Grant Agreement no. 643529

iManageCancer

Empowering patients and strengthening self-management in cancer diseases

Research and Innovation Action

PHC-26-2014: Self management of health and disease: citizen engagement and mHealth

D8.1
Report on implemented data analysis and data mining services

Contractual Due Date: 31 July 2017
Actual Submission Date: 11 August 2017

Lead partner for deliverable: FORTH

Dissemination Level: Public

Revision: v1.0

COVER AND CONTROL PAGE OF DOCUMENT	
Project Acronym:	***iManageCancer***
Project Full Name:	Empowering patients and strengthening self-management in cancer diseases
Project Duration	1 February 2015 - 31 July 2018
Deliverable No.:	D8.1
Deliverable Name:	Report on implemented data analysis and data mining services
Nature (R, DEM)[1]	DEM
Dissemination Level (PU, CO)[2]	PU
Version:	1.0
Actual Submission Date:	11 August 17
Editor: Institution: E-Mail:	Lefteris Koumakis FORTH koumakis@ics.forth.gr
Contributors (Institution)	FORTH, CI-eCancer, IEO, USAAR, BED
Reviewers (Institution)	Stephan Kiefer (FRAU), Eric Herve Ngantchjon (Philips)

Copyright

The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 643529.

[1] **R** = Document, report (excluding the periodic and final reports), **DEM** = Demonstrator, pilot, prototype, plan designs

[2] **PU** = Public, fully open, e.g. web, **CO** = Confidential, restricted under conditions set out in Model Grant Agreement

Document History

Issue Date	Version	Changes Made / Reason for this Issue
31/07/2017	V0.9	First version for internal review.
11/08/2017	V1.0	First official version after review process.

Table of Contents

1 Executive summary 5
2 Introduction 6
2.1 Reference architecture 6
2.2 Supported scenarios 7
2.2.1 For the doctor and the researcher 7
2.2.2 For the patient 8
2.2.3 For the administrator 8
2.2.4 For the research project 9
2.3 Data 10
3 Query builder 10
4 Data analysis 15
4.1 Clustering 15
4.2 Feature selection 15
5 Alerts 16
6 Data anonymization 18
7 Conclusions 20
8 References 22

1 Executive summary

The iManageCancer platform collects multidisciplinary data covering areas from the medical, the environmental and the lifestyle domains. The objective of the WP8: Smart analytical data services is to extract information from the diverse data of iManageCancer and transform it into an understandable structure for better knowledge and further use. To do so, an effective analytical framework has been designed and implemented based on real user requirements. At the first step we gathered information related to users requirements in order to understand the needs from such tools not only for the statisticians but also for the doctors and the researchers. For that reason the project conducted an in depth user requirements analysis along with a survey for the patients and a workshop where physicians, developers and patients discussed and concluded to the user needs and the smart analytical services scenario.

The implemented framework (Smart Analytics Framework) does not aim to provide new data mining and data analysis algorithms but an end to end solution for not IT users to analyse and understand the data collected from the iManageCancer tools. It targets better knowledge understanding and to bring together valuable information in a visual form, supporting exploration. Information about the visualization techniques for the smart analytics can be found in the public deliverable/demonstrator D8.2 of iManageCancer.

2 Introduction

The iManageCancer platform supports different types of data including lifestyle data and clinical data which will be continuously evaluated against the personal health record and history. Feedback towards individuals will be automatically generated at the point of need. For a successful data analysis framework, apart from the data mining algorithms, access to integrated data and knowledge of the underlying data and data structures is needed.

The heterogeneity and scale of clinical, environmental and lifestyle data raises the demand for seamless data access along with the availability of powerful and reliable data analysis operations, tools and services. Obviously the amount of information available, the heterogeneity of the information and the wide range of biomedical ontologies dictate the identification of a solution able to handle all this information. iManageCancer implements interlinking of several ontologies into a global schema to integrate all internal and external data, called Smart Access Layer (see bottom of **Figure 1**).

The data analysis and data mining tools aim to extract information from the diverse data of iManageCancer and transform it into an understandable structure for enhancing knowledge extraction. Smart data analytics provides mechanisms able to identify patterns or trends in data, screen pre-frailty states and provide different views of data for new management plans. Data mining consists of various methods and algorithms which have been applied to many research areas and the healthcare domain is not an exception. Nevertheless, the platform is modular enough to allow any other data mining algorithm to be directly embedded in the whole workflow.

In order to take into account the opinion of the patients, an online survey was created and promoted to the cancer community via ecancer.org[3] along with a host of other platforms to distribute this in English, Italian, German and Greek. People from all across Europe with 226 surveys submitted their opinion. The report of the surveys can be found in the deliverable of iManageCancer D2.2 'Scenarios and use cases including the ethical and legal aspects' from the web site of the project[4] and the questionnaire is provided as Appendix in the deliverable. Among others, the following conclusions related to the analytical services are driven from this survey:

- More than 80% of responders want to have tools for analysing their health data.
- Around 80% are willing to provide their health data for research with only 50% giving consent online.
- Security for sharing is most important for all respondents. Interestingly the security is seen more important in sharing health data than in online banking

Based on the results of the survey, an intensive review of existing scenarios and use cases the consortium concluded to five categories of scenarios including a scenario for data analysis.

2.1 Reference architecture

An outline of the reference architecture for the smart analytical services is shown in **Figure 1** where, the basic operational modules of the system are also shown. Smart analytical services try to go much further than traditional statistics by examining the raw data and then attempting to hypothesise relationships within the data. As shown in **Figure 1**, data analysis and data mining in iManageCancer is an iterative approach, which combines data from the semantic layer of iManageCancer, pre-processes the data, performs the analysis and provides the results for visualization based on the data distillation model [1]. The loop closes with the interaction of the

[3] http://ecancer.org

[4] http://imanagecancer.eu/resource-centre

end user who can refine the results and continue with a drill-down analysis to extract knowledge from cohorts with specific criteria.

