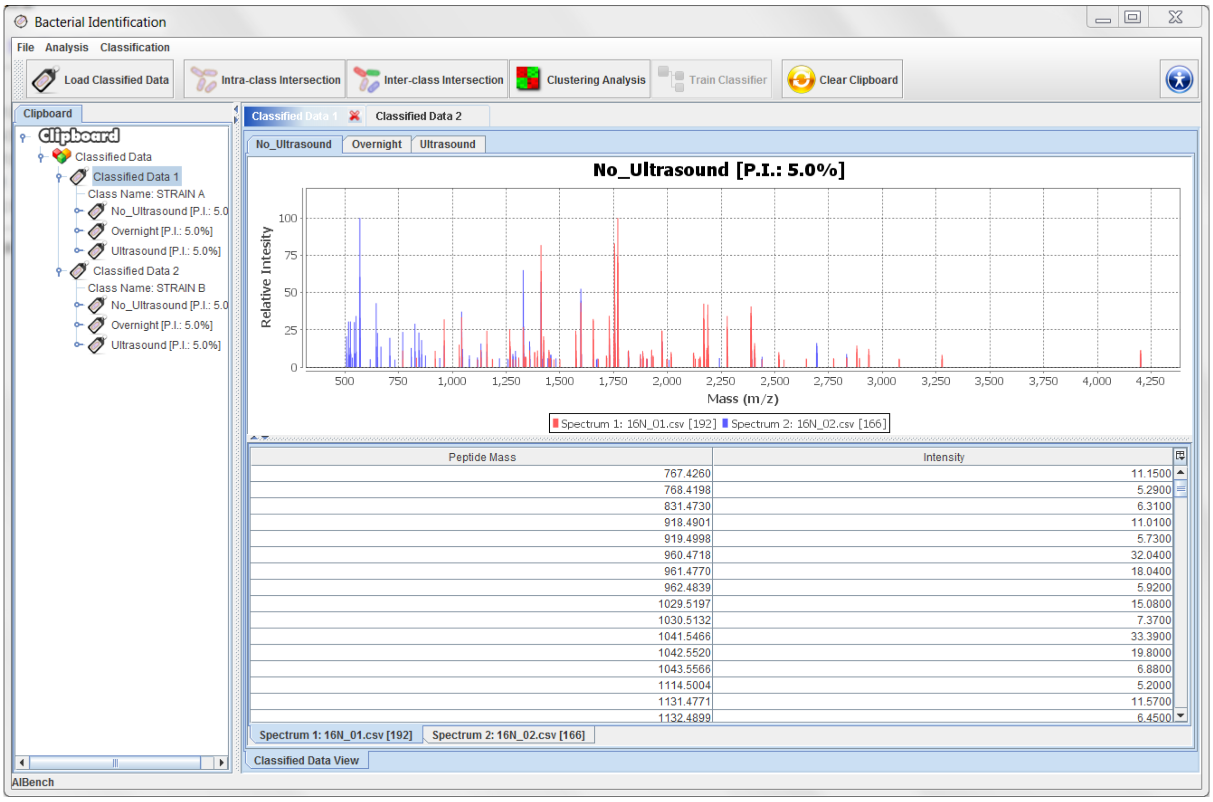
The main functionalities of BI tool allow the researcher to (*i*) load and visualize spectra data containing peaks of each bacterial strain, (*ii*) identify a peptide fingerprint of each strain, containing all common peaks of all replicates, (*iii*) fingerprint comparison to obtain peaks which are unique of a given strain, being potential biomarkers, (*iv*) train and test a Support Vector Machines (SVM) classifier using the most predictive peptides and (*v*) perform and visualize a hierarchical clustering of the strains with those predictive peptides. Main views of the application are shown in Figures 7 and 8.

Figure 9 shows the internal workflow of the BI tool. Besides the data loading, this workflow can be divided into three sub-workflows, according to the functionalities they cover. The 'Strains Analysis' sub-workflow includes the 'intra' and 'inter' class analysis, useful for identifying the peptide fingerprint of TB strains and the discriminant peaks among different strains. The 'Classification' sub-workflow allows the user to train a SVM classifier using TB data previously classified and the correspondent list of discriminant masses. Once the classifier is trained, it can be used to classify new TB data. Finally, in the 'Clustering' sub-workflow, user can perform a hierarchical clustering, creating a heat map to study the relationship between the analyzed strains and, additionally, between their peptides.

