Correlation Analysis Cell: User’s Guide

Version 1.0

Table of Contents

About this Guide

Informatics for Integrating Biology and the Bedside (i2b2) is one of the sponsored initiatives of the NIH Roadmap National Centers for Biomedical Computing (http://www.bisti.nih.gov/ncbc/). One of the goals of i2b2 is to provide clinical investigators broadly with the software tools necessary to collect and manage project-related clinical research data in the genomics age as a cohesive entity—a software suite to construct and manage the modern clinical research chart.

This *Guide* provides users of the *Correlation Analysis Cell* with common use-case scenarios. This specialized analysis cell uses mutual information theory to calculate observed correlations within the data of the hive. This type of cell represents an important achievement of the hive.

It is assumed one has installed and configured the *i2b2 Workbench* and the *Correlation Analysis* module, either on source-code or binary distribution level. If this is not the case, please consult Section 1 of this document, *Prerequisites and Third-Party Software*.

Document Version History

Date	**Revision**	**Description**	**Author(s)**
June 20, 2008	version 1.0	Initial revision, 1.0	Vlad Valtchinov
July 14, 2008	Version 1.0	Small corrections	Vlad Valtchinov
Aug 05, 2008	Version 1.0	Clarifications, formatting	Vlad V

1

Prerequisites and Third-Party Software

Downloads and Installation

a. i2b2 Workbench version 1.2.1

Download i2b2 Workbench version 1.2.1 (i2b2Workbench-src-121.zip) from https://www.i2b2.org/software/repository.html?t=demo&p=14. Follow installation and configuration instructions as given in the *i2b2 Workbench Developers' Guide v1.2.1* which can be found under the Docs tab.

b. Java JDK 5.0 – needed for i2b2 Workbench

This version of the DJK is needed for running the Eclipse Workbench. Download JDK 5.0 Update 11 (jdk-1_5_0_11-windows-i586-p.exe) from http://java.sun.com/products/archive/

Run the installer. Setup JAVA_HOME and CLASSPATH environment variables after installation.

c. Eclipse

You will need to use version 3.2.1 of the Eclipse SDK (eclipse-SDK-3.2.1-win32.zip), which can be found at http://archive.eclipse.org/eclipse/downloads. If you install Eclipse, be sure to install it in an area separate from any previous Eclipse installations.

To install, extract the zip file into a directory on local disk. Create a local desktop shortcut to eclipse.exe.

d. yEd Graph Editor

The Correlation Analysis Cell uses the yWorks' yED Graph Editor for viewing and editing Relevance Networks graph files. Download the most recent version for your platform available at http://www.yworks.com/en/products_yed_about.html. After installing, make sure the GRAPHML file format is opened by default by yEd.

2

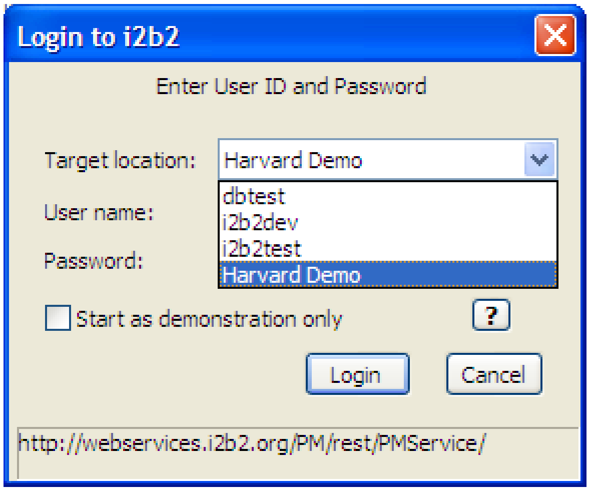
Login and i2b2 Data Mart Selection

Login

Starting the *i2b2 Workbench* with the *Correlation Analysis Cell* installed and configured will first prompt for an i2b2 Data Mart selection and corresponding credentials.

The choice "Harvard Demo" corresponds to a Demo i2b2 Data Mart containing curated data from the public archive of the NCBI site, the Gene Expression Omnibus (GEO) in addition to the Asthma data.

Login with credentials supplied to you at the i2b2 web site or contact support for obtaining these.



Login to i2b2

Enter User ID and Password

Target location: Harvard Demo

User name:

Password:

□ Start as demonstration only

http://webservices.i2b2.org/PM/rest/PMService/

3

Correlation Analysis: An Overview

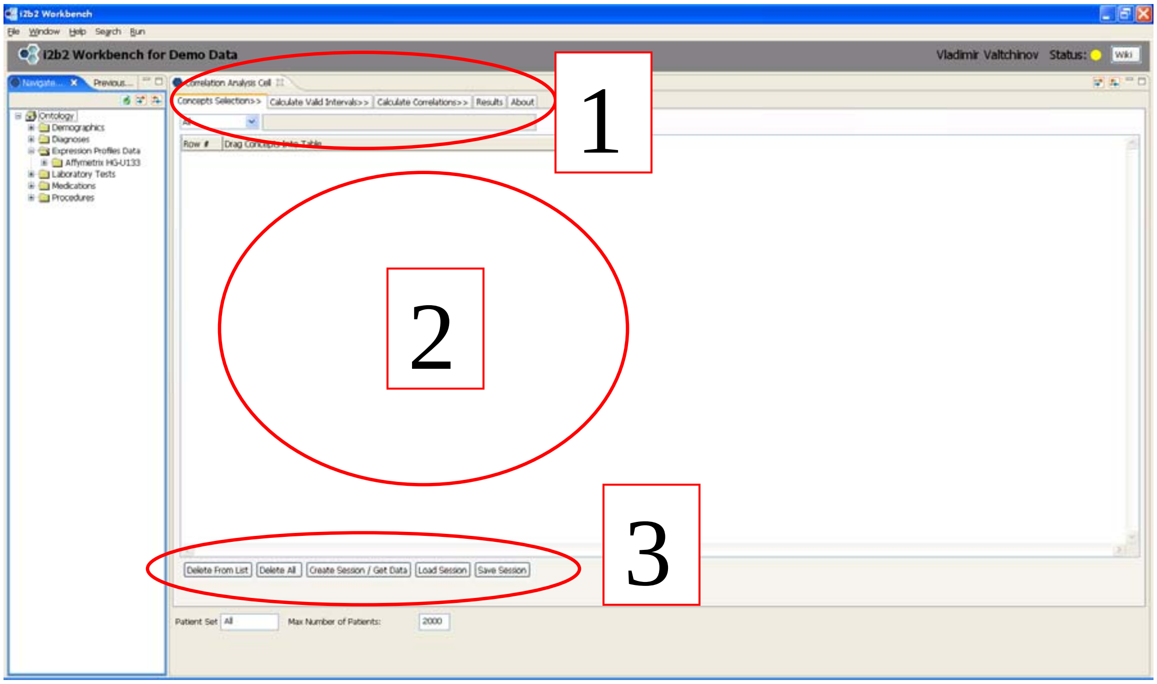
The Correlation Cell is one of the first specialized analysis cells from the i2b2 hive. It uses model-less data mining and clustering techniques based on direct pair-wise correlation calculation. The most common ‘similarity measures’ used are the Mutual Information Content (MIC), the *Pearson’s* linear correlation coefficient, and the Euclidean distance between the vectors. Both time-independent and time-resolved versions of the main algorithm were developed which makes the general approach extremely well-suited for applications for analysis of medical data (like diagnostic labs, diagnoses, medications etc.) as well as various “omic” data sets (genomics, proteomics, etc.).

An additional advantage of the main algorithm is its ability to be applied to finding correlations among homogeneous data types (i.e. labs-labs) as well as among heterogeneous data types (i.e. labs-diagnoses), with a suitable extension of the similarity measures. This very feature allows clinical records data to be co-analyzed together with generally the more quantitative genomics data – a fact that could potentially lead to generation of novel insights and hypotheses for further, more focused investigations.

The *Correlation Analysis* plug-in is a utility fully integrated within the i2b2 hive to analyze and display hidden correlations between concepts. For a review of the exact theoretical foundation of the Correlation Analysis Cell please consult Chapter *Relevance Networks: Theory and Implementation* later in this *User’s Guide*.

Please review the *Correlation Analysis Cell Tutorial* document for a step-by-step Tutorial including a detailed walk through the actual calculations being performed.

Upon starting, you should see an *i2b2 Workbench* layout similar to the one shown below. If a different layout is displayed (or if this is the first time displaying *Correlation Analysis Cell* within the *i2b2 Workbench*), please arrange the self-docking windows of the plugins to their desired positions.



Patient Set All

Max Number of Patients: 2000

The *Ontology Management (OM)* and the *Correlation Analysis Cells* are displayed side-by-side to facilitate the process of selecting (via drag-and-drop) of concepts to be analyzed for correlations.