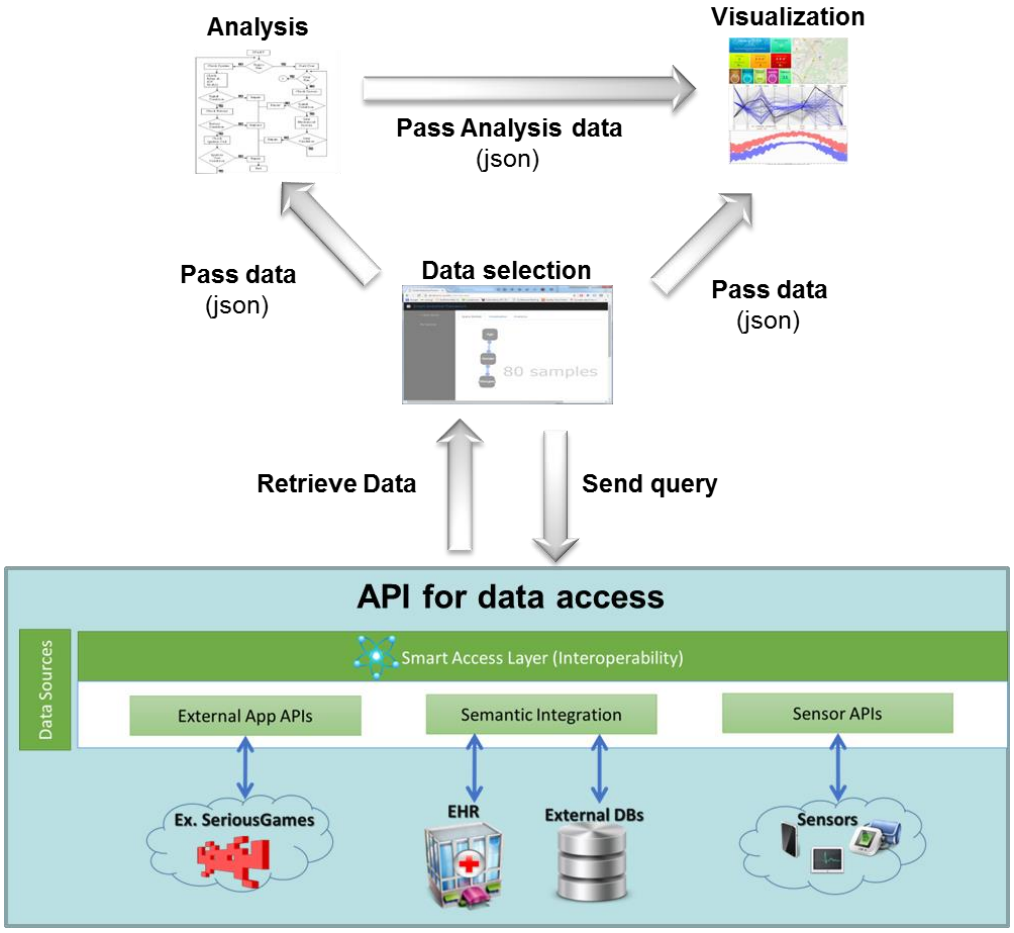

Figure 1. Architecture of the analytical services over personal health infrastructure

2.2 Supported scenarios

2.2.1 For the doctor and the researcher

End users of the smart analytical services are physicians, data miners, statisticians and data managers. The workflow for the main scenario of the data analysis and mining consists of the following steps:

- **Step 1 (Overview of the data)**: The end user logs-in to the system. An overview of the iManageCancer data for the whole population using various visualization techniques, such as charts and plots, is available.
- **Step 2 (Create and analyze cohorts):** The user can select, using the interactive visualization techniques, specific features and set inclusion/exclusion criteria in order to create a new cohort, e.g. select only male patients and age > 60. Alternatively the user can select specific features, e.g. age and disease, and request from the system to propose new cohorts. The proposed cohorts can be extracted using clustering techniques along with feature selection algorithms for the identification of relationships in the data. Analysis results are presented to the end user using the same techniques as of step 1.

- **Step 3 (Monitor specific patient):** The user can select one or more patients and plot patient's data in conjunction with average values of specific cohort(s) in order to highlight and identify deviations.
- **Step 4 (Alerts):** Finally, the user can select and create a new alert for specific cohorts (from step 1 or 2) or specific patients (step 3). The user is able to select features and add an automatic alert/indication. The system will monitor the specific feature(s) for the selected cohorts and if significant differences appear the researcher is informed via email.

As we can see from the high level architecture, the main components of the smarts analytical framework of the iManageCancer project are three: the query builder, the analysis and the visualization. In this deliverable we describe the query builder and the data analysis, while the data visualization has been described in the deliverable D8.2 "Report on implemented visualization techniques".

2.2.2 For the patient

Apart from the main scenario described above, the work package 8 contributes and supports three more scenarios requested by the end users. The first one is a simple visualization/analysis of the patient data from the patient. Work package 8 created an application inside the iPHR which is able to visualize many features in one chart with the X axis being common and representing the date. Such a visualization provides the possibility to the patient to identify immediately correlations between the features e.g. when a flue appears in the timeline the body temperature rises while when a particular medication is provided the fever/high temperature disappears. Such an example is provided at Figure 2 where the date (X axis) is provided as months (numeric value) from the first event.

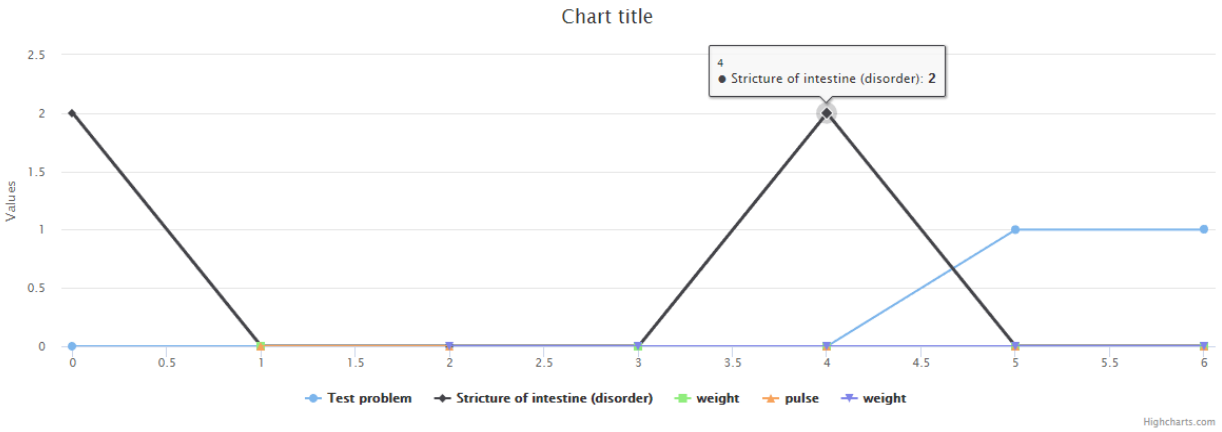

Figure 2: Analytics visualization for the patient

2.2.3 For the administrator

Apart from the patient view, a simple statistics application for the administrator was requested. The administrator of the iManageCancer platform has access to the log files and the log events produced by the applications of the consortium. The mobile applications, the games and the web applications produce a significant amount of event data. Since the work package 8 deals with visualization of statistics in the Query builder, the same visualization component has been extended to support the administrator in the exploration of the audit and log data. A screenshot of the analytics visualization for the administrator is shown at Figure 3, while detailed information about the usage of the filters can be found in the section Query builder (Using filters with the graphical view of the query results). Such a visualization provides the option to the administrator to select one or more applications and identify the modules and the events produced.

