SugarArray: A User-centred-designed Platform for the Analysis of Lectin and Glycan Microarrays

Aurora Sucre[1,2][a], Raquel Pazos[3], Niels-Christian Reichardt[3,4] and Alba Garín-Muga[1,2][b]
[1]*Biodonostia, eHealth Group, Donostia-San Sebastián 20014, Spain*
[2]*Vicomtech, eHealth and Biomedical Applications, Donostia-San Sebastian 20014, Spain*
[3]*CIC biomaGUNE, Glycotechnology Laboratory, Paseo Miramón 182, 20014, San Sebastián, Spain*
[4]*CIBER-BBN, Paseo Miramón 182, 20014, San Sebastian, Spain*

Keywords: Glycomics, Lectin Microarray, Glycan Microarray, Microarray Data Analysis, Data Visualization.

Abstract: Glycan and lectin microarrays are two arising technologies, very important to the glycomics field. Glycomics is the science that focuses on defining the structures and functions of carbohydrates in nature. These microarrays provide information regarding the interactions between specific carbohydrates and proteins, and it has many applications in clinical and research settings. Nevertheless, the availability of analytical software for these types of arrays is very limited, so researchers usually perform data processing and analytical pipelines manually, which is very time consuming and prone to error. SugarArray was born as a user-friendly and intuitive stand-alone solution that process the intensity data generated from glycan or lectin array studies, and displays the results to the user in an understandable manner. The solution also allows the users to manage the data as needed, create data plots and automatically generate reports. This tool was intended to simplify the processing steps of the analytical pipeline, so the users can focus on what really matters: understanding the results.

1 INTRODUCTION

In recent years, the usage of microarray technologies in functional glycomics has grown exponentially due to the great potential of lectin and glycan microarrays for this field. These types of array provide deep insight regarding the interactions between glycans and lectins, useful in multiple clinical and research settings. For example, they can be used to analyse the glycosylation profile of glycoconjugates, to perform quantitative analysis of lectin-glycoprotein interactions, to discover glycan-related biomarkers in cancer and to study the cell-surface glycans, among many other applications (Hu and Wong, 2009).

In this context, the project Glicobiomed was born as a collaboration between various centres in order to study the role of glycans in different settings, to develop new methodologies for glycoanalysis and ultimately, to obtain novel biomarkers.

To develop a successful project, it is necessary to have tools that harness the potential of these microarray techniques. Nevertheless, data analysis in this field has not reach is highest potential due to the limited availability of analytical software. Even if some platforms may be available, they do not fulfil the requirements of the end-users involved in this project. Therefore, they usually follow a manual analytical pipeline, which is not desired.

To be able to surpass the limitations associated to the traditional approach, we have developed a microarray analysis software called SugarArray, which provides a solution regarding lectin and glycan microarray data processing, visualization and analysis, encapsulated in a user-friendly stand-alone software. In order to formulate the software requirements and develop the desired solution, we followed a user-centred design approach, allowing the end-users to be involved in every stage of the development, and assist in the making of a tool that actually fulfils the needs of the users.

[a] https://orcid.org/0000-0002-4078-9275
[b] https://orcid.org/0000-0002-7160-1191

Sucre, A., Pazos, R., Reichardt, N. and Garín-Muga, A.
SugarArray: A User-centred-designed Platform for the Analysis of Lectin and Glycan Microarrays.
DOI: 10.5220/0008960101060116
In *Proceedings of the 13th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2020) - Volume 5: HEALTHINF*, pages 106-116
ISBN: 978-989-758-398-8; ISSN: 2184-4305

2 RELATED WORK

No tools have been found in the literature that fill all the gaps in microarray analysis detected by the end-users involved in the project, as most of the scientific efforts in this field are put on developing better and more diverse microarrays instead of developing tools for analysis.

Almost every commercially available software for microarray analysis focus on gene or peptide microarrays, which is not useful for lectin and glycan analyses. Also, some other tools were detected that do focus on glycans and lectins, but they analyse data coming from other technologies, such as mass spectrometry (Goldberg et. al., 2005; Maass et. al., 2007), or focus more on the molecules structure and their representation (Aoki-Kinoshita, 2008), which is still not the target of this project.

Nevertheless, two software developed specifically for glycan analysis were detected. The first software is a stand-alone program composed by a suite of modules to store, retrieve and display glycan microarray data. It provides an internal database to store all the information related to the glycans, their associated proteins and the experimental data; it also provides different tools for data visualization, sorting and filtering; and finally, it includes modules for automatic plots generation (Stoll and Feizi, 2010).

The second tool is called GLAD (Glycan Array Dashboard), and it is a web-based tool that provides functions to visualize and analyse glycan microarray data, and also compare information coming from different experiments. It has a module for basic plot generation (i.e. bar charts) for single-sample data, or complex plot generation for comparison of samples. This tool also includes a module for data normalization between samples, necessary for comparison (Mehta and Cummings, 2019).

However, none of the described programs fulfil all the needs of the project, according to the participating end-users. The main drawback is that the software interfaces are not as intuitive and easy-to-use as desired, so the different included functionalities are not easy to exploit. Another downside is that these programs were designed strictly for glycan array analysis, so they do not work with lectin arrays, which is one of the main requirements. Therefore, the best approach was to create a tool from scratch and include all the desired functionalities gradually.