Area "1" circled in the figure above shows a series of tabs in the *Correlation Analysis Cell* that represent the main succession of steps in using the Cell. The arrangement of the top tabs in the GUI is deliberate and is intended for guiding the user during the process in a wizard-like fashion. These top level tabs are:

a) "Concepts Selection>>"
b) "Calculation of Valid Intervals>>"
c) "Calculate Correlations>>": options and available algorithms
d) "Results>>": display and analyze results: both for valid intervals and correlation calculations

The area designated as "2" on the screen shot above is where the selection of concepts takes place, via drag-and-drop from the *Ontology Management Cell*. Generally, the *Correlation Analysis Cell* is capable of calculating the following type of correlations between these concepts:

a) Intra-types correlations (among homogeneous concept types):

- diagnostic labs to diagnostic labs
- diagnoses to diagnoses
- gene expression to gene expression (if expression data is available)

b) Inter-types correlations (among heterogeneous concept types):

- diagnostic labs to diagnoses
- diagnoses to gene expression
- diagnostic labs to diagnoses

The area numbered “3” is where most of the concepts list manipulation, data imports and session management functionality takes place.

4

Correlation Analysis Workflow

The overall workflow in using the *Correlation Analysis Cell* is as follows.

One would start with defining the lists concepts for analysis. The choice of concepts to get correlated is guided purely by user's own investigative agenda. As the amount of data available in the Data Mart (across patient populations – regular or de-identified depending on the particular IRB, and for a given clinical context) will determine the applicability of the selections to the specific questions at hand, one should be aware of the type and amount of data behind each of the login options, see Chapter 2 *Login and Data Mart Selection* earlier in this *User's Guide*.

Two main modes of compounding the data for a selected concept can be used. The algorithms in the *Correlation Cell* will use data from all patients by default. An option exists to limit this process to only patients in a pre-defined list (patient lists or patient cohorts). The inclusion of this finer selection is dictated by the ability to address many interesting questions related to defining cohorts and considering specific clinical and other data pertinent to them.

In general, one can either define lists of concepts that are of a single type or – perhaps of more potential interest, include multiple such types in a selection. The possible cross-correlations were previously defined.

The computation of the *valid intervals* is the next step in the analysis. The *overlap in time* and the *valid intervals* are required for all intra-correlations of diagnostic labs and diagnoses as well as for their cross-correlations. When expression data is correlated with itself the algorithm doesn't use *time overlap*; it however does when expression data is cross-correlated with labs or diagnoses.

A final stage of the computation is the calculation of the pair-wise correlation coefficients. One can use the valid intervals (when needed) from the manipulation during the last step, or can set valid intervals manually. Chronic conditions (i.e. diagnoses) could automatically be assigned infinitely long valid intervals.

One can review the results, for both *valid intervals* (sometimes interesting in their own merit as they would generally reflect ordering patterns for the data against which the analysis is run) and the correlations. The results can be analyzed using multiple search criteria. The resulting data sub-sets can be graphically visualized to show the Relevance Networks.

Finally, use session management tools to save either complete sessions (selected concepts, computation results and graphs) or arbitrary session parts for later use and comparison.

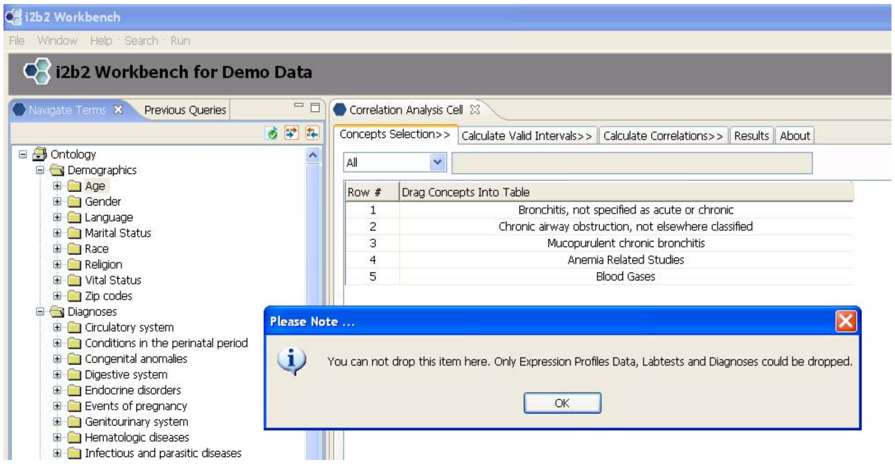
5

Concepts Selection and Session Management

When the *Correlation Analysis Cell* is started it displays the “Concept Selection>>” tab by default. The active data area below the title of the tab allows for drag-and-drop selection of concepts to occur.

Go to the *Ontology Management* (OM) window; navigate to the desired branch of the ontology and drag-drop a concept. Use the **Control** key or the **Shift** key with a range for multiple selections.

The list will get automatically enumerated when dropped in the data area of the Concept Selection tab, see figure.



Row #	Drag Concepts Into Table
1	Bronchitis, not specified as acute or chronic
2	Chronic airway obstruction, not elsewhere classified
3	Mucopurulent chronic bronchitis
4	Anemia Related Studies
5	Blood Gases

You can not drop this item here. Only Expression Profiles Data, Labtests and Diagnoses could be dropped.

Only concepts of following types are currently allowed by the plug-in:

i) Diagnoses
ii) Laboratory tests
iii) Expression profiles data.

If one attempts to select another type, a warning message will be displayed (e.g. if one attempts drag-dropping Age from the Demographics sub-ontology the warning will appear, see the message in the screen shot above).

The number of concepts defined in this process will determine the number of pairs (cross-correlation pairs between the defined concepts) the Cell will pre-calculate further.

One can manipulate the concepts list by using the *Delete from List* and *Delete All* buttons below the active data area. These buttons’ actions are self-explanatory.

Session Management

The *Correlation Analysis Cell* provides a utility to save correlation results and analyses from the current session.

Session Creation
Upon defining the number of concepts for analysis, one has to create session and load raw data from back-end Data Mart. Use the *Create Session/Get Data* button. If no session is created upon completion of concepts' selection, all subsequent analysis menus will be unavailable.

Open Session and *Save Session*
These buttons allow for session management and persistence. *Open Session* displays an *Open File* window to navigate to an XML-formatted session file. *Save Session* saves the current session results (but no input data) to a local file for later use.

Patient Sets

By default, all patients will contribute to the data set when a concept is selected for analysis. One can restrict this set by defining a Patient Set. Usually, a pre-defined patient set from the *Previous Queries* window can be drag-dropped in the *Patient Set* active area. This functionality allows data from only specific patient cohorts to be analyzed for hidden correlations.

To revert back to the default behavior of including data from all available patients, type “All” in the *Patient Set* field and click *Create Session / Get Data* button.

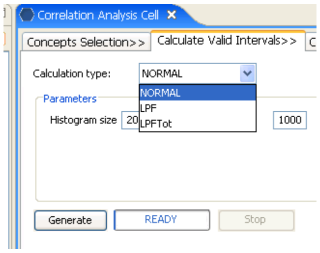
6

Valid intervals Calculation

Valid intervals need to be computed as a first step in the correlation calculation. This allows for constructing the numerical vectors for computing the correlation out of generally temporal data, i.e. when the two entities whose correlation is considered have been measured in different time moments. This is the case when dealing with cross-correlations of diagnoses and diagnostic labs for example. Please also consult Chapter *Relevance Networks Theory* later in this *User's Guide*.

Valid Intervals Calculation Types
Three main algorithms for calculating *valid intervals* were implemented at the time of the first Correlation Analysis Cell release (see screen shots below).

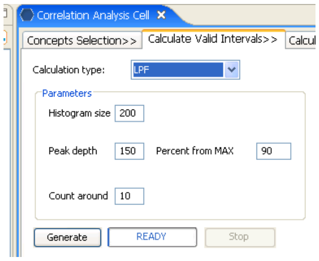
Normal computation
Compute two successive histograms (of default sizes 200 and 1000 respectively) of all time intervals for a given concept. Retain the bins with maximum count after the two binning procedures.



Calculation type: NORMAL

Histogram size 20 1000

LPF ("Low-Pass Filter"-like algorithm)
This implements a sliding window algorithm on the time intervals histogram for a given concept. The peak and its surrounding peaks need so satisfy some smoothness requirements to be considered as candidates.



Calculation type: LPF

Histogram size 200

Peak depth 150 Percent from MAX 90

Count around 10

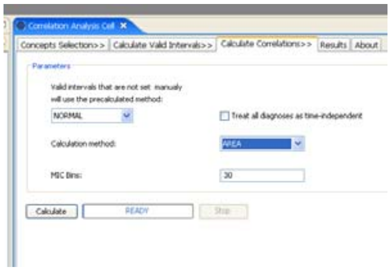
LPFTot
This is a generalization of the *LPF* algorithm with the additional requirement of maximization of the spectral power under the histogram curve. Default parameters are very similar to the *LPF* case.

7

Pair-wise Correlation Computation

With the *valid intervals* already calculated, one next computes the actual pair-wise correlation coefficients. First step for the algorithm is to enumerate all possible pairs between selected concepts. The total number of pairs is constructed as follows. For similar types, number of pairs is N (N-1)/2, self-correlations are excluded. For cross-type correlations, one would have a similar dependence, N1xN2, where N1 and N2 are the corresponding numbers of concepts in case of 2 types.