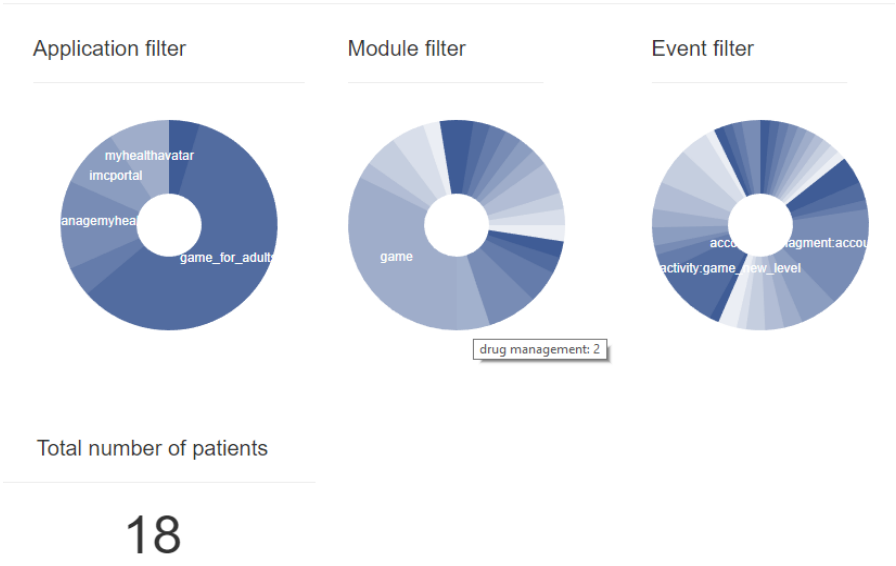

Figure 3: Analytics visualization for the administrator

2.2.4 For the research project

In iManageCancer a research project can request access to specific data for anonymized analysis. A research project account has no access to raw data of the patients. iManageCancer platform allows to execute data analysis tools on the data of a specific cohort of patients who have given consent for the use of their data of a specific research. Everyone can register for a research project with the assumption that this account is created for one specific research project only. The research project account needs to be confirmed by the iMC administrator. The iMC Administrator will need to run the e-consent tool, before confirmation. Once confirmed the researcher (research project account) is allowed to use the data analysis tools on the anonymised health data of the requested cohort. The flow of operations for this scenario is as follows:

1. A research project account requests specific data for anonymized analysis (e.g. a research project would like to have data for patients under 10 years old, male and for the specific analysis at least 100 patients are needed).
2. The administrator has at his/her disposal a tool (Query builder as described in the next section) which can create complex queries in the iPHR databases and get back information such as number of patients for the specific query.
3. If the requested data exist the administrator request access (eConsent) from the eligible patients. The e-consent tool send a request for consent to those patients that match the profile of the research question. The request contain the description of the planned research in layman's language. In the case that a patient has select not to participate at research projects the system will not provide access to that data nor request for eConsent. In the case that a patient gave general consent to donate his/her data for research, the (anonymized) data are available without further request.
4. As soon as the administrator has sufficient approvals for research use from the eligible patients (eConsent), he/she passes the query to the anonymization tool (see section Data anonymization) and the anonymized data are passed to a Virtuoso database with the research id.
5. The administrator informs the research project account (researcher) that can have access to anonymized data from his/her analytics platform.

6. The analytics platform of the research project account provides the full functionality of the Query builder (see section 3), the analytics (see section 4) and the visualization (described in Deliverable D8.2).

2.3 Data

A burden for the implementation of the smart analytical framework was the lack of data. During the timeframe of the implementation the iManageCancer pilots were in the process of preparation and no real data were available. For that reason we generated artificial but realistic data for testing and development of the framework and the algorithms. The artificially generated data contains the basic PHR characteristics that exist in real medical databases such as patients' admission details, demographics, socioeconomic details, labs, medications, etc. The artificial data used provide information for 100 patients with basic demographic data (age, gender, race, and language) and 19000 records for Laboratory Results (CBC: HEMATOCRIT, CBC: RED BLOOD CELL COUNT, CBC: WHITE BLOOD CELL COUNT, METABOLIC: AST/SGOT, METABOLIC: CALCIUM and METABOLIC: CREATININE) at different time points linked with the 100 patients.

The implementation of the framework is modular and easily extendable to other data sources making the linking to the iPHR data straightforward. More details about the iPHR can be found in the deliverable D3.4 "Extended integrated prototype of iManageCancer platform" and the https://www.iphr.care[5]. An API for the data retrieval has been implemented using the standards for all the APIs in iManageCancer (e.g. security OAuth 2.0 tokens). The smart analytics framework of the iManageCancer is accessible only via the iPHR web portal https://www.iphr.care and available only to doctors and research project (upon request).

3 Query builder

The Smart Analytics Framework (SAF) is a web based application and integrates all the components defined by the architecture (**Figure 1**) into a user friendly interface. The SAF provides a collapsible menu to the left where the user can create a new query for the data or view/load existing queries from his/her profile as shown in Figure 4. By default the user views the query builder tab. Query builder is the place where the end user poses the research question and pull the data from the iManageCancer databases. Since the end users are non-IT experts we implemented a graphical interface intended to be simple but yet powerful enough for complex queries. The user actually draws an SQL query with his/her preconditions and selects the attributes that would like to retrieve data from. The implementation is based on graphs and the user has to create/draw a graph where each node is a feature with specific conditions (e.g. the user wants to view data for all the patients with age over 18) while the edges represent the logical condition between the features, e.g. age under 18 AND gender male, while another example could be age under 18 OR age over 67. The query builder provides the possibility to the user to create more complicated queries using groups e.g. (age under 18 OR age over 67) AND gender male.

While the user draws the graph/query the query builder shows the preconditions (similar to the where clause of an sql query) as shown in Figure 4 at the top. Furthermore, a watermark on the query builder area (top left part) provides the query name (if it has been saved by the user) and the number of records/patients that the specific query will retrieve from the iManageCancer databases.

[5] Link to the iPHR manual: https://www.iphr.care/apps/procedures/static/Tutorial.pdf

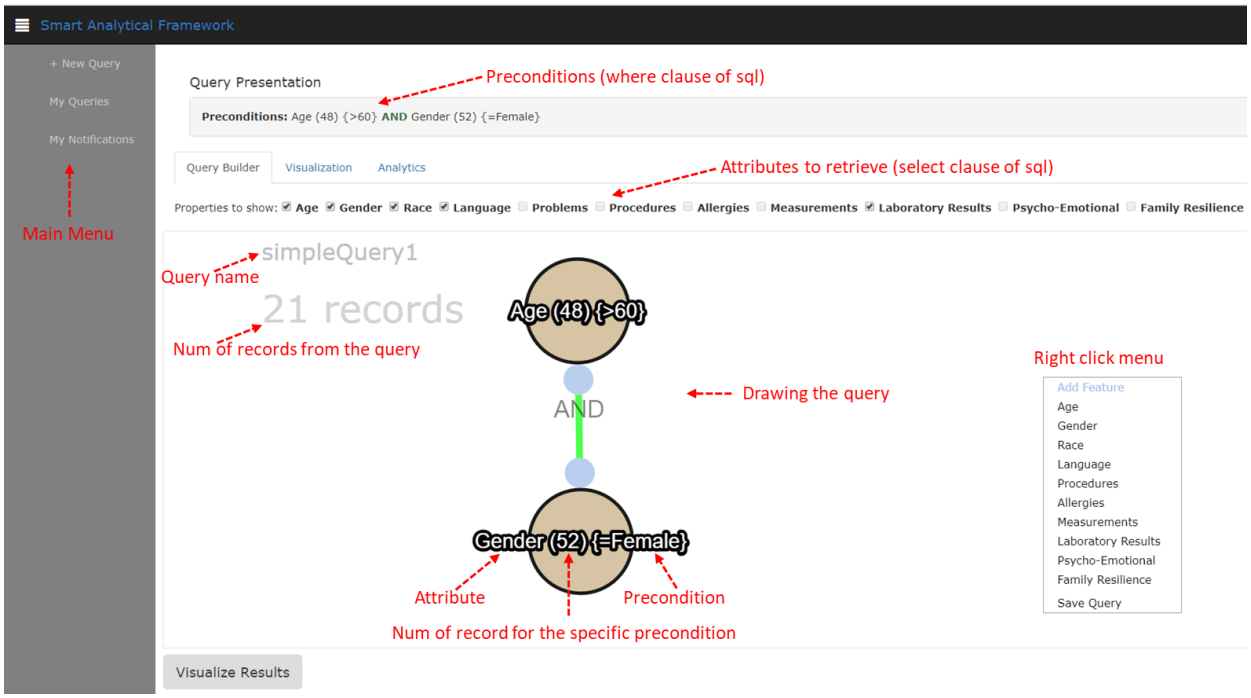

Figure 4: The Query builder user interface.