3 LECTIN AND GLYCAN MICROARRAYS

3.1 Lectins, Glycans and Glycoproteins

Lectins are a group of proteins that present certain binding behaviour toward carbohydrates, specifically soluble carbohydrates and the residues of glycoconjugates (i.e. glycans). These proteins bind saccharides reversibly and with high specificity, but can have more than one-binding size along a single molecule, so they can be specific for more than one sugar molecule at once.

Lectins can be found commonly in nature: in plants, animals and bacteria; and they have been associated with a broad set of functions depending on where they are found. It is interesting to highlight their role as recognition molecules in cell-molecule and cell-cell interactions, affecting a wide range of cellular events (Lis and Sharon, 1998).

These molecules do not only have a role in nature, but many different usages have been given to them in clinical and experimental settings, for example, in blood typing, histochemical analyses and biomolecules purification.

Another important group of biomolecules are the glycans. In general, glycan is considered as a synonym for polysaccharide, which are “compounds consisting of a large number of monosaccharides linked glycosidically” (IUPAC, 1997), usually formed by more than 10 sugar residues. Nevertheless, in this context the term will be associated to the saccharide portion of a glycoconjugate molecule, which are molecules of carbohydrate bonded to other compounds (i.e. glycoprotein, glycolipid or proteoglycan) (Dwek, 1996).

Historically, it was believed that the solely function of sugar molecules was being a source of energy, but now it is well-known that they have many other functions in the biological systems. Glycomics is the science that focuses on studying the glycome of the organisms, trying to define structures and functions of carbohydrates in nature.

Glycoproteins are formed through direct interactions between glycans and proteins. The glycans of these molecules can also be attached to other macromolecules, which indirectly control the glycoprotein conformation, stability, turnover, oligomerization and cell surface resident time (Cummings and Pierce, 2014).

It is important to highlight that the proteins conforming the glycoproteins are not lectins. Lectins may temporary bind the sugar fragment of glycoproteins in order to execute certain functions,

but the interaction between the glycan and protein forming a single glycoprotein is permanent and formed differently.

Based on how the glycoproteins are joined, they can be classified as N-linked or O-linked molecules. N-linked glycoproteins are formed in the endoplasmic reticulum, where the glycan is attached to the protein through a nitrogen atom. On the other hand, O-linked glycoproteins are formed in the Golgi apparatus, where the linked is created through an oxygen atom. Then, the newly formed molecules travel to the plasma membrane, where the saccharide part is placed facing out (Robb, 2019).

Glycoproteins promote various cellular functions such as cell adhesion, cell-matrix interactions and cellular signalling.

It can be expected that abnormalities in the synthesis of glycoproteins can be associated with numerous conditions and diseases; therefore, it is important to understand their mechanisms and functions in the different organisms.

3.2 Microarray Technologies

Glycan and lectin microarrays are currently considered two of the most relevant technologies in functional glycomics, as they help on understanding the function, interactions and structures of glycans and glycoconjugates.

A lectin microarray is a functionalized glass plate with numerous micrometric wells containing immobilized lectins. As previously described, lectins have a recognition domain for carbohydrates, so these panels are very useful in order to study glycans and glycoproteins. In a microarray experiment, each of the immobilized lectins can interact with a specifically fluorescently labelled molecule, thus generating a characteristic interaction profile for each glycoconjugate, i.e. it is possible to identify and measure the glycoconjugates found on a sample based on the interactions with the lectins on a plate (Hu and Wong, 2009).

Equivalently, a glycan microarray has wells containing immobilized glycans instead. These glycans are able to interact with specifically fluorescently labelled lectins and help on detecting the presence and estimating the concentration of different lectins in a given sample.

Functioning microarrays are created using a spotter, which is a tool that deposits the different ligands in the well where they correspond, based on the experiment design information sheet (GAL file). Usually, biological replicates are included in the design in order to obtain more reliable and significant results; this is done by filling multiple wells with the same ligand.

In a single microarray plate, it is possible to have multiple replicated subarrays of ligands, which allow the researchers to study multiple samples at once with respect to the same set of ligands. Each sample is poured over one of the subarrays, so the information generated on each section of the microarray will correspond to a single sample. Each subarray is delimited, so the samples are not mixed up.

3.3 Microarray Data Processing

As previously mentioned, the samples that are going to be analysed in a microarray should be molecularly marked somehow. The most common technique is to use fluorescent molecules, which are bonded to the molecules in the sample of interest. Then, when these molecules interact with the ligands in the microarray wells, the fluorescent molecule is released, and a fluorescent signal is emitted. By analysing the emissions of fluorescent signals, the researcher is able to identify the molecules in the sample and estimate their concentration, based on the known relations between glycans and lectins. Depending on the location of the fluorescent emissions, it is possible to know which microarray ligands interacted with molecules of the sample.

In order to obtain information regarding the fluorescent emissions, a specific scanner is needed, which captures the emission of light and generates a monochromatic image (TIFF file) describing the emissions. Then, different programs can be used to extract the intensity data, for instance the ScanExpress, which measures fluorescence intensity at each point, recognises different patterns, makes certain corrections and adapts the measurements to the array design according to the GAL file finally generates a CSV file describing the intensities associated to each microarray spot and various associated statistics.