In this case, we do not have a sequential workflow but, as noted above, the main workflow is divided into three different sub-workflows according to the analysis to perform. As the data loaded can be used in different analysis, the AIBench capability to keep data on the Clipboard so you can reuse it in different operations was very useful when developing this application.

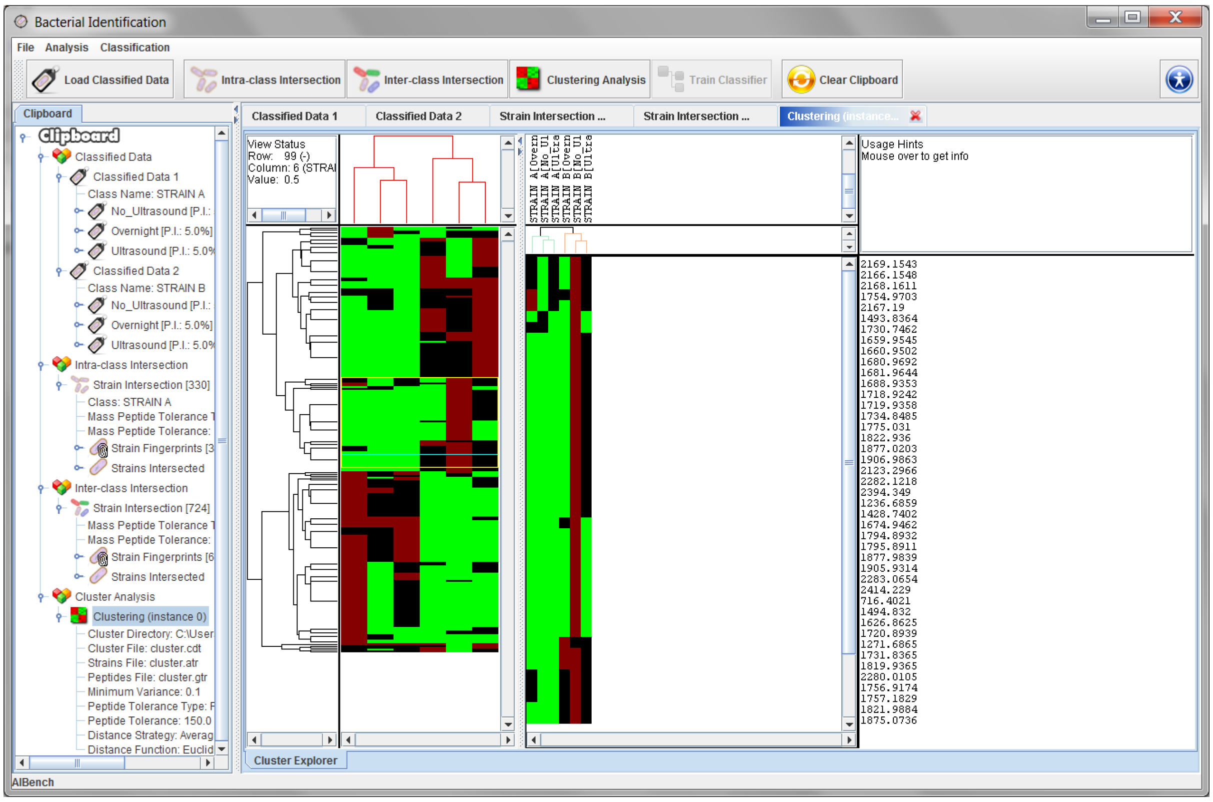
The AIBench components used to build the BI application are distributed in two plug-ins: the 'Peptide Analysis' plug-in and the 'Bacterial Identification' plug-in, containing all operations, data-types and views of the software. As in the previous case, the second plug-in depends on the first one. The list of components included in the 'Bacterial Identification' plug-in are summarized in Table 2.