The user can select a feature of the data model to use by right clicking in the area of the query builder (as it is shown in Figure 4 to the right). Then the system shows a list of options for the preconditions to be used, e.g. for the attribute Age the user can select one of the conditions (=,>,<,>=,<=) and a value. More advanced options appear to attributes with more values like the Laboratory results attribute where the user can select the name of the laboratory results and/or the date and/or value. As soon as the user provides the preconditions and clicks the save button a node is generated in the query builder. The node appears as a circle and the text in it indicates the name of the attribute (left part of the text), the number of records/patients for the specific/single node precondition (middle part of the text) and the precondition in brackets (right part of the text) as it is shown at the bottom of Figure 4. Two or more nodes/attributes can be connected by clicking on the source node and dragging to the target node. The connection can be defined to be the logic AND or the logic OR by right clicking on the connection/edge. Figure 4 shows a query where the user requests all the data from patients with Age>60 and Gender=Female. The first precondition (Age>60) is satisfied by 48 patients in our dataset, the second precondition (Gender=Female) is satisfied by 52 patients, while the whole query (Age>60 AND Gender=Female) is satisfied by 21 patients/records. The information about the number of records/patients satisfied by the attributes and the whole query is provided to the end user while he/she draws the preconditions.

In the example shown in Figure 4, a simple query is presented. The SAF gives the option to the user to create more complex queries including groups and nested groups. Let's assume that a researcher would like to create a query where the patients should be over 70 years old or Female and both of them to have the English mother language. In terms of sql, this would be formulated as *(Age>70 OR Gender=Female) AND Language=English*. The steps for creating such a query with the query builder are shown in Figure 5. Initially the user right clicks and selects the feature *Age*. The popup menu for the feature *Age* provides options to set the appropriate conditions for this field. The user can select the appropriate condition (>) and value (70) as shown in Figure 5A. Then she/he can create a group to nest sequentially the age condition and then the gender condition. To create a group, the user click on the node *Age* and sets a group number. For our example the user sets the number 1, shown in Figure 5 B. The nodes belonging to that group are grouped in the blue rectangle.

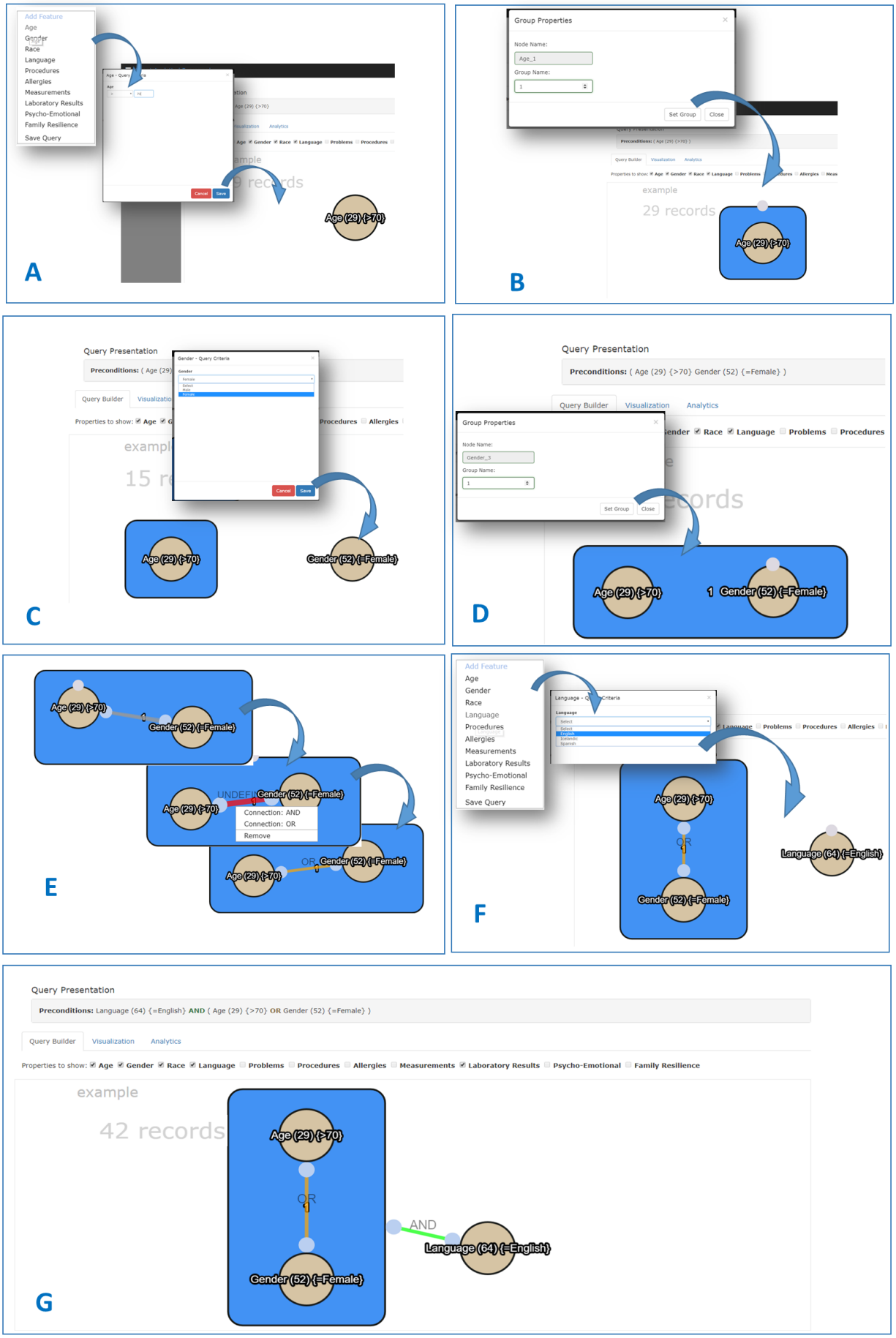

Figure 5: Step by step creation of a (complex) query with the Query Builder.