3.4 Microarray Data Analysis

Traditionally, researchers read and analyse the intensities CSV files using a spreadsheet software such as Microsoft Excel. This approach allows them to create data charts and basic plots to visualize and understand the data, but managing large amounts of data and stablishing comparison protocols is not so easy due to limitations of spreadsheet software. Also, this approach is prone to misleading results, due to the multiple variables that are manually controlled and can lead to erroneous calculations. Finally, this

approach is very time consuming because every calculation and chart is created manually; the whole analytical process should be supervised by the researchers.

Based on these observations it became interesting to develop a tool that automatizes the analytical pipeline and enables the users to visualize the data and generate reports on a fraction of the time. This process assures reliable results, as they are processed automatically, not relying in manually performed tasks that are prone to errors.

4 SugarArray

The SugarArray solution is a stand-alone software that was developed in order to analyse the intensity data generated after scanning lectin and glycan microarrays. SugarArray processes the data and generates various types of plots so that it is possible to analyse the data in depth and generate reports in an easy and visual manner.

The software was developed following a user-centred design approach and consists of a graphical user interface (GUI) designed using QT, and functionalized using Python; and a set of analytical modules, also developed in Python.

4.1 Software Design Approach

The user-centred design (UCD) approach is an iterative design process where the end-users are involved in all the stages of design and development. This approach enables to better describe the user needs and help the developers while defining the solutions to the user-detected problems. Different methods were considered along the process, but they can be sorted in 2 categories: investigative and generative methods. The first category comprises the techniques that allowed us to understand the context of the problem and to better define the user needs, while the second set of techniques allow the users to present their requirements for the software and their ideas that may help developers achieving the project goals (Nugraha and Benyon, 2010).

As previously stated, the development of this software was embedded within a larger glycomics project, so a team of end-users was available in order to follow this approach properly. This team consisted of five glycomics researchers which were actively involved in the design process along with the developers. The number of involved researchers was chosen following Jakob Nielsen's recommendations regarding usability testing (Nielsen, 1993; Nielsen, 2000).

The workflow of the followed UCD methodology is represented in Figure 1.

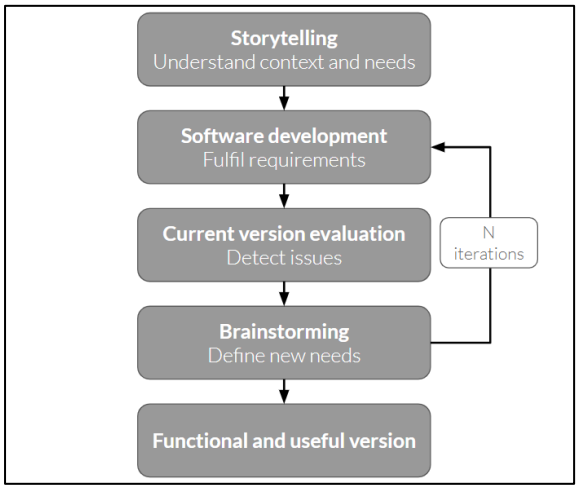

Figure 1: User-centred design methodology workflow.

The initial design step consisted on having a face-to-face meeting between developers and two of the chosen end-users where we used a storytelling tool so the users could help developers understand their current analytical practices and protocols, the data they usually exploit, the results they obtain and the general context of the project. They also provide information regarding their initial expectations with respect to the new analytical tool, and described their dissatisfaction about the commercial tools available.

The design team prepared an initial basic version of the software, trying to incorporate the different elements needed to replicate the behaviour of their current practices, but in a simpler and automated manner.

All the involved end-users tested this first version of the software and were all pleased with the initial results. Afterwards, the design process consisted on having periodically face-to-face meetings where a brainstorming approach was followed so both, the end-users and developers, define together further requirements of the tool, interesting functionalities to incorporate and the expected looks of the application. After each brainstorming session, the developers modify the software in order to fulfil the newly appearing requirements and present each new version to the end-users for feedback, who evaluated the tool focusing on its usability and effectiveness. This cycle continued until a fully-functional tool that fulfil all the user requirements was developed.

4.2 Analytical Pipeline

In order to perform a complete analysis on the data generated from a microarray experiment the steps shown in Figure 2 are followed. This pipeline represents the interfaces through which the user interacts with the software, and the functional modules that are executed after each user's action.

As shown in Figure 2, the GUI comprises 4 dialogs: (1) the new project dialog , (2) the main window, (3) the charts dialog and (4) the report dialog.

On the other hand, the set of analytical tools includes 3 main modules for: (A) data processing, (B) plot generation and (C) report generation.