The *Correlation Analysis Cell* implements 3 different recipes for computing the pair-wise correlation coefficients. They correspond to different ways of factoring in each datum’s value, together with various ways of modeling the probabilistic expiration of validity at a given time point. Their exact mathematical definitions are given in the Chapter *Relevance Networks Theory and Implementation* later in this *Guide*. The recipes are arbitrarily called



Valid intervals that are not set manually will use the precalculated method: NORMAL

☐ Treat all diagnoses as time-independent

Calculation method: AREA

MIC Bins: 30

- *i)* *normal,*
- *ii)* *area,*
- *iii)* *area-only.*

These methods were designed to take into account the specific information about the concept (i.e. its value or the value and time-stamp) together with various ways to ‘expire validity’ of a time-resolved measurement and thus can give slightly different results for the numerical correlation coefficients. Often when only cross-types correlations are calculated specific time-overlap procedures make sense or are used by default.

For example, if a pair contains both concepts of the type “diagnostic labs” (i.e. having a numerical value, time of measurement and normal range associated with them), the correlation algorithm can use *normal*, *area* or *area-only* time-overlap algorithms.

Similarly, when a diagnose-to-diagnose correlation is assumed, there can only be an *area-only* time overlap, related to the fact that a diagnose can be regarded as having a value of 1 together with the time stamp.

When both members of a pair are concepts of type "expression data" the algorithm assumes no *time overlap* is necessary and proceeds to order the data across patients and directly compute the MIC and Pearson numerical coefficients.

The *area* and *area-only* time-overlap recipes seem to performs a bit better than *normal*. This seems to be due to the fact that under these two schemes the contributions to the vectors correlation are a continuous function of the "amount of time overlap". This is opposed to the *normal* algorithm's case of constructing the overlap vectors as binary, an "on-off" type of contributions. No specific recommendations about one or another algorithm however are made here as no clear comparison metric is defined.

The table below lists all possible inter- and intra-correlations and the corresponding time overlapping procedures defined for them (N (normal), A (area), A-O (area-only)):

	Labs	**Diagnoses**	**Expression Data**
Correlate With:			
Labs	N, A, A-O	A, A-O	A
Diagnoses	N, A, A-O	A-O	A-O
Expression data	A	A-O	No Time-Overlap

There are default choices for the cases of concept list containing alike types. The algorithm will usually select the *normal* method (whenever available) to correlate these, except for the case of inter-correlations of expression data. For the latter case no time overlap will be performed, as previously specified. When however multiple concepts types are selected for correlation analysis the program will make default choices in performing all possible cross-correlations between N items of disparate types.

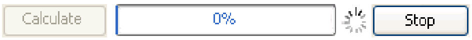
The user is encouraged to explore these choices and develop their own insight in which is the most suitable method for the problem at hand.

Following section explains selections of some specific parameters in a typical analysis session.

First select the type of valid interval to be used in the time-overlapping portion of the correlation calculation. Possible choices are *normal*, *LPF* and *LPFTot*, their corresponding meanings are explained in this text above. By default, the algorithm will use the *normal* valid intervals, if not set manually. To find which type of valid intervals are calculated and can be used or manually pre-set some valid intervals use the *Results->Valid Intervals* tab.

Select also number of bins to be used with the Mutual Information Content (MIC) calculation. A meaningful default of 30 is pre-set and can be used for most of the cases.

Check the *Treat all diagnoses as time-independent* check-box to automatically make all diagnoses have an 'infinitely-long' *valid interval* (technically set to be around 40 years). This setting will make each concept of type "diagnose" to be considered "a chronic condition" during the overlap portion of the calculation, thus producing potentially much longer vectors for the correlation computation.



Click *Calculate* when all the selections are specified. A progress bar together with an activity indicator (the little rotating wheel immediately to the right) will appear. The *Stop* button can be used to interrupt longer calculations. A single-pair thread can not be interrupted and it can create an illusion of 'stalled application'.

The JVM configuration and specifically the amount of available memory play an important role in determining how large computational jobs can be run using the *Correlation Analysis Cell*. For discussion of these consult the *Correlation Analysis Developer's Guide* or the *Installation Guide* available from the i2b2 web site.

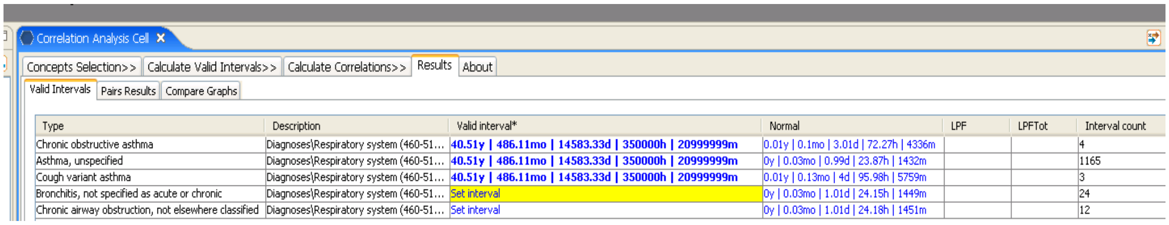
8

Results Viewing, Analysis and Graphing

The *Results* tab of the application contains utilities to review *valid interval* and correlation pairs results. Use this tab to access utilities for graphing various data distributions and charting RelNets. The tab also contains utilities for graph comparisons.

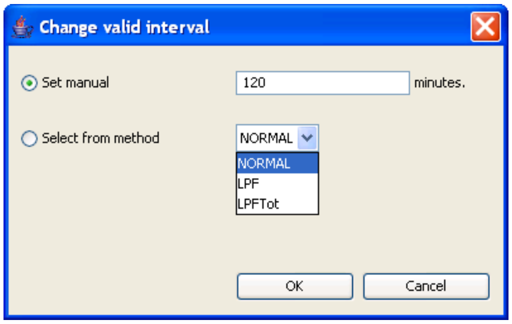
Valid intervals results viewing

All calculated valid intervals are shown in the *Results->Valid Intervals* tab.



Type	Description	Valid interval*	Normal	LPF	LPFTot	Interval count
Chronic obstructive asthma	Diagnoses\Respiratory system (460-51…	**40.51y | 486.11mo | 14583.33d | 350000h | 20999999m**	0.01y | 0.1mo | 3.01d | 72.27h | 4336m			4
Asthma, unspecified	Diagnoses\Respiratory system (460-51…	**40.51y | 486.11mo | 14583.33d | 350000h | 20999999m**	0y | 0.03mo | 0.99d | 23.87h | 1432m			1165
Cough variant asthma	Diagnoses\Respiratory system (460-51…	**40.51y | 486.11mo | 14583.33d | 350000h | 20999999m**	0.01y | 0.13mo | 4d | 95.98h | 5759m			3
Bronchitis, not specified as acute or chronic	Diagnoses\Respiratory system (460-51…	Set interval	0y | 0.03mo | 1.01d | 24.15h | 1449m			24
Chronic airway obstruction, not elsewhere classified	Diagnoses\Respiratory system (460-51…	Set interval	0y | 0.03mo | 1.01d | 24.18h | 1451m			12

The actual valid intervals used in the correlation calculation are shown under the *Valid interval* column. Clicking on the entry brings up a window that allows manually setting the current *valid interval* to a desired value (specified in minutes), or importing one from the pre-calculated set.



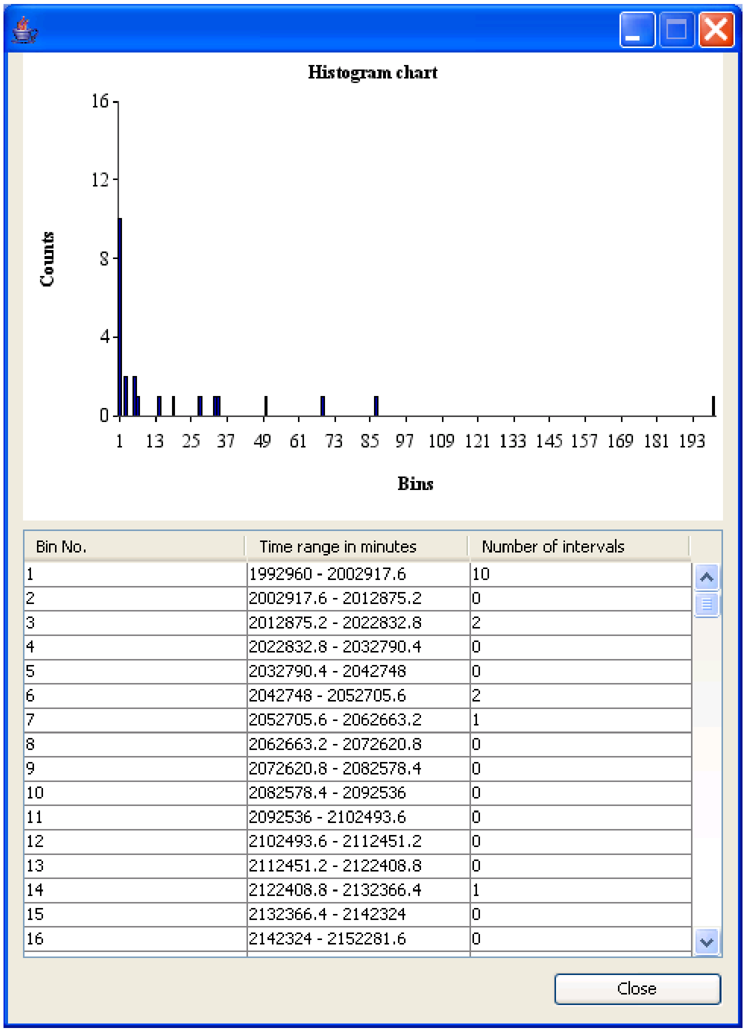
● Set manual 120 minutes.