Then the user creates the precondition *Gender=Female* by right clicking in the blank area of the Query builder and selecting the attribute *Gender*. The user selects the precondition (*=Female*) and the new node is created as shown in Figure 5 C. Next, the node *Gender* should be placed in the group 1 - with the existing node *Age*. The user clicks on the node *Gender* and selects the group label (1) as it is shown in Figure 5 D. The two nodes are in the same blue rectangle and they can be connected. Multiple features can be grouped at any depth level. The next step is to create a link between the two features and set the *(Age>70 OR Gender=Female*). To create a (pre)connection between two features the user has to click on a feature and drag the mouse to the other feature as it is shown in Figure 5 E first part. Then the user can define or remove the connection between the two features by pressing a right mouse click on the edge/connection and select the appropriate connection. Next the user would like to connect with a logical OR (Figure 5 E). Two more steps are needed for the complex query to be completed. The first one is similar to step A and step C where a new feature is added, with its precondition *Language=English* as shown in Figure 5 F. The second one is to link the group with the new feature using the logical AND, similarly to step E, as it is shown in Figure 5 G. Now the query *(Age>70 OR Gender=Female) AND Language=English* has been created using the Query Builder. While the user graphically designs the query, the system informs him/her on the fly about the number of patients that fulfil each independent precondition and the whole query. As we can see from Figure 5 G the precondition *Age>70* retrieves 29 patients, the *Gender=Female* retrieves 52 patients, the *Language=English* retrieves 64 patients, while the query *(Age>70 OR Gender=Female) AND Language=English* retrieves 42 patients.

Another feature of the query builder provides the capability to the user to view more statistics of the generated query at any time and update/modify the query accordingly. The results of each query can be viewed in a graphical way enabling further exploration and enhancement. By pressing the button “View Results” below the query area, charts appear that represent the query results. Each chart can be used as a filter and give instant feedback. The graphical view of the query results from our query *(Age>70 OR Gender=Female) AND Language=English* are shown in Figure 6. Features with numeric values such as *Age* are visualized as bars charts while the nominal features such as *Gender* are visualized as pies. The total number of patients is shown to the right of the viewer area. All the charts can be used as single filters or multiple filters (using the logical AND operation for more than one filter).

Figure 6: Graphical view of query results.

Figure 7 shows the differences when a user selects multiple filters over the results of our query. Figure 7 A shows all the statistics without any filter, Figure 7 B shows the statistics when the users selects a custom range for the feature *Age* (between 50-80 years old). As we can see from the figure, the total number of patients has been affected (from 42 dropped to 21), while the other features (pies) have been adjusted to reflect the updated selection. For example the percentage of *Males* in the *Gender* pie has been dropped down which signifies that for the selected age range (50-80) we have less males percentage in our dataset. Figure 7 C shows statistics of the features when we select only the *Females* in conjunction with the *Age* range 50-80, and Figure 7 D shows statistics of the features when we select *Race=White* in conjunction with the previous filters.

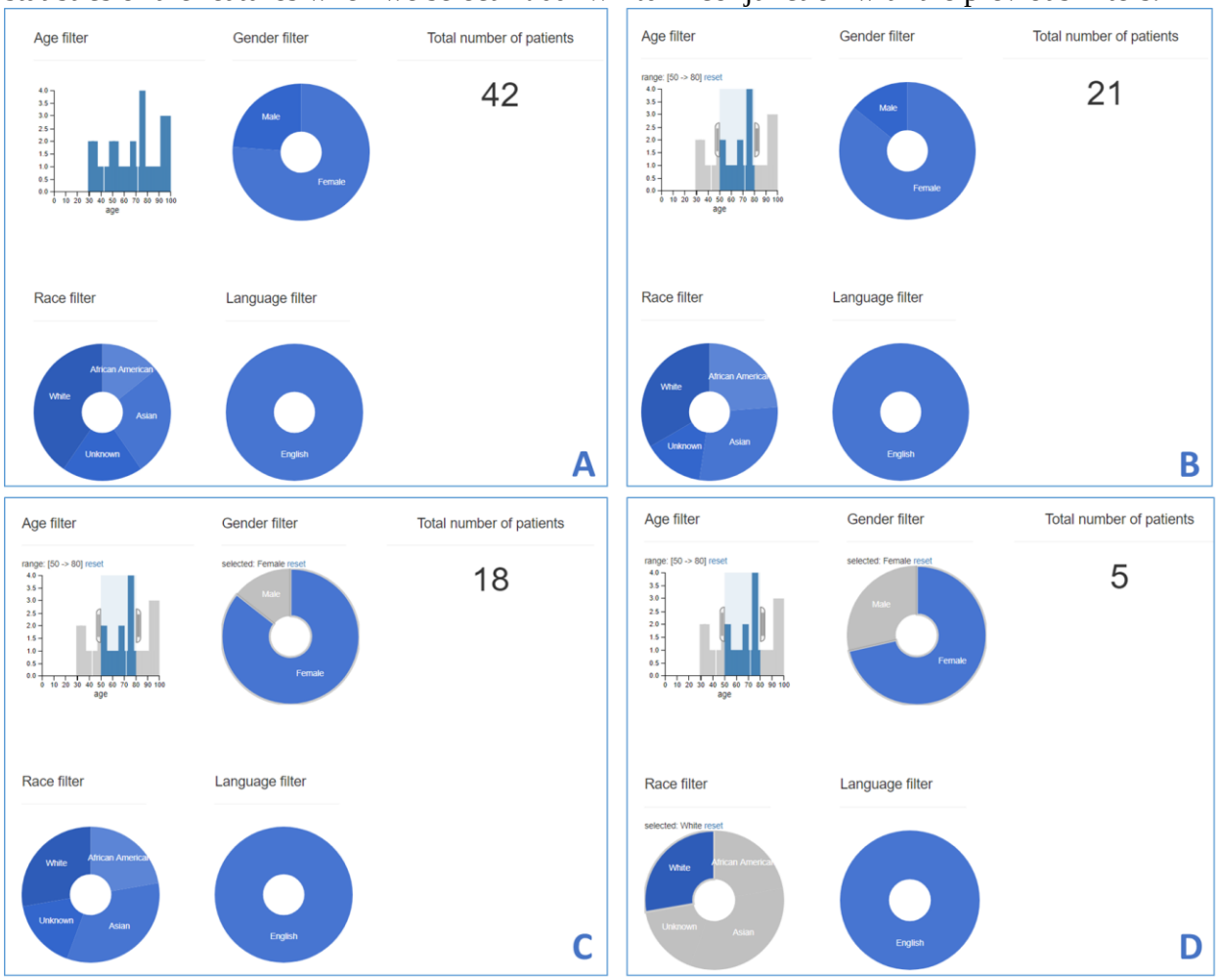

Figure 7: Using filters with the graphical view of the query results.

The implementation of the query builder is based on JavaScript and the Cytoscape graph visualization Web library http://js.cytoscape.org while communication with the Data API is based on Ajax calls. The implementation of the charts is based on the *JavaScript Library for Multi-Dimensional Charting (dc.js).* Dc.js is an open source javascript charting library with native crossfilter support and allowing highly efficient exploration on large multi-dimensional dataset. Dimensional Charting (dc.js) leverages the Data Driven Documents (d3.js) visualization power and the Crossfilter (crossfilter.js) interactive/coordination. The input data in the dc.js are in json format customized and grouped in order to meet the needs of the service.

4 Data analysis

Main objective of the smart analytics framework is to hide the complexity of data mining and statistical algorithms from the end user. The framework supports the well-known K-means clustering algorithm and a feature selection algorithm based on the principal components analysis algorithm.

4.1 Clustering

One of the most important questions in data analysis is to find the "similar" cases/records in our data. If we do not have labels/outcomes in our cases this is an unsupervised learning problem and the method of identifying similar groups of data is called clustering. Clustering is the task of grouping a set of objects in such a way that objects in the same group (cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters). The goal of clustering is to determine the intrinsic grouping in a set of unlabelled data.