[2]http://sing.ei.uvigo.es/BI



Peptide Mass	Intensity
767.4260	11.1500
768.4198	5.2900
831.4730	6.3100
918.4901	11.0100
919.4998	5.7300
960.4718	32.0400
961.4770	18.0400
962.4839	5.9200
1029.5197	15.0800
1030.5132	7.3700
1041.5466	33.3900
1042.5520	19.8000
1043.5566	6.8800
1114.5004	5.2000
1131.4771	11.5700
1132.4899	6.4500

(a)



Cluster Directory: C:\User
Cluster File: cluster.cdt
Strains File: cluster.atr
Peptides File: cluster.gtr
Minimum Variance: 0.1
Peptide Tolerance Type: F
Peptide Tolerance: 150.0
Distance Strategy: Averag
Distance Function: Euclid

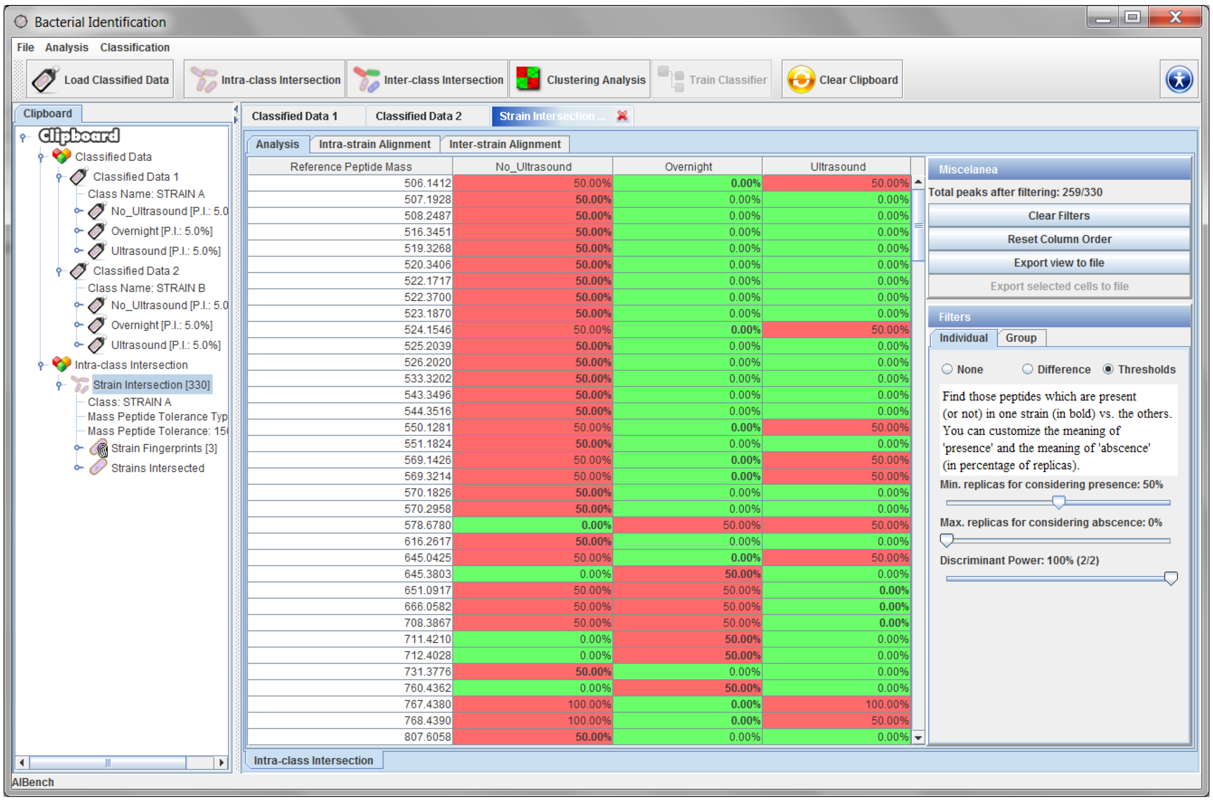
View Status
Row: 99 (-)
Column: 6 (STRAI
Value: 0.5