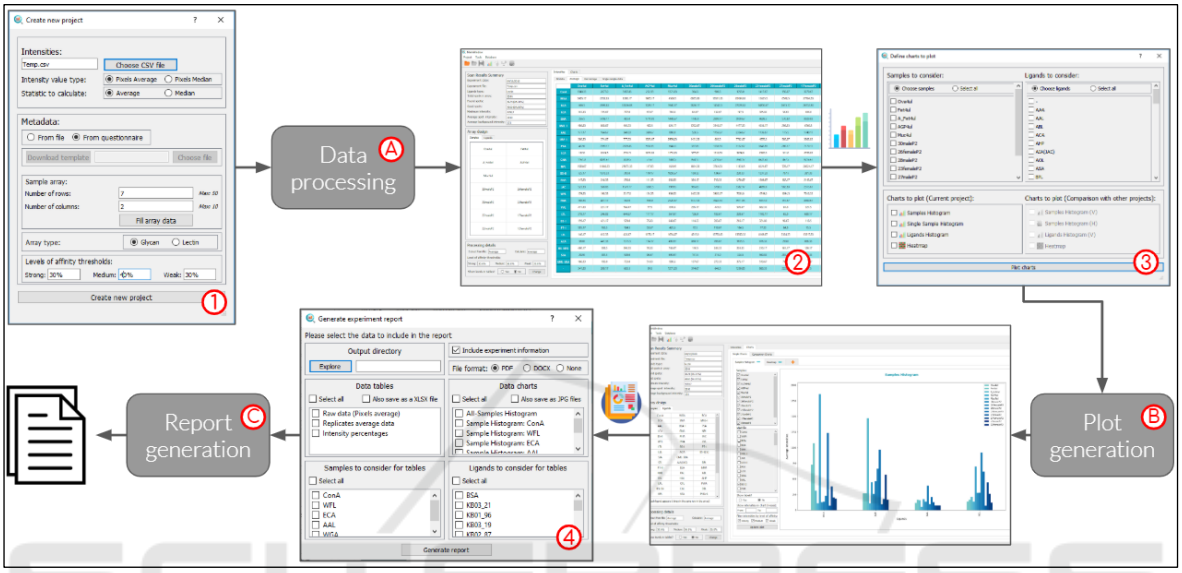

Figure 2: SugarArray analytical pipeline.

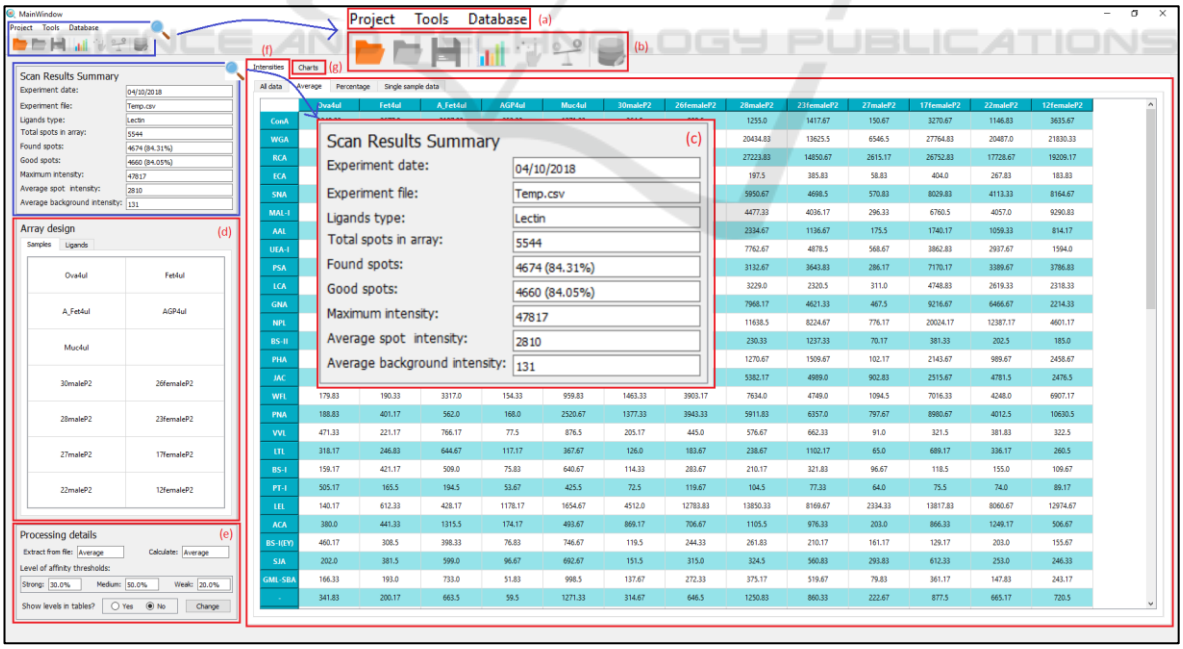

Figure 3: GUI - Main window: (a) Menu bar, (b) Tool bar, (c) Scanner metadata section, (d) Array design section, (e) Processing details section, (f) Data tables section and (g) Data charts section.

All the mentioned dialogs and modules will be shown and described in detail in following sections.

4.3 Graphical Interface

4.3.1 Main Window

The main window corresponds to the primary point of interaction between the user and the analytical software. It allows the user to start new analysis, visualize and manage the data and generate and view charts. The interface is shown in Figure 3 and all its composing elements will be described next.

The upper section of the window contains two independent bars: (a) the menu bar, where the different software actions are sorted in menus; (b) and the tool bar, where the same actions are presented, but in this case, using icons. This disposition allows the user to perform the analytical actions in the way that is more intuitive for each.

Then, on the left side of the window we have three elements: (c) the scanner metadata section, where all the information regarding the capture step is summarized (experiment date, experiment data file, total spots in the array, number of found spots, number of good spots, maximum detected intensity, average spot intensity and average background intensity) to provide certain insight on the performed experiment; (d) the array design section, where two data tabs are contained, one describing the disposition of samples within the whole experiment array and the other describing the disposition of the ligands within each of the subarrays; and finally, (e) the processing details section, where the statistics chosen to be extracted and calculated are shown, along with the level of affinity thresholds modifiable by the user.