○ Select from method NORMAL

The entries in bold correspond to *valid intervals* that are infinitely long. When the concepts to be correlated are of type clinical diagnoses, this corresponds to “chronic conditions”. Chronic conditions would have a *valid interval* longer to any time interval based on recorded measurements in the back-end data store, thus being essentially equal to infinity.

The *Correlation Analysis Cell* uses a configuration file (see *Correlation Analysis Developer’s Guide* or the *Installation Guide*) to control which diagnoses will be considered chronic. Naturally, all ‘children’ (as defined by the *Ontology Management Cell* hierarchy) of chronic conditions are considered chronic themselves.

The next three columns, *Normal*, *LPF* and *LPFTot* display results if the corresponding computations were run. The following window is displayed by clicking on an available *valid interval* result. It shows the histogram of all the time intervals for the given concept, over 200 bins. The data table used for constructing the histogram is listed below the histogram.



Bin No.	Time range in minutes	Number of intervals
1	1992960 - 2002917.6	10
2	2002917.6 - 2012875.2	0
3	2012875.2 - 2022832.8	2
4	2022832.8 - 2032790.4	0
5	2032790.4 - 2042748	0
6	2042748 - 2052705.6	2
7	2052705.6 - 2062663.2	1
8	2062663.2 - 2072620.8	0
9	2072620.8 - 2082578.4	0
10	2082578.4 - 2092536	0
11	2092536 - 2102493.6	0
12	2102493.6 - 2112451.2	0
13	2112451.2 - 2122408.8	0
14	2122408.8 - 2132366.4	1
15	2132366.4 - 2142324	0
16	2142324 - 2152281.6	0

The final column on the *Results->Valid Interval* tab, *Interval Count*, lists the number of time intervals the input data for the concept has produced. It is a measure of how many data points were utilized in the calculation and thus serves as an indicator of the statistical significance of the amount of *time overlap*.

All columns on the *Results->Valid Intervals* data grid are sortable by clicking on the column's title.

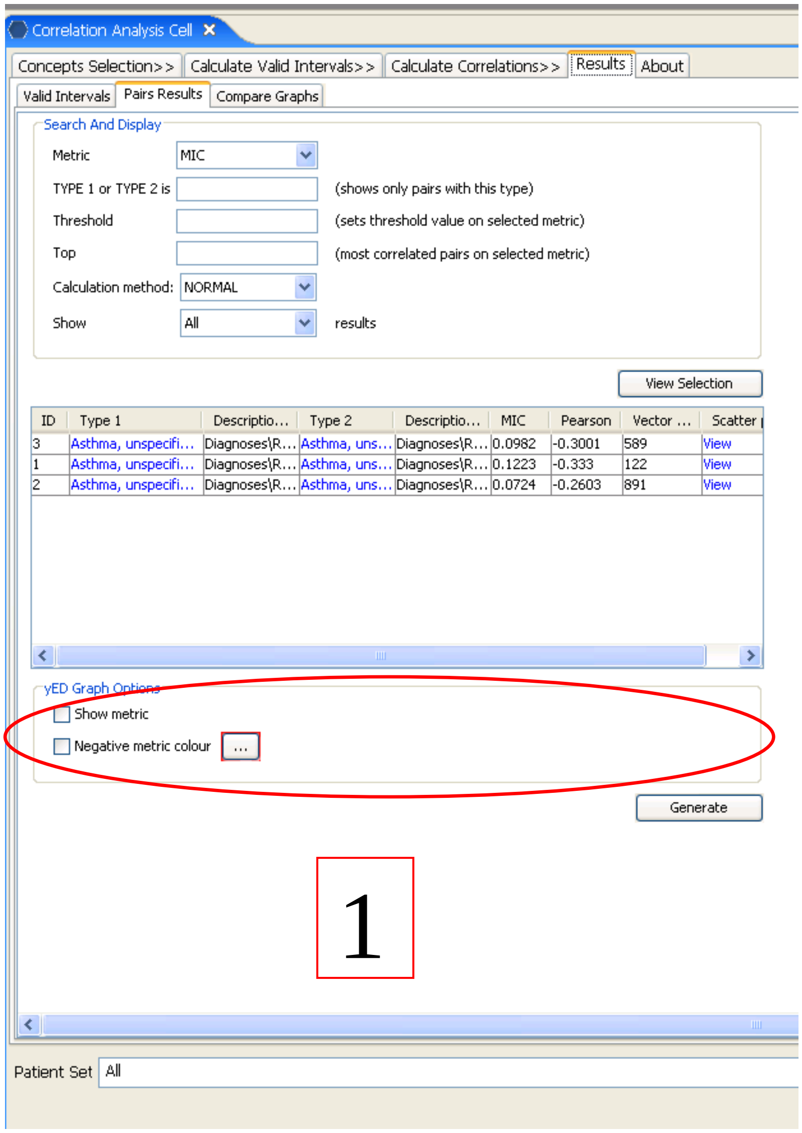
Pair Results

This tab contains utilities to search, display and further analyze pair correlations data. It also supplies utilities to construct Relevance Networks based on the analyzed correlations data sets.

After running the pair-wise correlation computation, select *Results->Pair Results* to display them.

This tab consists of 3 major functionality areas. The *Search and Display* panel is on the top. It allows for constructing searches with fine-tuned search parameters. Multiple non-zero selections are connected with AND. Most parameters and selections are self-explanatory. After making your search adjustments, select All or Non-Zero from the drop-down box next to *Show* and click *View Selection*.

Results matching the selected search criteria are listed in the data area below the search panel. Most columns are self-explanatory. All result sets are sortable by clicking on the respective column's title.



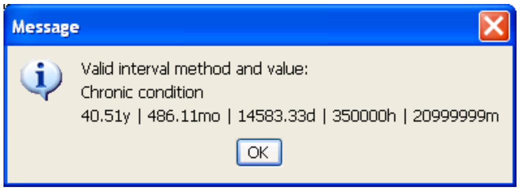
Metric: MIC
TYPE 1 or TYPE 2 is: (shows only pairs with this type)
Threshold: (sets threshold value on selected metric)
Top: (most correlated pairs on selected metric)
Calculation method: NORMAL
Show: All results

ID	Type 1	Descriptio...	Type 2	Descriptio...	MIC	Pearson	Vector ...	Scatter
3	Asthma, unspecifi...	Diagnoses\R...	Asthma, uns...	Diagnoses\R...	0.0982	-0.3001	589	View
1	Asthma, unspecifi...	Diagnoses\R...	Asthma, uns...	Diagnoses\R...	0.1223	-0.333	122	View
2	Asthma, unspecifi...	Diagnoses\R...	Asthma, uns...	Diagnoses\R...	0.0724	-0.2603	891	View