Usage Hints
Mouse over to get info

```
2169.1543
2166.1548
2168.1611
1754.9703
2167.19
1493.8364
1730.7462
1659.9545
1660.9502
1680.9692
1681.9644
1688.9353
1718.9242
1719.9358
1734.8485
1775.031
1822.936
1877.0203
1906.9863
2123.2966
2282.1218
2394.349
1236.6859
1428.7402
1674.9462
1794.8932
1795.8911
1877.9839
1905.9314
2283.0654
2414.229
716.4021
1494.832
1626.8625
1720.8939
1271.6865
1731.8365
1819.9365
2280.0105
1756.9174
1757.1829
1821.9884
1875.0736
```

(b)

Figure 7: Snapshots of (*a*) the data view and (*b*) the clustering view of the BI Software.



Clipboard
- Classified Data
  - Classified Data 1
    - Class Name: STRAIN A
    - No_Ultrasound [P.I.: 5.0
    - Overnight [P.I.: 5.0%]
    - Ultrasound [P.I.: 5.0%]
  - Classified Data 2
    - Class Name: STRAIN B
    - No_Ultrasound [P.I.: 5.0
    - Overnight [P.I.: 5.0%]
    - Ultrasound [P.I.: 5.0%]
- Intra-class Intersection
  - Strain Intersection [330]
    - Class: STRAIN A
    - Mass Peptide Tolerance Typ
    - Mass Peptide Tolerance: 15
    - Strain Fingerprints [3]
    - Strains Intersected

Reference Peptide Mass	No_Ultrasound	Overnight	Ultrasound
506.1412	50.00%	**0.00%**	50.00%
507.1928	**50.00%**	0.00%	0.00%
508.2487	**50.00%**	0.00%	0.00%
516.3451	**50.00%**	0.00%	0.00%
519.3268	**50.00%**	0.00%	0.00%
520.3406	**50.00%**	0.00%	0.00%
522.1717	**50.00%**	0.00%	0.00%
522.3700	**50.00%**	0.00%	0.00%
523.1870	**50.00%**	0.00%	0.00%
524.1546	50.00%	**0.00%**	50.00%
525.2039	**50.00%**	0.00%	0.00%
526.2020	**50.00%**	0.00%	0.00%
533.3202	**50.00%**	0.00%	0.00%
543.3496	**50.00%**	0.00%	0.00%
544.3516	**50.00%**	0.00%	0.00%
550.1281	50.00%	**0.00%**	50.00%
551.1824	**50.00%**	0.00%	0.00%
569.1426	50.00%	**0.00%**	50.00%
569.3214	50.00%	**0.00%**	50.00%
570.1826	**50.00%**	0.00%	0.00%
570.2958	**50.00%**	0.00%	0.00%
578.6780	**0.00%**	50.00%	50.00%
616.2617	**50.00%**	0.00%	0.00%
645.0425	50.00%	**0.00%**	50.00%
645.3803	0.00%	**50.00%**	0.00%
651.0917	50.00%	50.00%	**0.00%**
666.0582	50.00%	50.00%	**0.00%**
708.3867	50.00%	50.00%	**0.00%**
711.4210	0.00%	**50.00%**	0.00%
712.4028	0.00%	**50.00%**	0.00%
731.3776	**50.00%**	0.00%	0.00%
760.4362	0.00%	**50.00%**	0.00%
767.4380	100.00%	**0.00%**	100.00%
768.4390	100.00%	**0.00%**	50.00%
807.6058	**50.00%**	0.00%	0.00%

Miscelanea

Total peaks after filtering: 259/330

Clear Filters

Reset Column Order

Export view to file

Export selected cells to file

Filters

Individual Group

None Difference [✓ Thresholds]