On the right side of the main window we have (f) a data tab containing various tables describing the intensity data, and (g) another tab where all the generated charts will be displayed.

The intensities data section contains 4 tables in 4 tabs: (1) one displaying the data extracted from the intensities file in a table having as many rows as ligands and as many columns as samples, (2) another table showing the average/median values calculated between the replicates of the same ligand found within the array, (3) the third one displays the same calculated values but in terms of percentages, and (4) the final table shows the averages/medians and percentages associated to a single sample as chosen by the user. The latter allows the user to select which sample to display and how the data will be sorted to analyse each sample in depth (Figure 4).

Finally, the charts tab is filled-up on the fly as the user creates new plots. Each newly created chart will be displayed in an independent tab within the charts section. Each tab has a set of tools that allow the user to modify the charts interactively after creation, by modifying the ligands/samples included in the charts and how the information is sorted, deciding whether the labels are shown, etc. (Figure 5).

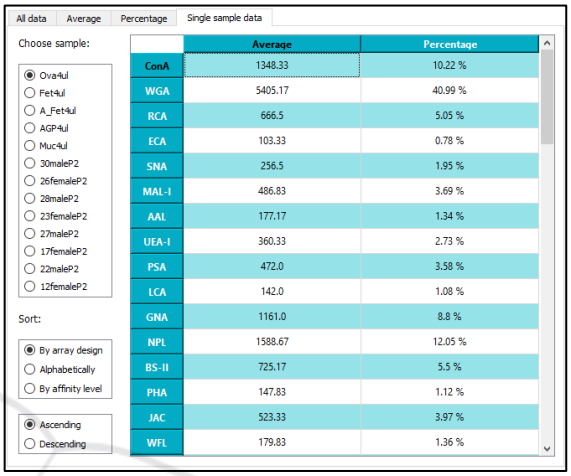

Figure 4: GUI - Main window (Sample view).

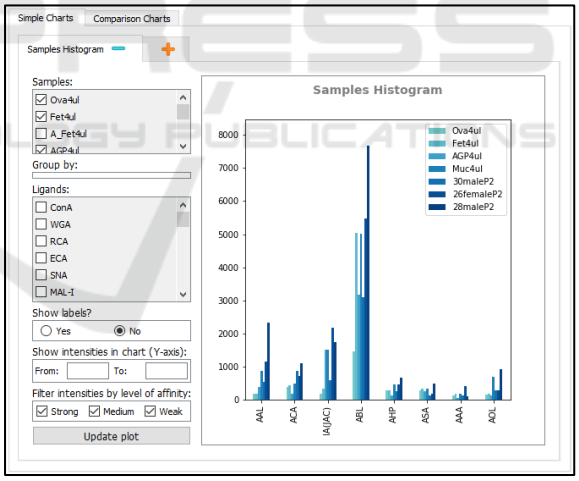

Figure 5: GUI - Main window (Chart view).

4.3.2 New Project Dialog

The new project dialog allows the user to input the files and data necessary to define a new experiment in the platform. The interface is shown in Figure 6.

On top of the dialog we have (a) the intensities section, where the user must select the intensities file generated by the scanner software and define which information to extract from it and which statistic to calculate between replicates.

Then, on the middle of the dialog (b) the metadata section is shown. This section allows the user to provide further information needed for the analysis. The user is able to (1) upload a metadata file or (2) define the metadata in a questionnaire. For the first option, a data template must be followed. For the second option, different widgets are displayed so the user can provide all the metadata values easily. First, the user must define the sample array disposition and then write down the names of the samples in the array. Then, the user must define the type of array and the percentages associated to the levels of affinity to classify the sample-ligand interactions.

Finally, in the bottom of the dialog (c) the "create new project" button is shown. When this button is clicked , the data processing pipeline starts.

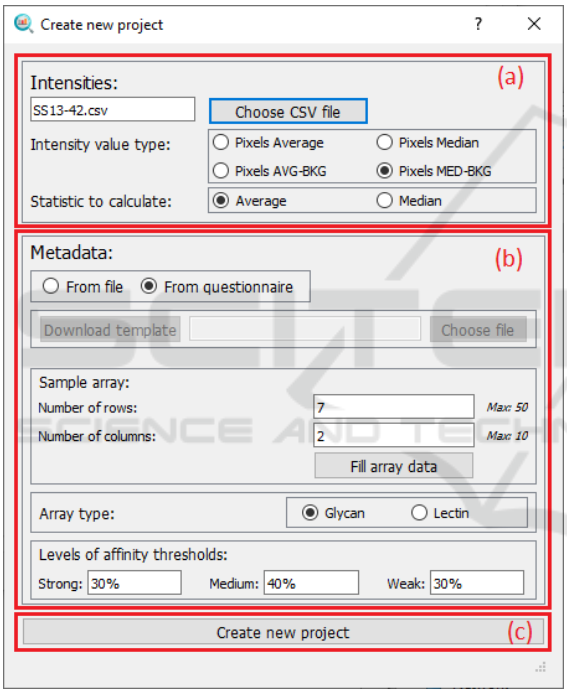

Figure 6: GUI - New project window: (a) Intensities section, (b) Metadata section and (c) Create button.