□ Show metric
□ Negative metric colour

Patient Set: All

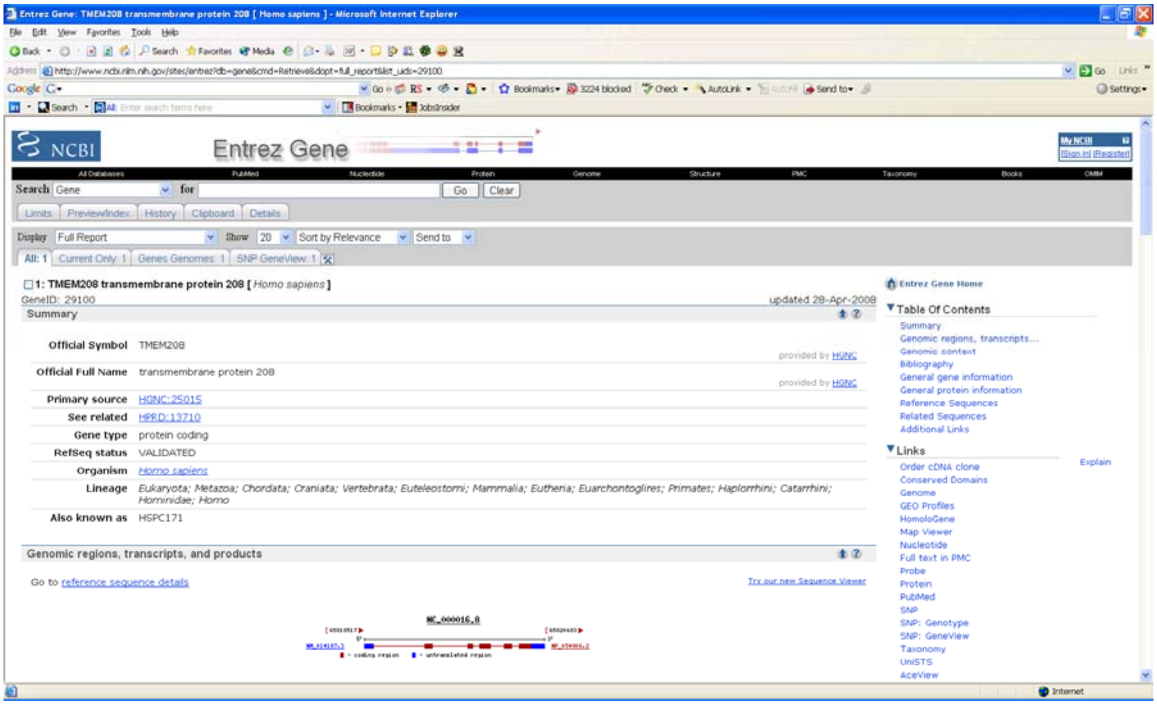
The column *Vector Length* lists the length of the numerical vectors resulting from the *time-overlap* portion of the correlation algorithm. Obviously, the more overlap existed the larger the number in this column thus more statistics will be accumulated in calculating the corresponding correlation.



Valid interval method and value:
Chronic condition
40.51y | 486.11mo | 14583.33d | 350000h | 20999999m

Click on concept's type to review the valid interval used in the correlation metric calculation.

When one of the pair's members is of type expression data, its *Description* column will contain an active URL link to the NCBI's online *ENTREZ* database with complete information for the specific *GeneID*. *GeneID*s are listed in brackets for the given *REF_ID*.



□1: TMEM208 transmembrane protein 208 [*Homo sapiens*]

GeneID: 29100 updated 28-Apr-2008

Summary

Official Symbol	TMEM208 provided by HGNC
Official Full Name	transmembrane protein 208 provided by HGNC
Primary source	HGNC:25015
See related	HPRD:13710
Gene type	protein coding
RefSeq status	VALIDATED
Organism	*Homo sapiens*
Lineage	*Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi; Mammalia; Eutheria; Euarchontoglires; Primates; Haplorrhini; Catarrhini; Hominidae; Homo*
Also known as	HSPC171

Genomic regions, transcripts, and products

Go to reference sequence details

Try our new Sequence Viewer

Entrez Gene Home

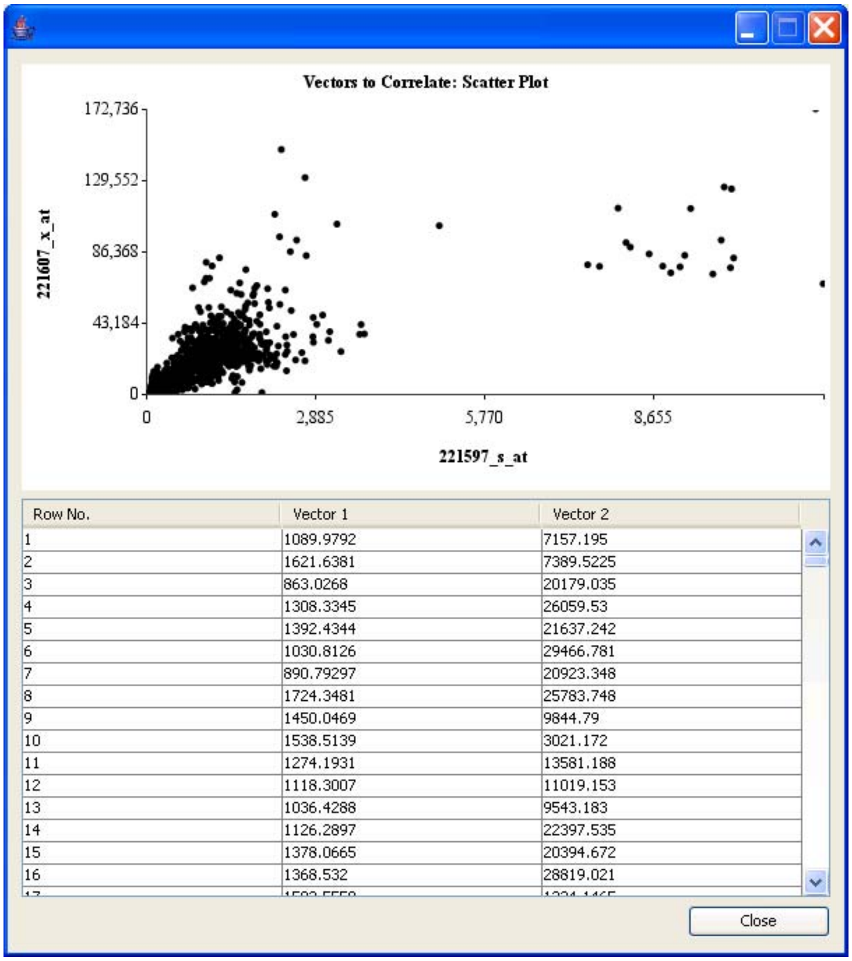
Table Of Contents

- Summary
- Genomic regions, transcripts...
- Genomic context
- Bibliography
- General gene information
- General protein information
- Reference Sequences
- Related Sequences
- Additional Links

Links Explain

- Order cDNA clone
- Conserved Domains
- Genome
- GEO Profiles
- HomoloGene
- Map Viewer
- Nucleotide
- Full text in PMC
- Probe
- Protein
- PubMed
- SNP
- SNP: Genotype
- SNP: GeneView
- Taxonomy
- UniSTS
- AceView

View a scatter plot of the two time-overlapped vectors before they were fed into the MIC and *Pearson* computation. Graph is displayed on the top with the actual coordinated of the overlap for both pair's members given below.

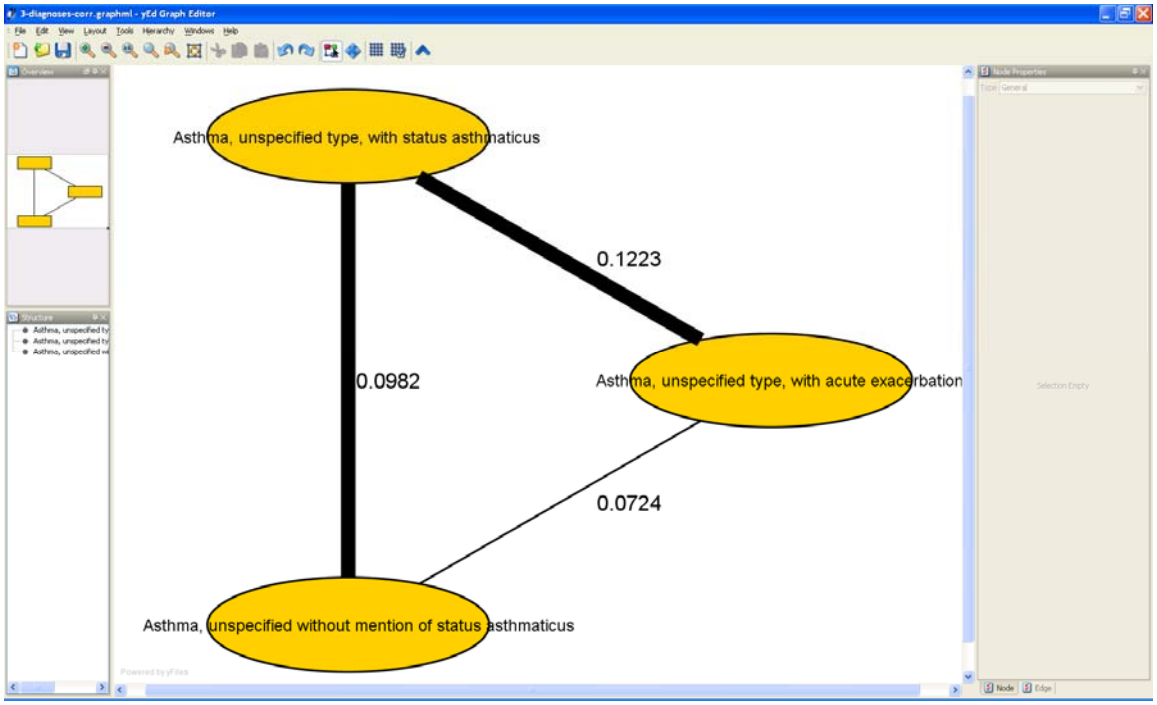


Row No.	Vector 1	Vector 2
1	1089.9792	7157.195
2	1621.6381	7389.5225
3	863.0268	20179.035
4	1308.3345	26059.53
5	1392.4344	21637.242
6	1030.8126	29466.781
7	890.79297	20923.348
8	1724.3481	25783.748
9	1450.0469	9844.79
10	1538.5139	3021.172
11	1274.1931	13581.188
12	1118.3007	11019.153
13	1036.4288	9543.183
14	1126.2897	22397.535
15	1378.0665	20394.672
16	1368.532	28819.021

The panel in the area "1" in the screen shot above allows for constructing and graphing the Relevance Networks based on the data set currently displayed in the data grid. Check the *Show Metric* checkbox to display numerical values of selected metric on the graph. Use the checkbox and color selector utility to specify/change the color of the negatively valued pairs in the correlation metric, i.e. *Pearson's* linear correlation coefficient values.