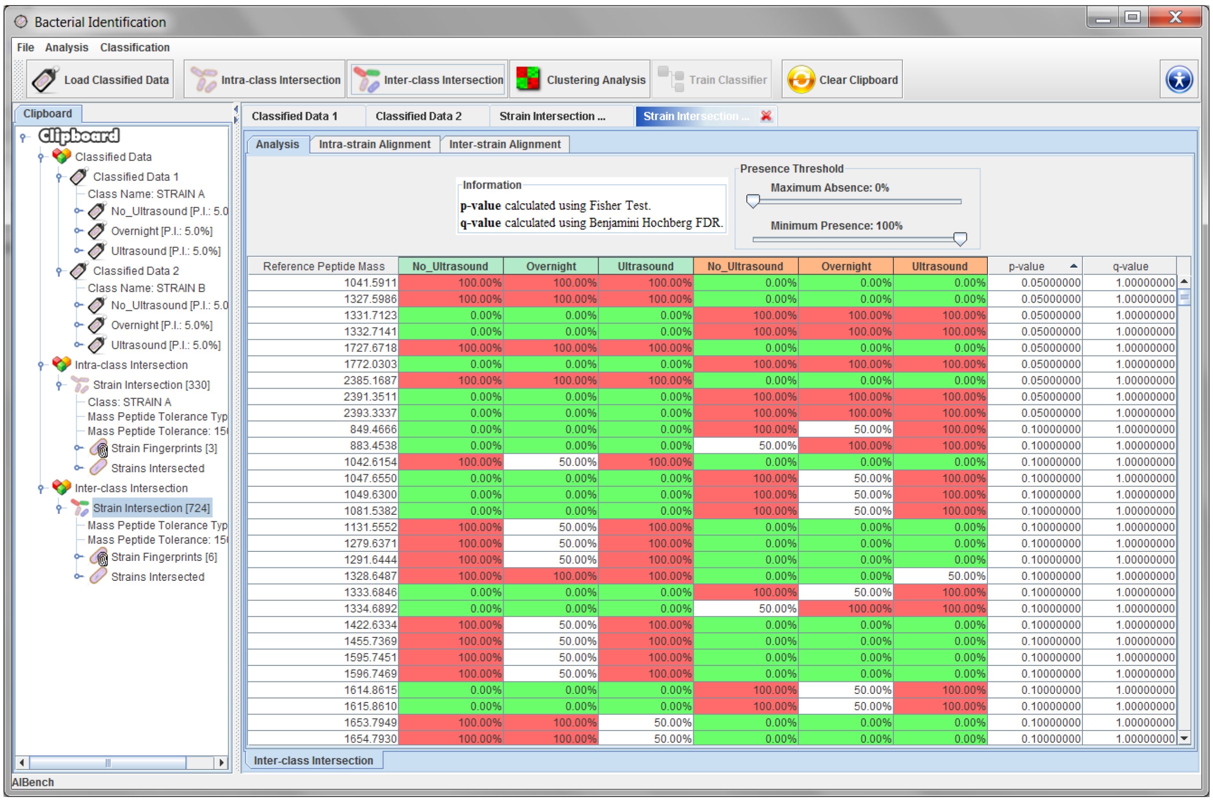
Find those peptides which are present (or not) in one strain (in bold) vs. the others. You can customize the meaning of 'presence' and the meaning of 'abscence' (in percentage of replicas).

Min. replicas for considering presence: 50%

Max. replicas for considering abscence: 0%

Discriminant Power: 100% (2/2)

(a)



Clipboard
- Classified Data
  - Classified Data 1
    - Class Name: STRAIN A
    - No_Ultrasound [P.I.: 5.0
    - Overnight [P.I.: 5.0%]
    - Ultrasound [P.I.: 5.0%]
  - Classified Data 2
    - Class Name: STRAIN B
    - No_Ultrasound [P.I.: 5.0
    - Overnight [P.I.: 5.0%]
    - Ultrasound [P.I.: 5.0%]
- Intra-class Intersection
  - Strain Intersection [330]
    - Class: STRAIN A
    - Mass Peptide Tolerance Typ
    - Mass Peptide Tolerance: 15
    - Strain Fingerprints [3]
    - Strains Intersected
- Inter-class Intersection
  - Strain Intersection [724]
    - Mass Peptide Tolerance Typ
    - Mass Peptide Tolerance: 15
    - Strain Fingerprints [6]
    - Strains Intersected

Information

p-value calculated using Fisher Test.
q-value calculated using Benjamini Hochberg FDR.