4.3.3 Charts Dialog

The charts dialog allows the user to define the different charts that are needed to analyse the data in depth. The interface is shown in Figure 7.

On the top section of the window we have two elements: (a) a list widget that allow the user to select the samples that must be included in the charts, and (b) another list widget, in this case to select ligands.

On the centre of the dialog we have (c) a list of check boxes, so the users can choose the plots to draw; and (d) a list of colour scales so the user can define the appearance of the charts.

Finally, in the bottom of the window we have (e) the "plot charts" button, so when it is clicked, the plots are created.

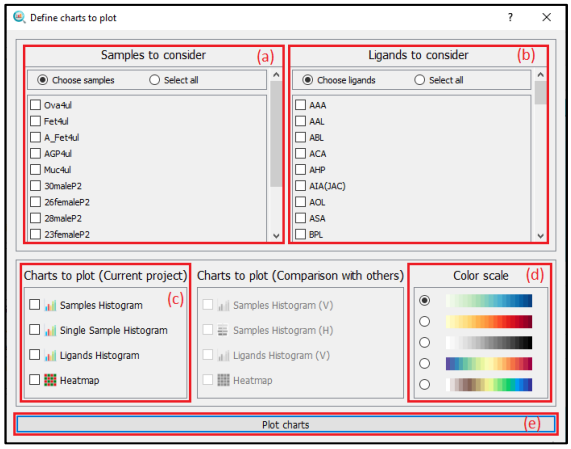

Figure 7: GUI - Charts dialog: (a) Samples list, (b) Ligands list, (c) Charts list, (d) Colour scale list and (e) Plot button.

4.3.4 Report dialog

The report dialog allows the user to generate reports and define which information to show in them. It is possible to generate full reports in PDF or DOCX format, but also generate data table reports (XLSX files) or export the generated plots for further use.

The interface associated to this report wizard is shown in Figure 8.

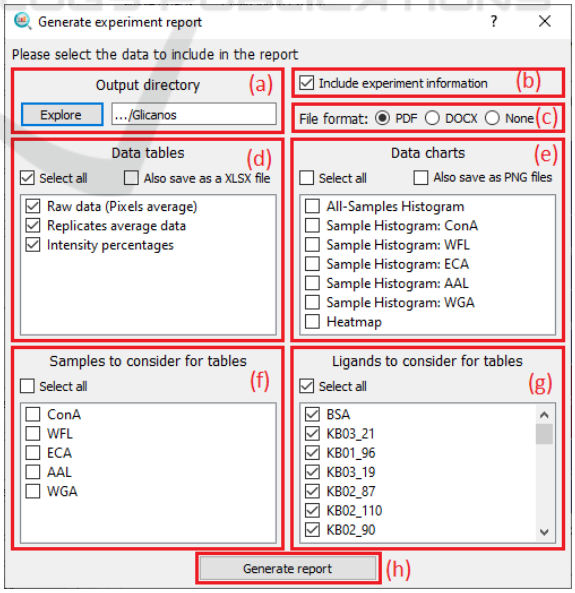

Figure 8: GUI - Report dialog: (a) File explorer section, (b) Information checkbox, (c) Format section, (d) Tables section, (e) Charts section, (f) Tables-Samples section, (g) Tables-Ligands section and (h) Generate button.

On the top of the dialog we have (a) the file explorer section, where the user must define where to store the files that will be generated, (b) a checkbox to decide whether or not to include the experiment information, and (c) a list to define the output format.

In the middle of the dialog we have 2 lists: (d) one allows the user to select the data tables to include in the report and (e) the other one, to decide which plots to include.

Finally, in the bottom of the dialog we have another set of lists and a "Generate report" button (h). The latter set of lists allow the user to select the samples (f) and the ligands (g) to be included in the tables.

4.4 Analytical Modules

4.4.1 Microarray Data Processing

The data processing module of the software is composed of two analytical sub-modules: GetMetadata and GetValues module. These modules include a variety of methods that are followed in order to process the data, extract the relevant information and calculate statistics; these methods are mainly based on well-known Python packages for data management and statistics: NumPy and Pandas.

The GetMetadata sub-module comprises the functions necessary to extract the metadata information from a text file if it was provided. Otherwise, the metadata information is directly obtained from the "New Project" interface.

First, the software checks if the provided file has the expected format and if so, extracts all the data and stores it in such a way that can be exploited by the software and easily shown in the main window later.

On the other hand, the GetValues sub-module contains all the functions needed for intensity data extraction and processing.

The first step is to read the intensities file, extract the metadata associated to the scanner, detect the file column where the chosen-statistic values are stored, and save the selected raw data in a dataframe. This initial dataframe contains a list of items, where each item contains the coordinates and intensity value associated to each array spot. However, this data structure is still not compatible with the software.

Then, a series of functions are performed on the raw data to extract the ligands information and modify the data structure based on that. In particular, we detect how many replicates are found for each ligand and define a dataframe describing how the different ligands were distributed in the microarray; we also assure that independent non-replica ligands are named differently; and finally, after a series of concatenated transformations, we obtain a dataframe having as many rows as ligands in and as many columns as samples, assuring that the data associated to non-existing ligands or samples is excluded.

Once the data is structured in a ready-to-show manner, the next step is to calculate the chosen statistic (average or median) between the replicates of the same ligand. The obtained dataframe has the one row for each ligand (instead of the n replicates that were shown before).