Clicking on *Generate* button will bring up a *Save File* dialog box. Navigate to a desired directory and give a descriptive name to the file. The *Correlation Cell* will add the "graphml" extension to it automatically.

To review the graphml file the *yEd* editor is needed. Follow installation and configuration directions earlier in this *User's Guide*. Open the file and adjust *Layout* settings to Circular, then accept next dialog's default selections. The Relevance Network based on the data selection detailed in the previous paragraph is displayed in the main window of the yEd editor.


3-diagnoses-corr.graphml - yEd Graph Editor
File Edit View Layout Tools Hierarchy Windows Help
Overview
Structure
Asthma, unspecified ty
Asthma, unspecified ty
Asthma, unspecified w
Asthma, unspecified type, with status asthmaticus
0.1223
0.0982
Asthma, unspecified type, with acute exacerbation
0.0724
Asthma, unspecified without mention of status asthmaticus
Powered by yFiles
Node Properties
Type General
Selection Empty
Node
Edge

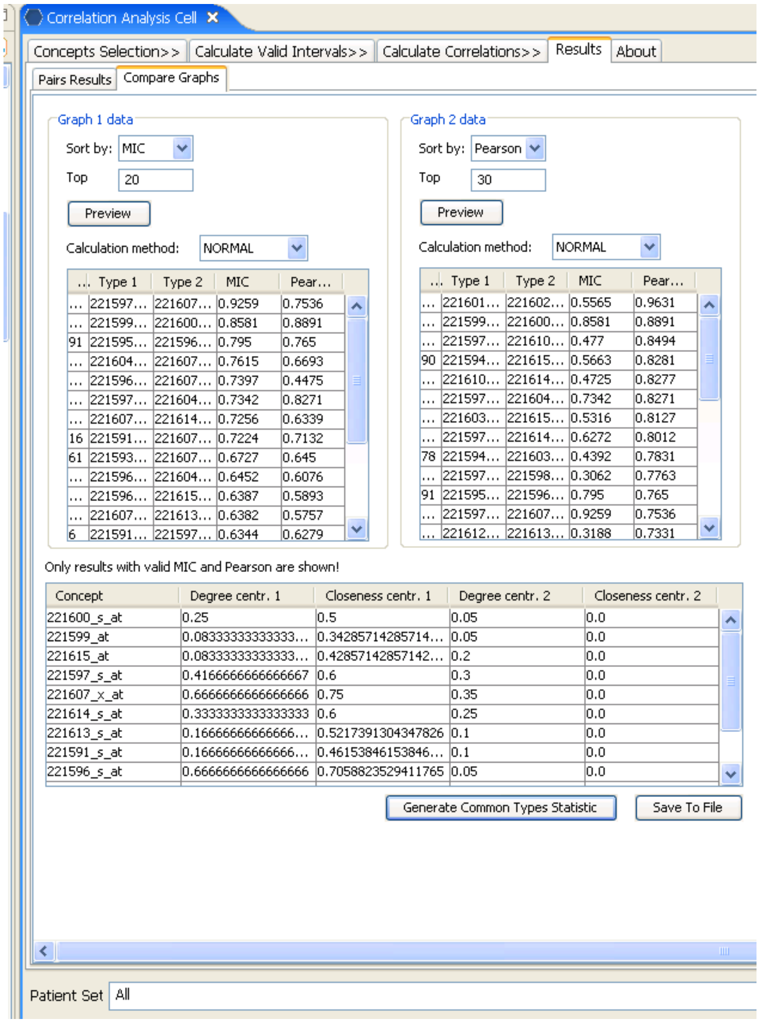
Compare Graphs Tab

One of the most common situations arising in practice is comparing two graphs, actually their corresponding RelNets. Generally, the two sets are drawn from the same super-set of cross-correlated entries. The *Results->Compare Graphs* tab provides some initial utilities to do that based on common similarity measures developed in recent theoretical investigations in Theory of Social Networks. For some exact definitions and relations consult the Appendix "*Social Networks Theory: Point Centrality Measures***"**
later in this *Guide*.

In this example, the two RelNets to be compared are defined as the "top 20 in MIC" and "top 30 in Pearson" genes with expression data.

Use the *Sort By*, *Top* and *Calculation Method* selections, then hit *Preview*; this defines the data set to be compared. Click *Generate Common Type Statistics* to calculate the two centrality measures defined, *Centrality Degree* and *Closeness Centrality*. The common nodes and the two centrality measures for the two sets are listed in the data area.

Save the results to a file by clicking on *Save To File* which brings up a *File Save* dialog box.



Graph 1 data

Sort by: MIC

Top 20

Calculation method: NORMAL

...	Type 1	Type 2	MIC	Pear...
...	221597...	221607...	0.9259	0.7536
...	221599...	221600...	0.8581	0.8891
91	221595...	221596...	0.795	0.765
...	221604...	221607...	0.7615	0.6693
...	221596...	221607...	0.7397	0.4475
...	221597...	221604...	0.7342	0.8271
...	221607...	221614...	0.7256	0.6339
16	221591...	221607...	0.7224	0.7132
61	221593...	221607...	0.6727	0.645
...	221596...	221604...	0.6452	0.6076
...	221596...	221615...	0.6387	0.5893
...	221607...	221613...	0.6382	0.5757
6	221591...	221597...	0.6344	0.6279

Graph 2 data

Sort by: Pearson

Top 30

Calculation method: NORMAL

...	Type 1	Type 2	MIC	Pear...
...	221601...	221602...	0.5565	0.9631
...	221599...	221600...	0.8581	0.8891
...	221597...	221610...	0.477	0.8494
90	221594...	221615...	0.5663	0.8281
...	221610...	221614...	0.4725	0.8277
...	221597...	221604...	0.7342	0.8271
...	221603...	221615...	0.5316	0.8127
...	221597...	221614...	0.6272	0.8012
78	221594...	221603...	0.4392	0.7831
...	221597...	221598...	0.3062	0.7763
91	221595...	221596...	0.795	0.765
...	221597...	221607...	0.9259	0.7536
...	221612...	221613...	0.3188	0.7331

Only results with valid MIC and Pearson are shown!

Concept	Degree centr. 1	Closeness centr. 1	Degree centr. 2	Closeness centr. 2
221600_s_at	0.25	0.5	0.05	0.0
221599_at	0.08333333333333...	0.34285714285714...	0.05	0.0
221615_at	0.08333333333333...	0.42857142857142...	0.2	0.0
221597_s_at	0.4166666666666667	0.6	0.3	0.0
221607_x_at	0.6666666666666666	0.75	0.35	0.0
221614_s_at	0.3333333333333333	0.6	0.25	0.0
221613_s_at	0.16666666666666...	0.5217391304347826	0.1	0.0
221591_s_at	0.16666666666666...	0.46153846153846...	0.1	0.0
221596_s_at	0.6666666666666666	0.7058823529411765	0.05	0.0

Patient Set All

9

Relevance Networks: Theory and Implementation Steps

The theoretical foundation of the Relevance Networks methodology and computational approach was put forward in a series of recent papers [1-3, 4]. The reader is encouraged to consult these references for a more detailed discussion of the approach as well as for review of some initial applications.

The classification of the RelNet approach falls under the general category of unsupervised discovery algorithms without a prior model (see [4] for example Chapter 4, or [5], Chapter 12]. It requires a pre-calculation of all possible pair-wise comparisons between the features under consideration using one or more similarity measures.

The Relevance Networks method was first put forward for the problem of finding hidden relationships in medical databases, thus dealing with correlating temporal data [1]. Later on the approach was extended and applied to the general problem of finding correlation patterns between *N* entities over a number of measurement cases [2, 3].

The most commonly used similarity measures are the *Pearson's* linear correlation coefficient, the Euclidean distance, and the Mutual Information Content (MIC). The *Pearson* coefficient and the Euclidean distance have been routinely used as correlation and clustering similarity measures, with MIC only lately being recognized as such a metric.

The algorithm consists of these general steps.

- Definition of data sets,
- Decision which form of the algorithm to use: temporal vs. general,
- Temporal data case: *valid intervals* finding and selection of vectors construction algorithm,
- Time-independent case: replacement of missing values,
- Removal of Outliers. Low-Entropy filters,
- Selection of similarity measures,
- Pair-wise correlation pre-calculation,
- *Permutation Threshold* testing,
- Correlation results search and display,
- Relevance Networks construction and graphing.