Presence Threshold

Maximum Absence: 0%

Minimum Presence: 100%

Reference Peptide Mass	No_Ultrasound	Overnight	Ultrasound	No_Ultrasound	Overnight	Ultrasound	p-value	q-value
1041.5911	100.00%	100.00%	100.00%	0.00%	0.00%	0.00%	0.05000000	1.00000000
1327.5986	100.00%	100.00%	100.00%	0.00%	0.00%	0.00%	0.05000000	1.00000000
1331.7123	0.00%	0.00%	0.00%	100.00%	100.00%	100.00%	0.05000000	1.00000000
1332.7141	0.00%	0.00%	0.00%	100.00%	100.00%	100.00%	0.05000000	1.00000000
1727.6718	100.00%	100.00%	100.00%	0.00%	0.00%	0.00%	0.05000000	1.00000000
1772.0303	0.00%	0.00%	0.00%	100.00%	100.00%	100.00%	0.05000000	1.00000000
2385.1687	100.00%	100.00%	100.00%	0.00%	0.00%	0.00%	0.05000000	1.00000000
2391.3511	0.00%	0.00%	0.00%	100.00%	100.00%	100.00%	0.05000000	1.00000000
2393.3337	0.00%	0.00%	0.00%	100.00%	100.00%	100.00%	0.05000000	1.00000000
849.4666	0.00%	0.00%	0.00%	100.00%	50.00%	100.00%	0.10000000	1.00000000
883.4538	0.00%	0.00%	0.00%	50.00%	100.00%	100.00%	0.10000000	1.00000000
1042.6154	100.00%	50.00%	100.00%	0.00%	0.00%	0.00%	0.10000000	1.00000000
1047.6550	0.00%	0.00%	0.00%	100.00%	50.00%	100.00%	0.10000000	1.00000000
1049.6300	0.00%	0.00%	0.00%	100.00%	50.00%	100.00%	0.10000000	1.00000000
1081.5382	0.00%	0.00%	0.00%	100.00%	50.00%	100.00%	0.10000000	1.00000000
1131.5552	100.00%	50.00%	100.00%	0.00%	0.00%	0.00%	0.10000000	1.00000000
1279.6371	100.00%	50.00%	100.00%	0.00%	0.00%	0.00%	0.10000000	1.00000000
1291.6444	100.00%	50.00%	100.00%	0.00%	0.00%	0.00%	0.10000000	1.00000000
1328.6487	100.00%	100.00%	100.00%	0.00%	0.00%	50.00%	0.10000000	1.00000000
1333.6846	0.00%	0.00%	0.00%	100.00%	50.00%	100.00%	0.10000000	1.00000000
1334.6892	0.00%	0.00%	0.00%	50.00%	100.00%	100.00%	0.10000000	1.00000000
1422.6334	100.00%	50.00%	100.00%	0.00%	0.00%	0.00%	0.10000000	1.00000000
1455.7369	100.00%	50.00%	100.00%	0.00%	0.00%	0.00%	0.10000000	1.00000000
1595.7451	100.00%	50.00%	100.00%	0.00%	0.00%	0.00%	0.10000000	1.00000000
1596.7469	100.00%	50.00%	100.00%	0.00%	0.00%	0.00%	0.10000000	1.00000000
1614.8615	0.00%	0.00%	0.00%	100.00%	50.00%	100.00%	0.10000000	1.00000000
1615.8610	0.00%	0.00%	0.00%	100.00%	50.00%	100.00%	0.10000000	1.00000000
1653.7949	100.00%	100.00%	50.00%	0.00%	0.00%	0.00%	0.10000000	1.00000000
1654.7930	100.00%	100.00%	50.00%	0.00%	0.00%	0.00%	0.10000000	1.00000000

(b)

Figure 8: Snapshots of (*a*) the intra-class analysis view and (*b*) the inter-class analysis view of the BI Software.

Table 2: AIBench components of the Bacterial Identification software.