Next, we calculate the percentage associated to each calculated statistic. Each percentage represents the ratio of intensity of each ligand, in comparison to the maximum intensity associated to each sample.

Finally, we must define the "levels" dataframe based on the given values associated to each affinity threshold and on the calculated statistics dataframe. We must state which is the affinity level associated to each ligand for each sample. The affinity percentages describe which number of ligands can be considered as having a high/moderate/weak interaction with each studied sample. The default values defined by glycomics experts involved in the project state that the 30% of samples having a higher intensity value can be associated to strong interactions, the next 50% with moderate interactions and the bottom 20% of the ligands, a weak interaction with the sample. The percentages can be modified as desired, and so the number of ligands defined by each category. A dataframe is returned, describing the level associated to each position in the matrix.

The GetValues sub-module may also work following a shortened path, where only the levels dataframe is recalculated based on modifications made by the user in the main window.

4.4.2 Plots Generation

This analytical module comprises the functions needed for plot generation and edition. The methods followed to create the different plots are based on functions from a set of well-known Python packages for data visualization: matplotlib and seaborn.

Currently the tool allows the users to create four types of charts, shown in Figure 9.

The samples histogram describes the intensity data for all the chosen ligands regarding all the chosen samples. The chosen ligands are represented in the X-axis, the intensities in the Y-axis and a different colour is assigned to each sample. Therefore, each shown bar represents the mean/median intensity for a ligand regarding a sample. The single sample histogram is similar, but in this case, only the intensity values for

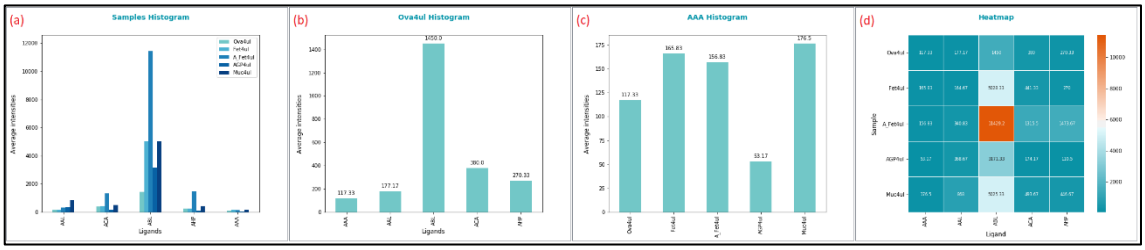

Figure 9: Generated plots: (a) Samples histogram, (b) Single sample histogram, (c) Ligand histogram and (d) Heatmap.

one single sample are represented and thus, one bar per ligand is shown.

On the other hand, the ligand histogram represents the chosen samples in the X-axis and draws the bars represented the mean/median intensity detected on each sample for a single chosen ligand.

Finally, the heatmap is a colour matrix having as many rows as chosen samples and as many columns as chosen ligands, where each cell is coloured based on the intensity value associated to each ligand for each sample, following a colormap.

The module creates the plots based on the initial choices of the user, but can also modify them based on the interactions between the user and the interface.

Every time the user changes something in a chart view on the main window, such as add/remove ligands/samples, show/hide labels, filter based on values or filter by level of affinity a new plot is created to substitute the previous one that was shown to the user. The only non-modifiable feature is the colour scale, so the one chosen at first is kept.

4.4.3 Generation of Reports

The report-generation module allows the user to export the processed data and the generated charts when desired, providing certain flexibility in order to fulfil all the requirements defined by the end-users.
Currently, it is possible to generate a full report in PDF or DOCX format. It is possible to include different sets of information on each generated report, choosing between (1) the experiment data (scanner-provided information, samples array and ligands array), (2) the data tables (raw, statistics or percentages tables) or (3) the generated plots. The user can decide to include all data tables/plots or just a selection of them.

Regarding table generation, the user is also able to decide which samples and ligands to include.

Finally, it is also possible to export the generated tables independently in a format that is easier to manage (XLSX), and also to save the plots as images (PNG).

When generating a report, the users are able to select between these options the ones that suit them better. The software will generate a set of files based on the user choices and stored them all in the selected location.

Just like the plots, the reports can be generated iteratively as needed, in order to include different sets of information on each, without the risk of losing the ignored experiment data.

5 DISCUSSION

In this paper we proposed a microarray analysis stand-alone software that is able to study both lectin and glycan microarrays, and we also describe the user-centred-design approach that was followed in order to develop a tool aligned with the real necessities of real users.

After having analysed the different commercial tools available we did not find a tool that was compatible with both types of arrays nor included the analytical functionalities that were desired for the software, so a new stand-alone program was created from scratch.

This newly-developed software allows the end-users to simplify their analytical pipeline, execute all the tedious tasks automatically and without mistakes (which cannot be assured when executing manual processing), and focus on result interpretation steps rather than on data processing itself.

SugarArray allows researchers to input the raw analytical data, obtain tables and plots in just a few steps, and exporting the generated information in different formats for further analysis, or for report generation.

A proper validation process was not conceived at this stage of the project, because the UCD approach that was followed allowed us to have constant testing and feedback from end-users regarding the software behaviour. Nevertheless, a pre-validation process was conducted by the end-users in order to test the final functional version of the software. Also, the development team is currently defining the protocols

that will be followed in the future in order to properly validate the software.