The brief explanations with examples given below can serve as general guidelines in the using the Relevance Networks methodology.

Definition of data sets

This step involves the overall identification of the data sources from which one is to draw data and the processes producing the data to be correlated via RelNets. Identify the “features” list and the number of measurements available. Identify if temporal correlations are important.

Selection of general algorithm

Make the selection for the general type of algorithm to use. If temporal behavior of the data points and their correlation patterns is important, consider *time-overlap* to construct the numerical vectors for correlation.

Temporal data case

Example: Analyze medical databases data and construct correlation Relevance Networks between diagnoses, diagnostic labs and prescribed medications. Cross-correlate with genomic data as it becomes available for more patients in the data store. As each of these concepts corresponds to the activity of ‘taking a medical measurement on a patient’ and inherently carries its own time-stamp, the *time-overlap* algorithm is the logical choice for constructing the numerical vectors for correlation. The notion of *valid interval* was defined to allow a comparison between two separate entities’ data points that took place at different time moments [1].

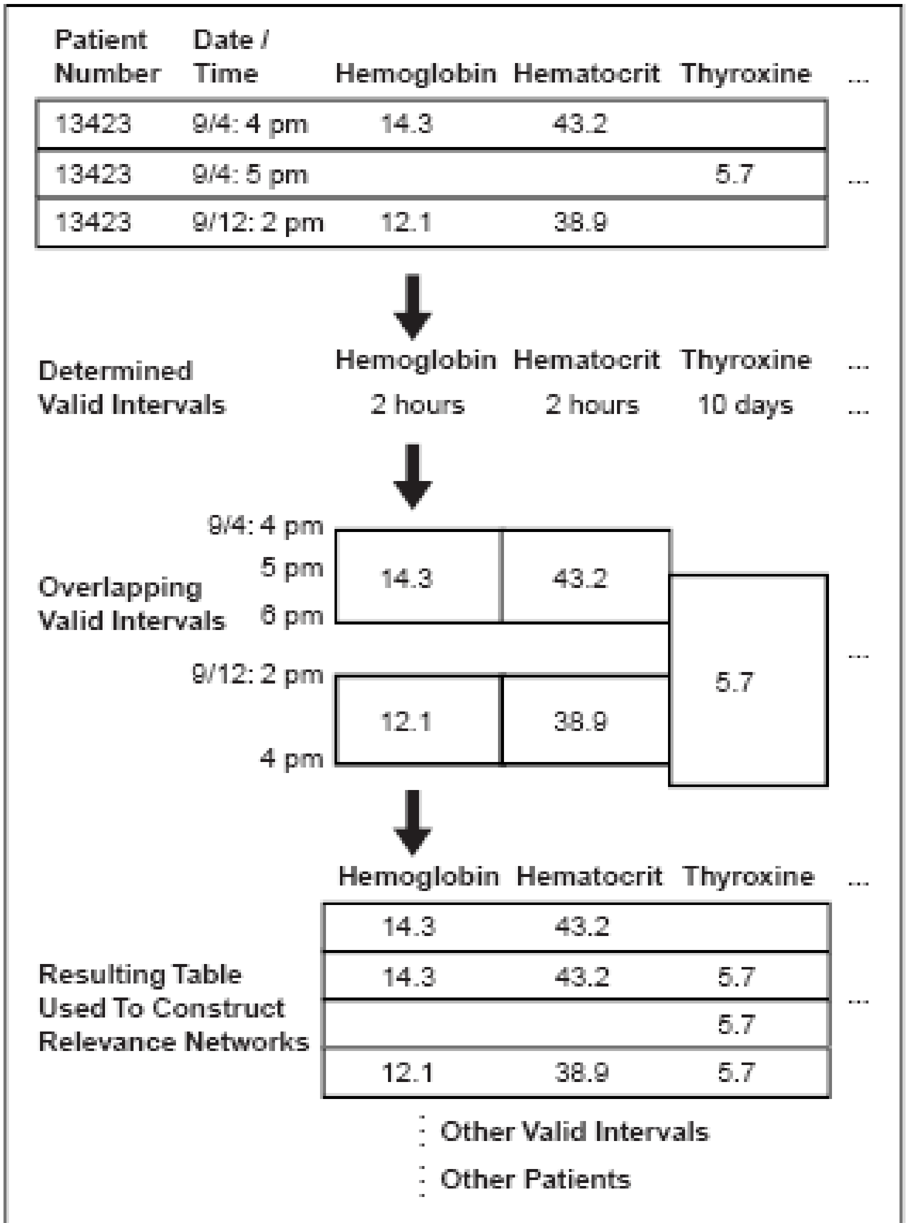
Three procedures have been defined to find the *valid intervals* based on the temporal patterns of the data.

A *valid interval* finder was proposed in the original work (called *normal* in the *Correlation Analysis Cell* implementation). It consists of traversing the data to find all time-intervals between all successive data points and taking the shortest such to be the overall *valid interval* [1].

Another two options are the *Low-Pass Filter* and the *Low-Pass Filter Total* valid interval finders. They both correspond to a process of first constructing the histogram of all time-intervals between all successive pairs of data points, then defining a ‘sliding window’ in which one is looking for s specific pattern of change of the histogram’s envelope function, this trying to only retain histogram peaks with sufficient smoothness. The additional restriction posed by the *LPFTot* finder is that of a maximum spectral power under the histogram envelope function within the window.

Another algorithm selection is required at this point. One needs to choose how to model the comparison between values at different time points, or the *time-overlap*.

One possible choice is a procedure schematically depicted in the figure below. A binary table of time-overlapped entries is constructed (see Fig. 1 in [1] depicting the process of creating the time-overlap between three fictitious lab points). Note the additional entry in the vector for the entity with a longer valid interval.



Patient Number	Date / Time	Hemoglobin	Hematocrit	Thyroxine	...
13423	9/4: 4 pm	14.3	43.2		
13423	9/4: 5 pm			5.7	...
13423	9/12: 2 pm	12.1	38.9		

	Hemoglobin	Hematocrit	Thyroxine	...
Determined Valid Intervals	2 hours	2 hours	10 days	...

Resulting Table Used To Construct Relevance Networks

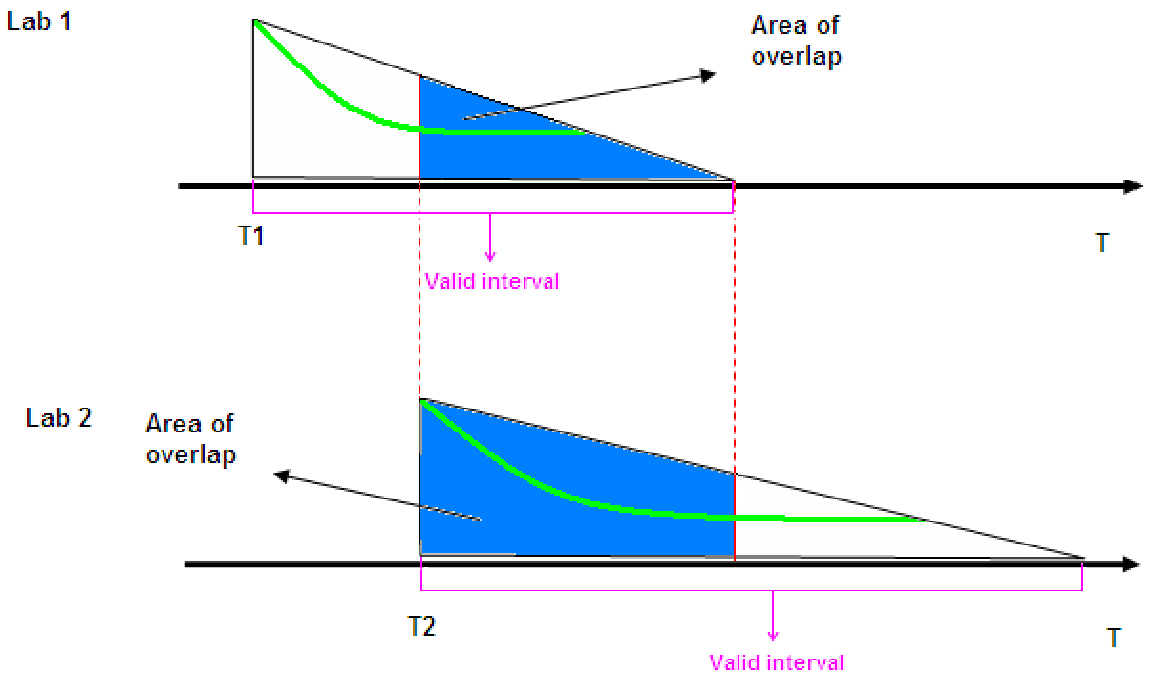
Hemoglobin	Hematocrit	Thyroxine	...
14.3	43.2		
14.3	43.2	5.7	...
		5.7	
12.1	38.9	5.7	

⋮ Other Valid Intervals

⋮ Other Patients

Figure 1: Methodology for creating table of overlapping laboratory results.