K-Means [2] one of the well-known unsupervised learning algorithms, clusters data by trying to separate samples in n groups of equal variance, minimizing a criterion known as the inertia or within-cluster sum-of-squares. The k-means algorithm divides a set of N samples X into K disjoint clusters C, each described by the mean μ_j of the samples in the cluster. The means are commonly called the cluster "centroids"; note that they are not, in general, points from X, although they live in the same space. The K-means algorithm aims to choose centroids that minimise the inertia, or within-cluster sum of squared criterion.

The SAF of iManageCancer provides k-means clustering to the end users. The user has to select the Analytics tab as shown in Figure 8 and set the number of clusters to be created.

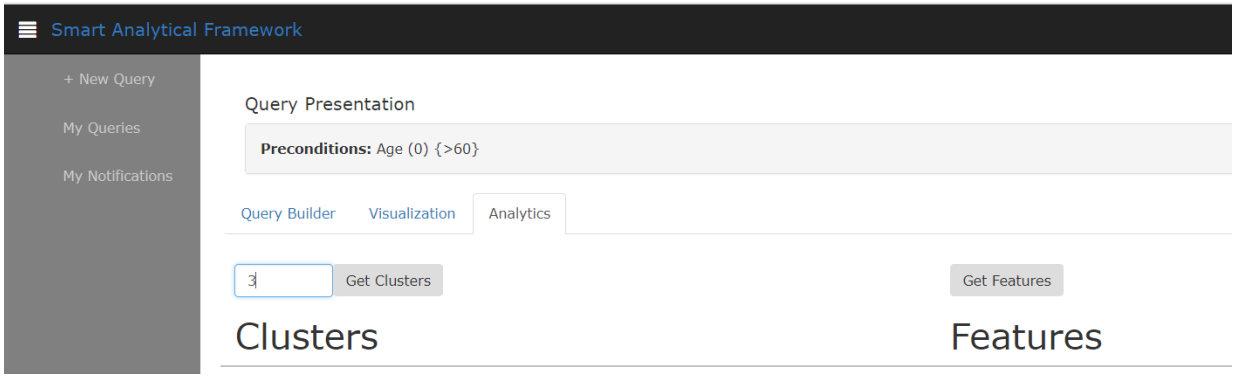

Figure 8: The Analytics tab.

The system makes a call to the backend where the computations take place. The backend has been implemented as a web service in a Tomcat container and gets as input the specific cohort's data with the number of clusters selected by the user. The implemented k-means algorithm is then invoked and the results are passed back to the interface in json format. The results contain list of patient ids per cluster and are visualized for easier interpretation. Details about the visualization of the clusters and the features can be found in D8.2.

4.2 Feature selection

Feature selection is the process of selecting a subset of relevant features (variables). From a computational and statistical point of view, the reduction of the feature set and the selection of the most relevant features could help: (i) to cope with highly dimensional domains and reduce computational cost, and (ii) to improve classification performance. From a theoretical perspective, the selection of the most relevant features employs the min-features inductive bias in which,

simpler prediction models are preferred against more complex ones [3]. In iManageCancer we use feature selection to identify the most informative feature in our dataset.

For that purpose we used the well-known principal component analysis (PCA) algorithm [4]. PCA is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. The idea is that PCA can provide a lower dimension (even one dimension) of features using linear equations (based on the original features). PCA have been used as part of unsupervised feature selection techniques called principal variables [5], [6] and aim to select a subset of variables that contain, in some sense, as much information as possible.

In SAF we followed the principal variables methodology to propose the most informative variables of a cohort to the end user. The user can request from the system to propose and rank the available variables of a cohort (data coming from a specific query). Such information can guide the end user to refine the query for the cohort selection or focus on specific features for the analysis. The feature selection algorithm run in the background and the user can trigger the functionality from the Analytics tab as shown in Figure 8 to the right.

Similarly to Clustering, the system makes a call to the backend where the computations take place. The backend gets as input the specific cohort's data, the implemented principal variables algorithm is then invoked and the results are passed back to the interface in json format. The results contain list of features with a weight that represents how informative is each feature. Details about the visualization of the features can be found in D8.2.

5 Alerts

Part of the smart analytical framework is the ability of the user to create notifications. The notifications are based on the implemented queries. The user can create a notification in order to be notified if the query has results (e.g. notify me when a patient has strep throat) or if the query has certain results based on features' values (e.g. notify me when a patient has fever over 39) or count of the results returned (e.g. notify me when more than 100 patients exist with breast cancer and are aged over 45).

The user can view, create or check his/her notifications from the left menu of the SAF (by selecting the "My Notifications" menu item). The page of "My Notifications" menu shows a list of the notifications that have been already created by the user as shown in Figure 9.

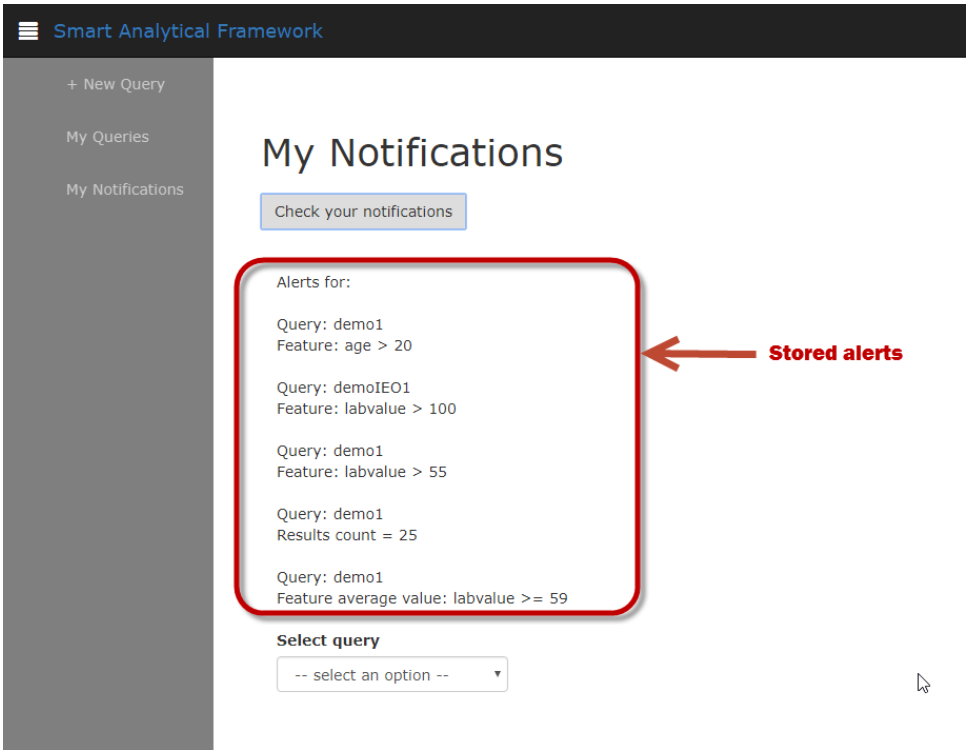

Figure 9: Notifications user interface

The user can create new notifications based on his/her queries. User's queries are shown in the dropdown menu "Select query" as shown in Figure 10.