The pre-validation analysis consisted of two main tasks: (1) evaluate the behaviour of the different modules and the interaction between the user and the software and (2) compare the analytical results obtained using the SugarArray tool with those obtained following the manual approach.

The feedback that has been given so far has been positive. The comparison between SugarArray results and manually obtained results was favourable; every experiment that was involved in the comparison returned the same results for both approaches. Concerning software usability, the users' opinions were also optimistic; all the required functionalities were included and the definition of the analytical pipeline within the software was simple and easy-to-understand. However, some minor visual improvements have been identified, and they will be considered in following iterations of development.

6 CONCLUSIONS

The chosen user-centred-design methodology allowed the developers to successfully capture the end-user necessities and develop a solution that appears to be appreciated by the end-user.

The development of this tool was embedded within a larger glycomics project associated to complex processes and analysis. Therefore, having a tool that automatizes the data analysis steps is important to allow researchers to focus on the meaning of the analytical results, accelerating the obtention of clinically significant insight.

This functional version of the software is still on its validation phase. As mentioned, initial pre-validation steps were followed in order to receive early feedback from the involved users regarding their impressions of the software. This initial feedback was positive, but further insight obtained on a structured manner, and coming from project-related and external end-users is needed.

Currently, the validation protocols are being defined in order to obtain unbiased reviews. These protocols will consider (1) software usability, (2) results accuracy, (3) software efficiency and (4) comparison with other software or approaches.

Besides software validation and the already-detected visual improvements, the future work will be focused in the development of new functionalities sorted in three development lines: Data annotation, data comparison and outlier management.

For data annotation we are willing to build a database to store information regarding ligands classification, interactions, structures, etc. The software will also include the tools to manage the database intuitively, and other modules to annotate the ligands based on the database-stored information.

For data comparison, we need to develop tools for data normalization in order to make the information associated to different studies comparable; we are also willing to expand the plot-generation module in order to incorporate data from different studies and generate new tools for comparison.

Finally, the necessity for an outlier management tool has arisen, that allow the users to detect those values that do not behave as expected and handle them in such ways that do not affect the analytical results.

ACKNOWLEDGEMENTS

This work has been funded by the Basque Government by means of the ELKARTEK program within the context of the Glicobiomed (BMG18) project: "Tools and Opportunities for Biomedical Glycoscience".

We would also like to offer our special thanks to the CIC biomaGUNE Glycotechnology Lab team for their participation in the project as the end-users committee involved in the user-centred design process.

REFERENCES

Aoki-Kinoshita, K. (2008) An introduction to bioinformatics for glycomics research. PLoS. Computational Biology 4(5):e1000075. doi: 10.1371/journal.pcbi.1000075.

Cummings, R. and Pierce, J. (2014) The Challenge and Promise of Glycomics. Chemistry and Biology 21(1):1-15. doi: 10.1016/j.chembiol.2013.12.010

Dwek R. (1996). Glycobiology: Toward Understanding the Function of Sugars. Chemical Reviews 96(2): 683-720. doi: 10.1021/cr940283b

Goldberg, D., Sutton-Smith, M., Paulson, J. and Dell, A. (2005). Automatic annotation of matrix-assisted laser desorption/ionization N-glycan spectra. Proteomics 5(4):865 – 875. doi: 10.1002/pmic.200401071.

Hu, S. and Wong, D. (2009) Lectin microarray. Proteomics Clin Appl. 3(2):148-154. doi: 10.1002/prca.200800153

IUPAC (1997). Compendium of Chemical Terminology. Compiled by A. D. McNaught and A. Wilkinson. Blackwell Scientific Publications, Oxford (1997).

Online version (2019-) created by S. J. Chalk. doi: 10.1351/goldbook.

Lis, H. and Sharon, N. (1998). Lectins: Carbohydrate-Specific Proteins That Mediate Cellular Recognition. Chemical Reviews 98(2):637-674. doi: 10.1021/cr940413g

Maass, K., Ranzinger, R., Geyer, H., von der Lieth, C. and Geyer, R. (2007) "Glyco-peakfinder" – de novo composition analysis of glycoconjugates. Proteomics 7(24):4435 – 4444. doi: 10.1002/pmic.200700253.

Mehta A. and Cummings R. (2019). GLAD: GLycan Array Dashboard, a visual analytics tool for glycan microarrays. Bioinformatics 35(18): 3536-3537. doi: 10.1093/bioinformatics/btz075

Nielsen, J. (1993). Usability engineering. Academic Press, Boston (1993).

Nielsen, J. (2000). Why you only need to test with 5 users? Nielsen Norman Group. Available at: www.nngroup.com/articles/why-you-only-need-to-test-with-5-users [Accessed December 11, 2019]

Nugraha, S. and Benyon, D. (2010). Designing Interactive Systems: A comprehensive guide to HCI and interaction design. Pearson Education Limited. Harlow, 2nd edition.

Robb, A. (2019). What Are Glycoproteins? - Definition, Functions & Examples. Study.com. Available at: study.com/academy/lesson/what-are-glycoproteins-definition-functions-examples.html [Accessed October 22, 2019]

Stoll M. and Feizi T. (2010). Software Tools for Storing, Processing and Displaying Carbohydrate Microarray Data. In Glyco-Bioinformatics – Bits 'n Bytes of Sugars Proceedings. BEILSTEIN INSTITUT.