Another method of modeling the process of comparison of values at different time-points is shown here.



In this modification, one models the value of the entity at a time later than the time it was originally measured by "expiring the probability of observation" (taken to be unity at the actual moment it was measured) times its original value. Expiration of probability is modeled either in a linear fashion (going from 1 to zero in a straight line), or via a time-decaying exponential tail (see the green exponent-like function on the drawing above). The corresponding components of each member of the correlation pair's numerical vector is given by the ratio of the area of the complete triangle and the portion of the triangle that is in overlap with the other concept's time and valid interval.

Thus, on the figure the 2 blue portions of the top and bottom triangles represent the "area of overlap" for the case of two diagnostic labs (Lab 1 and Lab 2) types that have different time stamps and valid intervals (of expiration of value). The "expiration of validity" function is modeled as a staring line, rather that a dying exponent.

This numerical ratio (of overlap are to area of whole triangle) is multiplied by the original value of the lab type (in the *area* correlation computation algorithm; see the corresponding Chapter in this *User's Guide*), or by 1 (as in the case of diagnoses where one can think of them having a 'value' of unity) to arrive at the *area-only* modification.

By definition, the time overlap algorithm will give equal in size numerical vectors for the two members of the pair. This is in marked contrast to the case when no time-overlap is performed in which case missing values will inevitably produce vectors with disparate lengths.

Time-independent case: replacement of missing values

Example: Correlate all ‘genes’ (PROBE_IDs) as measured by the Affymetrix HG-U133A gene chip. There are 22,287 distinct REF_IDs, i.e. genes. There are 132 measured samples, i.e. number of cases is 132. Therefore, each vector corresponding to a unique gene is of length 132. The ideal case is when all samples have produced numeric value for each PROBE_ID. In practice this is often not the case and one needs to deal with missing values. A possible work-around is to a) ignore values and shorten vectors (used in this implementation), b) pad shorter vector with a constant (usually zero), or c) replace missing value by an average of all other samples’ values.

The example specified above is the basis for the inclusion of the NCBI’s GEO (Genomic Expression Omnibus) data in the i2b2 Demo Data Mart and inclusion of functionality to allow direct cross-correlation of mRNA expression data as part of the *Correlation Analysis Cell* capabilities.

Removal of outliers. Low-Entropy Filters.

It is well-known that even very few outliers can drastically skew the true data distribution (see for example [4], Chapter 4). Once the data set is completely defined including the method to construct its numerical representation vector, an outlier analysis can be carried out. One of the most common such procedures is applying a Low-Entropy Filter. Calculate the entropy of all features to be involved in the correlation computation and remove from the list the ones in the lowest 5-percent range (i.e. the ones with the least uniformly distributed values). A downside of this approach is the exclusion of these features from the subsequent analysis thus not being able to form hypotheses about them.

It should be explicitly mentioned that the current version of the Correlation Analysis Cell does not include filters removal of outliers. One of the main reasons is the reduction of the initial selection list when such a procedure is applied. However, it can be easily added in future releases.

Selection of similarity measures

The most commonly used similarity measures are listed in the **Appendix: Mathematical Definitions**.

Pair-wise correlation pre-calculation

The correlation calculation itself can be run on a desktop or might require a more capable computational platform. It obviously depends on the number of selected concepts and the available data in the Data Mart. i2b2 hive will soon provide for additional utilities to utilize high-performance computing resources (i.e. an HP Cluster).

Permutation Threshold testing

The Relevance Networks methodology includes the notion of Permutation Testing [1, 3]. It is based on the premise that one can discover (and therefore discard) spurious pair correlation (correlations by chance or due to numerical artifacts). Each pair wise computation is repeated N times where the relationship between features and measurement cases is randomly broken. One then retains the largest correlations of the given pair. The global permutation threshold is the largest such value across all pairs. It is hoped that by retaining only correlation values (for a given metric) above the global threshold one is ensured that only biologically relevant pair correlations will be considered.

We have not included permutation testing and threshold finding in Version 1 of the *Correlation Analysis Cell*. Since the *Cell* allows for as little as 1 pair to be constructed and calculated, the statistics in finding the permutation threshold could potentially be misleading in this case, thus we have decided to not support the feature in this release.

Finally, we list the Relevance Networks approach strengths below:

- allows an equal-footing treatment and correlation of disparate characteristics, such as diagnoses, diagnostic labs, medications, genetic and genomic profiles, demographics, socio-economics characteristics etc.
- allows for discovering negative correlations, i.e. entities are “inversely” correlated (this is similarity metrics-specific)
- same general algorithm is applicable to both temporal and time-independent data variables
- works well for sparse (i.e. medical records) as well as for complete (i.e. mRNA expression) data types
- permutation testing allows for discarding correlations by chance therefore the result is having more reliable putative hypotheses generated
- it is a unsupervised, automatic method that is trivially parallelizable and as such can naturally benefit from massively parallel computational architectures (i.e. Cluster environments)

Appendix: Mathematical Definitions

1. *Pearson* linear correlation coefficient ***r*** is the most widely used linear correlation coefficient. For pairs of quantities (x_i, y_i), i = 1,…, N, the linear correlation coefficient ***r*** (also called *Pearson's **r***) is given by the formula

$$r = \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_i (x_i - \overline{x})^2}\sqrt{\sum_i (y_i - \overline{y})^2}} \qquad \text{(\textit{Equation 1})}$$

where, as usual, $\overline{x}$ is the mean of the x_i's, $\overline{y}$ is the mean of y_i's.

2. The Mutual Information Content (MIC) is a measure of the additional information known about one variable given the patterns of another, as shown in Equation 2.

$$MI(A, B) = H(A) - H(A \mid B) \qquad \textit{(Equation 2)}$$

Mutual information can be calculated by subtracting the entropy of the joint variables' patterns from the individual variable's entropies.

$$MI(A, B) = H(A) + H(B) - H(A,B) \qquad \textit{(Equation 3)}$$

As usual, the discrete entropy of the variable A is given by the summation over its *n* discrete values:

$$\mathrm{H(A)} = \sum_{i=1}^{n} \mathrm{p(x)\,log2(\,p(x\,)\,)}\,, \qquad \textit{(Equation 4)}$$

One needs to construct the *Contingency Table* for the case of the 2-dimensional entropy defined in Equation 3 (see [5], Chapter 14]).

Social Networks Theory: Point Centrality Measures

Point centrality measures how centrally a point is located in a given graph. Usually the point at the center of a star or the hub of a wheel is the most central possible position. There are three approaches to measuring the point centrality:

1) Point centrality based on **the degree**: the point with the largest degree is the most central point in the graph.

Nieminen's (1974) measurement: count of the degree or number of adjacencies, for a point p_k:

$$CD\left(p_k\right)=\sum_{i=1}^{n} a\left(p_i, p_k\right)$$

where n = number of points

$a(p_i,p_k)$=1 if and only if pi and p_k are connected by a line, 0 otherwise

If we control the size of the graph by dividing $C_D(p_k)$ by n-1, which is the maximum degree of p_k in any graph, and make the measurement comparable between graphs, we get a "relative centrality" for p_k, $C'_D(p_k)$.

2) Point centrality based on **closeness**: the point which is closest to all other points in the graph is the most central point.

Sabidussi's (1966) measurement:

$$C_c\left(p_k\right)^{-1}=\sum_{i=1}^{n} d\left(p_i, p_k\right)$$

where $d(p_i,p_k)$ = the number of edges in the geodesic linking pi and p_k.

The "relative centrality" based on closeness is:

$$C'_c(p_k) = \left[\frac{\sum\limits_{i=1}^{n} d(p_i, p_k)}{n-1} \right]^{-1} = \frac{n-1}{\sum\limits_{i=1}^{n} d(p_i, p_k)}$$

References

1. Butte, A. & Kohane, I.S. Unsupervised Knowledge Discovery in Medical Databases Using Relevance Networks. in *Fall Symposium, American Medical Informatics Association* Vol. (ed. Lorenzi, N.) (Hanley and Belfus, Washington, DC,1999).

2. Butte, A.J. & Kohane, I.S. Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. *Pac Symp Biocomput*, 418-29 (2000).

3. Butte, A.J., Tamayo, P., Slonim, D., Golub, T.R. & Kohane, I.S. Discovering functional relationships between RNA expression and chemotherapeutic susceptibility using relevance networks. *Proc Natl Acad Sci U S A* **97**, 12182-6 (2000).

4. In: *Microarrays for an Integrative Genomics*, Isaac Kohane, Alvin Kho, and Atul Butte, MIT Press, August 2002.

5. In: *Numerical Recipes in C: the art of scientific computing*, William Press, Saul Teukolsky, William Vetterling, and Brian Flannery, Cambridge University Press, Second Edition (1992).

6. Nieminen, J., On centrality in a graph, Scandinavian Journal of Phychology, **15**:322-336 (1974).

7. Sabidussi, G. The centrality index of a graph. Psychometrika, **31**, pp. 581-603 (1966).