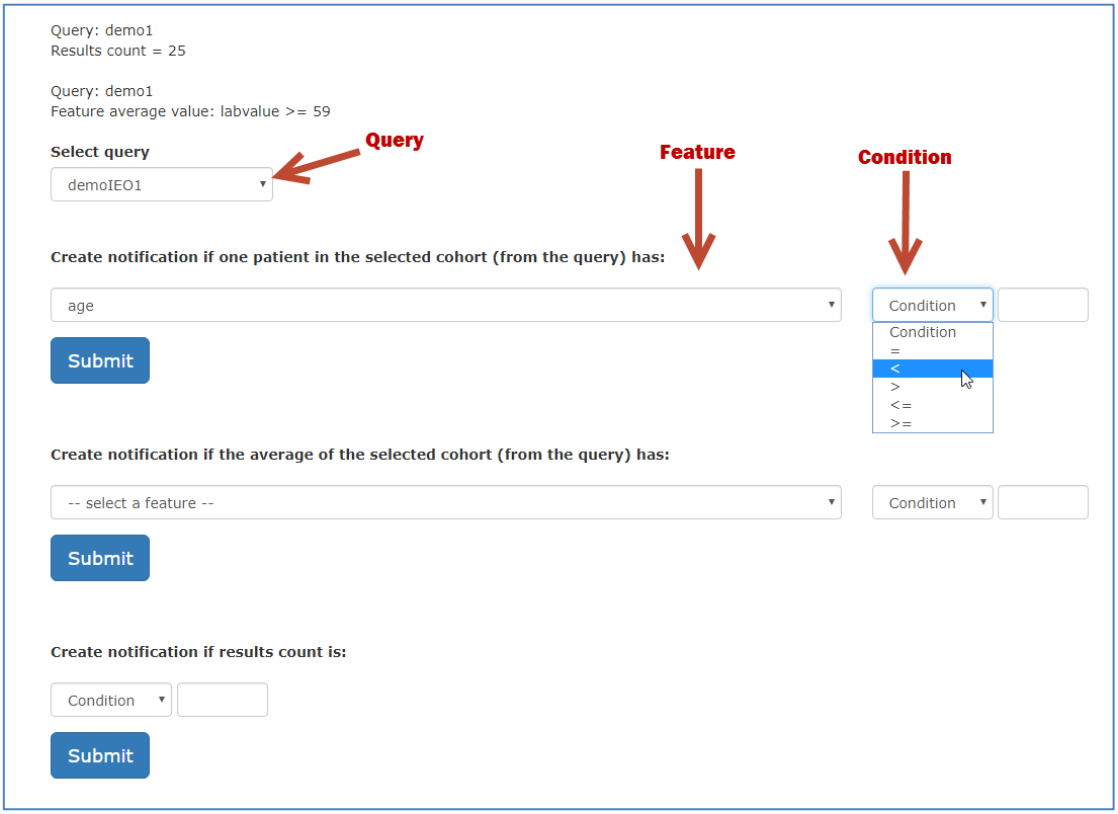

Figure 10: Create a new notification

The user can select a query and based on the queries' features, new dropdown menus, such as selecting features and condition, appear in order to create the new notification. The user can create

notification for a feature (e.g. is equal, greater, greater than, less or less than a value). Another option is to create a notification concerning the average value of a feature, again based on the selected cohort of the query, with conditions greater, greater than, less, less than and equal. And finally, a notification could be created based on users' counts (e.g. if the result of the query covers number of patients than a value).

The created alerts are saved in the system and every day the system is checking if the conditions are met. If a condition is met the system is sending a notification email to the registered user and the notification is deleted from the system. The user could also select to be notified every day and keep the notification saved in the system.

6 Data anonymization

Apart from the analysis, another objective of the WP8 is to provide a complete data anonymization toolkit and give the possibility to the user to work with or analyze anonymized data. For data anonymization iManageCancer is using the ARX data anonymization tool [7]. The whole process is shown in Figure 11. The administrator using the data analysis tool identifies whether the eligible number of patients have been collected. As soon as all conditions are satisfied he/she decides to release the available data. In order to do that the data to be released are accessed by the ARX-Data Anonymization Tool, de-identified and loaded to the Virtuoso Triple store in a personal space for the researcher.

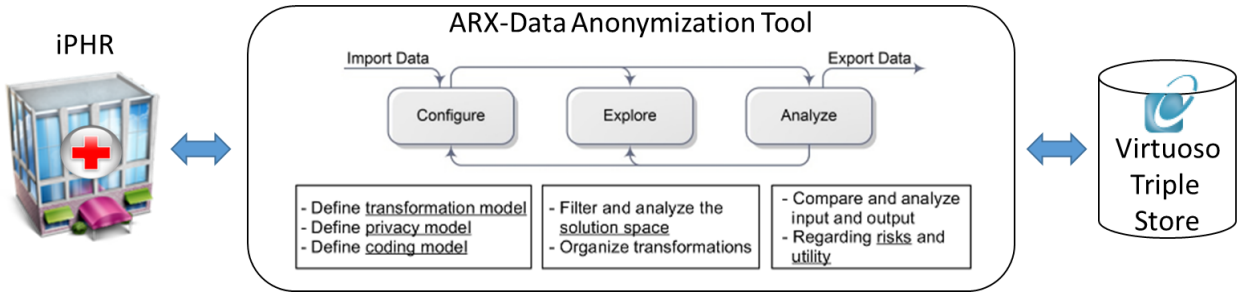

Figure 11. Workflow of the anonymization process

The whole process is automatic and transparent to the administrator and is executed using the ARX APIs and the Virtuoso APIs. However in order to achieve the aforementioned result the ARX-Data Anonymization tool has been appropriately configured.

ARX configuration:

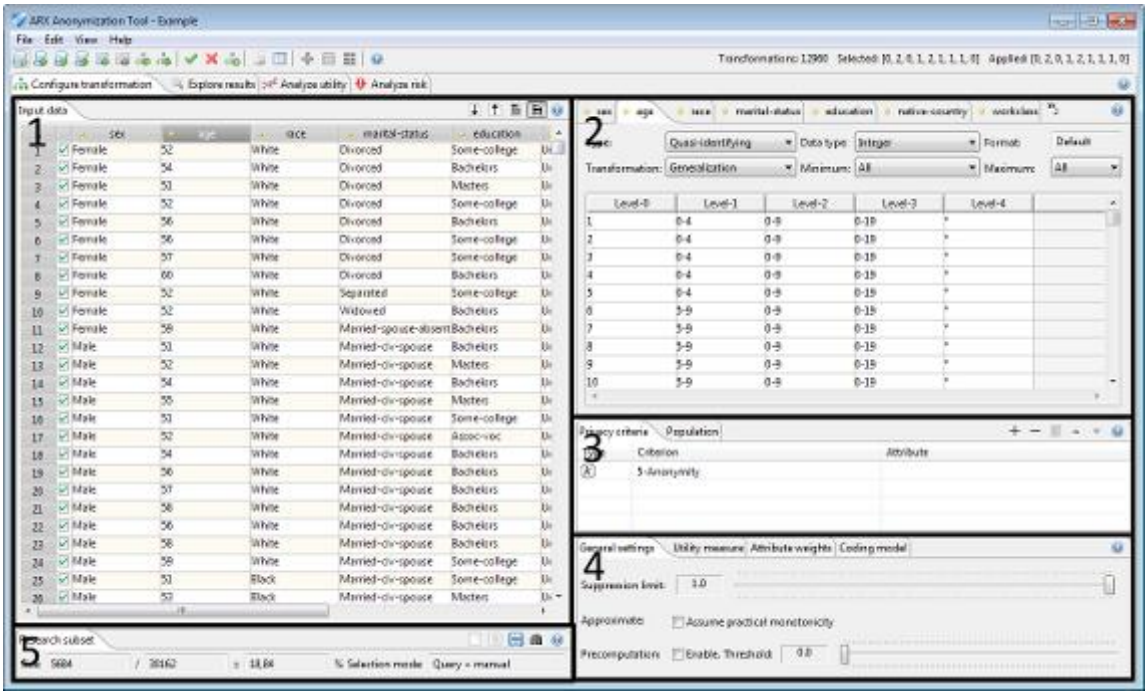

Figure 12. Using ARX tool to configure the tool.