Component	Description
Operations	
Clustering Analysis	Performs a hierarchical clusternig analysis.
Create Discriminant Peptide List	Creates a new discriminant peptide list.
Inter-class Intersection	Intersect several strain data with different class and generates a peptide occurrence matrix.
Intersect Discriminant Peptide Lists	Intersects two or more discriminant peptide lists.
Intra-class Intersection	Intersects several strain data with the same class and generates a peptide occurrence matrix.
Load Classified Data	Loads several strains data of the same class. Strains data must be in separated directories.
Load Clustering	Loads clustering analysis results.
Load Discriminant Peptide List	Loads a stored list of discriminant peptides.
Load Trained Classifier	Loads a trained classifier.
Merge Discriminant Peptide Lists	Merges two or more discriminant peptide lists.
Save Discriminant Peptide List	Stores a list of discriminant peptides.
Save Trained Classifier	Stores a trained classifier.
Single Inter-class Intersection	Intersects several strain data with different class and generates a peptide occurrence matrix.
Single Intra-class Intersection	Intersects several strain data with the same class and generates a peptide occurrence matrix.
Substract Discriminant Peptide Lists	Subtract two or more discriminant peptide lists to a reference peptide list.
Train Classifier	Trains a new SVM-based strain classifier.
Data-types	
ClassifiedData	Strain data associated with a class.
Clustering	Clustering data.
DiscriminantPeptideList	List of discriminant peptides.
InterClassIntersection	Peptide occurrence matrix coming from an inter–class intersection.
IntraClassIntersection	Peptide occurrence matrix coming from an intra–class intersection.
TrainedClassifier	Trained strain classifier.
PeptideFingerprints	List of characteristic peptides of a class.
Views	
Classified Data View	Shows chart and table representation of the strains data.
Cluster Explorer	Shows a heat map and hierarchical view of clustering data.
Data Classification	Allows the user to train a classifier and to classify data with it.
Inter-class Intersection	Allows the user to analyze a peptide occurrence matrix coming from an inter-class intersection.
Intra-class Intersection	Allows the user to analyze a peptide occurrence matrix coming from an inter-class intersection.