As such the transformation, the privacy and the coding models should have been configured a priori. Configuring encompasses exploring the solution space, analysing input and output data and analyse risks and utility. All configuration parameters have been set before executing the aforementioned workflow considering the different anonymization models and the input data using the provided GUI shown in Figure 12.

The GUI shown is divided into five sections:

- Section 1 shows the current input data set
- Section 2 provides means for specifying meta data about its attributes
- Section 3 supports configuring the privacy model
- Section 4 implements controls for configuring further properties of the transformation process, such as the coding model, how data utility should be measured and how important certain attributes are
- Section 5 provides methods for extracting a research sample, which is a subset of the overall data set that is to be de-identified and exported. As soon as the privacy model for each column/table is defined all parameters are saved into a configuration file which is later used by the APIs for executing the workflow presented in Figure 11.

Privacy models:

The main privacy models that are implemented in the ARX data anonymization tool that are exploited by the iManageCancer consortium are the following

- *k-Anonymity*: This is a well-known model aiming at protecting datasets from identity disclosure following the prosecutor attacker model [8]. A dataset is k-anonymous if, regarding the quasi-identifiers, each data item cannot be distinguished from at least $k - 1$ other data items. The tuples with identical values for all quasi-identifiers form an equivalence class. A variant of k-anonymity also implemented by ARX is *k-Map*.

- *Population & Sample uniqueness*: ARX also supports several relaxed privacy models for protecting datasets against re-identification attacks following the marketer model. For example, thresholds can be enforced of the proportion of records that are unique within the underlying population[9]. ARX also implements a privacy model which restricts the fraction of records that are unique within the dataset.
- *Strict-average risk*: ARX also implements strict-average risk, which is a combination of a threshold on average re-identification risks combined with k-anonymity. It can be used to protect datasets from marketer attacks.
- *ℓ-Diversity*: This privacy model [10] protects a dataset against attribute disclosure. It ensures that the values of a set of predefined sensitive attributes are at least l-diverse within each equivalence class. l-Diversity also implies l-anonymity. To fulfil the basic definition of l-diversity, a sensitive attribute must have at least ”well represented” distinct values in each equivalence class. Different variants, such as entropy-l-diversity and recursive-(c,l)-diversity, have been proposed, which implement different measures of diversity. It was shown that recursive-(c,l)-diversity delivers the best trade-of between data quality and privacy.
- *t-Closeness*: This privacy model [11] is an alternative for the protection against attribute disclosure. The basic idea is that equivalence classes are not allowed to stand out in the dataset. To achieve this, the distributions of the values of the sensitive attribute within each equivalence class must have a distance of less than *t* to the distribution of the attribute values in the original dataset. For measuring distances between distributions, the earth mover’s distance (EMD) is used. Different variants exist, which use different ground distances when computing the EMD: (1) equal ground distance, which considers all values to be equally distant from each other, and, (2) hierarchical ground distance, which utilizes generalization hierarchies to determine the distance between data items.

7 Conclusions

The data analysis and data mining tools aim to extract information from the diverse data of the healthcare domain and transform it into an understandable structure for better knowledge discovery. Smart data analytics provide mechanisms able to identify patterns or trends in data, screen pre-frailty states and set different views of data for new management plans. Analysis tools can be used for extracting new knowledge, by the effective integration of data mining and expert knowledge. Visual analytics make use of information from iManageCancer data sources, and bring together valuable information in visual form to support exploration (information about the visualization techniques for the smart analytics can be found in the public deliverable/demonstrator D8.2 of iManageCancer). Such a system is expected to successfully overcome the limitation of traditional intelligent data analysis that works only with a small number of well-defined and well trained cases.

Apart from the knowledge discovery capabilities, WP8 contributes and supports a simple visualization/analysis of the patient data from the patient view and a simple statistics application for the administrator (statistics for log files and audit events). Furthermore, the eConsent scenario of iManageCancer (some of the modules are still under development in WP3 such as the eConsent communication tool) uses the data anonymization tool developed by WP8 and the smart analytical framework over anonymized data for data analysis.

The implemented data driven tools analyse the information in the iManageCancer database and draw conclusions related to the usage of the self-management platform, reported adverse events and several health issues. Smart analytical services aims to provide physicians a global view of

the end users data and monitor the evolvement over time. Furthermore, the platform provides specific services for data-driven analysis on anonymised clinical information for public health research.

8 References

[1] F. Frankel and R. Reid, "Big data: Distilling meaning from data," *Nature*, vol. 455, no. 7209, pp. 30–30, 2008.

[2] J. B. MacQueen, "Kmeans Some Methods for classification and Analysis of Multivariate Observations," *5th Berkeley Symp. Math. Stat. Probab. 1967*, vol. 1, no. 233, pp. 281–297, 1967.

[3] H. Almuallim, H. Almuallim, T. G. Dietterich, and T. G. Dietterich, "Learning With Many Irrelevant Features," *Proc. Ninth Natl. Conf. Artif. Intell.*, vol. 3, no. Quinlan, pp. 547–552, 1991.

[4] H. Hotelling, "Analysis of a complex of statistical variables into principal components.," *J. Educ. Psychol.*, vol. 24, no. 6, pp. 417–441, 1933.

[5] G. P. McCabe, "Principal Variables," *Technometrics*, vol. 26, no. 2, pp. 137–144, 1984.

[6] R. W. Swiniarski and A. Skowron, "Rough set methods in feature selection and recognition," *Pattern Recognit. Lett.*, vol. 24, no. 6, pp. 833–849, 2003.

[7] F. Prasser and F. Kohlmayer, "Putting statistical disclosure control into practice: The ARX data anonymization tool," in *Medical Data Privacy Handbook*, 2015, pp. 111–148.

[8] L. Sweeney, "k-ANONYMITY: A MODEL FOR PROTECTING PRIVACY," *Int. J. Uncertain.*, vol. 10, no. 5, pp. 557–570, 2002.

[9] F. Dankar, K. El Emam, A. Neisa, and T. Roffey, "Estimating the re-identification risk of clinical data sets," *BMC Med. Inform. Decis. Mak.*, vol. 12, no. 1, p. 66, 2012.

[10] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "ℓ-Diversity: Privacy beyond k-anonymity," in *Proceedings - International Conference on Data Engineering*, 2006, vol. 2006, p. 24.

[11] L. Ninghui, L. Tiancheng, and S. Venkatasubramanian, "t-Closeness: Privacy beyond k-anonymity and ℓ-diversity," in *Proceedings - International Conference on Data Engineering*, 2007, pp. 106–115.