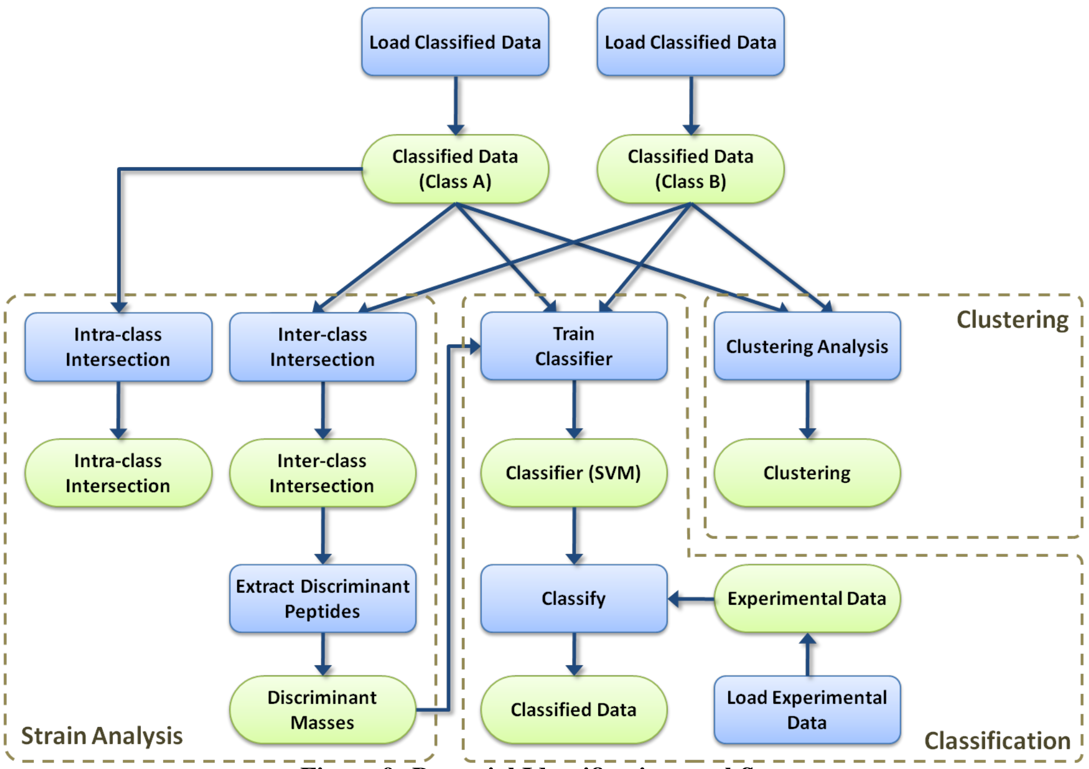


Figure 9: Bacterial Identification workflow.

6 Conclusions

This paper has presented two new applications, Decision Peptide-Driven and Bacterial Identification, in the field of Proteomics, showing the suitability of our AIBench framework to fast-develop user friendly applications in Biomedicine. The flexibility of AIBench has become evident as it was easily adapted to the different workflows of both applications. Additionally, both applications share components via common plug-ins, one of the most powerful aspects of AIBench.

The future work is divided in two objectives. By one hand, the two presented applications will be improved in their future versions. Decision Peptide-Driven will be integrated with Swissprot in order to download *in silico* data from the desired proteins and Bacterial Identification will include more classifier algorithms. In the other hand, the AIBench framework is in continuous development, including new features and bug fixes frequently.

Acknowledgements

This work is supported in part by the project 'MEDICAL-BENCH: Platform for the development and integration of knowledge-based data mining techniques and their application to the clinical domain' (TIN2009-14057-C03-02) from Ministerio de Ciencia e Innovación (Spain). M. Reboiro-Jato was supported by a pre-doctoral fellowship from the University of Vigo.

References

[1] P. Hogeweg. The roots of bioinformatics in theoretical biology. PLoS Computational Biology, 7(3):e1002021. 2011.

[2] A.S. Nair. Computational biology & bioinformatics: a gentle overview. Communications of the Computer Society of India. 2007.

[3] D. Glez-Peña, M. Reboiro-Jato, P. Maia, M. Rocha, F. Díaz and F. Fdez-Riverola. A rapid application development framework for translational research in biomedicine. Computer Methods and Programs in Biomedicine, 98(2):191-203, 2010.

[4] O.D. Sparkman. Mass spectrometry desk reference. Global View Publishing. 2000.

[5] P. Findeisen and M. Neumaier. Mass spectrometry based proteomics profiling as diagnostic tool in oncology: current status and future perspective. Clinical Chemistry and Laboratory Medicine, 47(6):666-684, 2009.

[6] X.J. Li, E.C. Yi, C.J. Kemp, H. Zhang and R. Aebersold. A software suite for the generation and comparison of peptide arrays from sets of data collected by liquid chromatography-mass spectrometry. Molecular & Cell Proteomics, 4(9):1328-1340, 2005.

[7] H.M. Santos, M. Reboiro-Jato, D. Glez-Peña, J.D. Nunes-Miranda, F. Fdez-Riverola, R. Carvallo and J.L. Capelo. Decision Peptide-Driven: a free software tool for accurate protein quantification using gel electrophoresis and matrix assisted laser desorption ionization time of flight mass spectrometry. Talanta, 82(4):1412-1420, 2010.

[8] H.M. Santos, D. Glez-Peña, M. Reboiro-Jato, F, Fdez-Riverola, M.S. Diniz, C. Lodeiro and J.L. Capelo-Martínez. A novel 18O inverse labeling-based workflow for accurate bottom-up mass spectrometry quantification of proteins separated by gel electrophoresis. Electrophoresis, 31:3407-3419, 2010.

[9] D. Kirschner. Timebomb: the global epidemic of multidrug resistant tuberculosis. Nature Medicine, 7:1173-1174, 2001.

[10] J.M. Hettick, M.L. Kashon, J.E. Slaven, Y. Ma, J.P. Simpson, P.D. Siegel, G.N. Mazurek and D.N. Weissman. Discrimination of intact mycobacteria at the strain level: a combined MALDI-TOF MS and biostatistical analysis. Proteomics, 6(24):6416-6425, 2006.