

Metagenomic analysis with Merlin

Pedro Barbosa

July, 2014

Contents

1	Introduction	2
2	Tutorial for Using Merlin	3
2.1	Creating a new project	3
2.2	Load KEGG data to internal database	5
2.3	Annotation of the metagenome	5
2.3.1	Local BLAST	5
2.3.2	Remote search	7
2.3.3	Enzyme assignments	7
2.3.4	Parallel annotations	9
2.4	Data integration	10
2.5	Taxonomic composition	12
2.5.1	Methodology	12
2.5.2	Merlin's operation mode for taxonomy	14
2.6	Enzyme analysis	16
2.6.1	Methodology	16
2.6.2	Merlin's operation mode for enzymes	16
2.7	Pathways characterization	19
2.7.1	Methodology	19
2.7.2	Merlin's operation mode for pathways	20
3	Using more memory	23
4	Contact	23

1 Introduction

The extension of Merlin from the previously version [1] enables the analysis of shotgun metagenomic data both at taxonomic and functional level. The software has a user-friendly and intuitive interface not requiring command-line knowledge and further libraries dependencies or installation. It is an open-source application implemented in *Java*TM and is built on top of the AIBench (<http://www.aibench.org>) software development framework [2].

It requires as input a file with gene encoding sequences, therefore a metagenomics prediction software must be run before over the contigs/scaffolds generated by the assembling process. Since the execution of the features depends on the results from an homology search, the BLAST tool has to be run first to perform the annotation of the sample and afterwards loaded into a *Merlin* internal database. Finally, the results from BLAST are used to feed the taxonomic and functional routines, respectively.

This extended version of Merlin can predict the taxonomic composition of an environmental sample based on the results of homology searches, where the proportions of phyla and genera present are discriminated. Regarding the metabolic analysis, it allows to identify which enzymes are present and calculate their abundance, as well as to find out which metabolic pathways are effectively present.

The workflow for metagenomics projects in *Merlin* is displayed in Figure 1. It is only available on Linux for now, and it works reasonably on a computer with 3GB of memory, although more memory is advantageous.

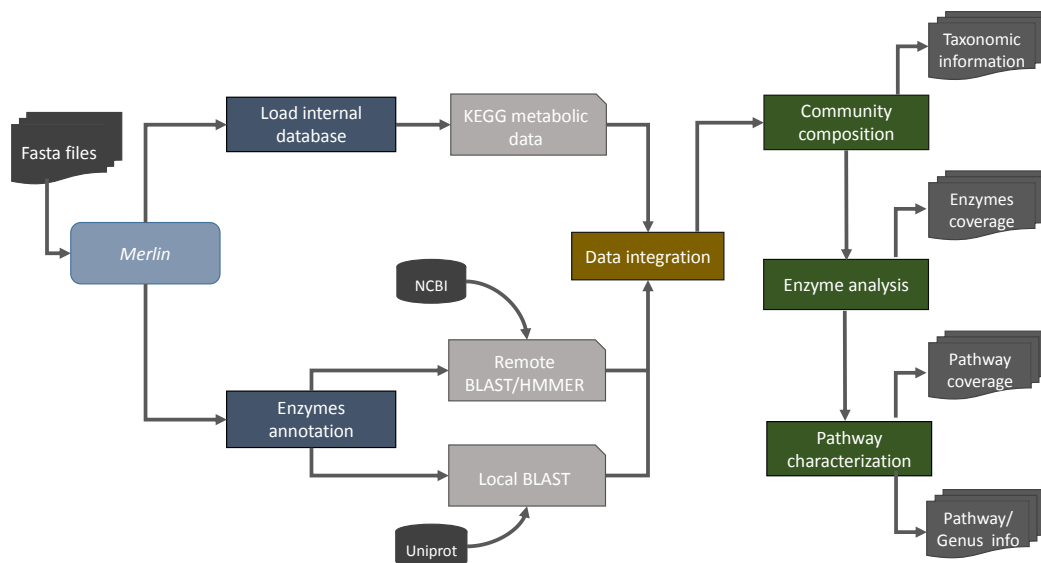


Figure 1: Schematic representation of Merlin architecture for metagenomic analysis.

2 Tutorial for Using Merlin

2.1 Creating a new project

Since the information of each project is stored in an internal database, the first step is to create a new database for the project. Therefore, a MySQL server instance must be installed in order to Merlin be able to execute the operation. In the '*Database*' menu, click on 'New Database' and insert a new name for the database as well as the arguments required to connect the installed MySQL server instance (Figure 2).

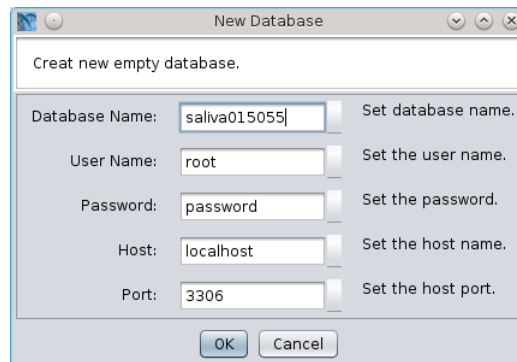


Figure 2: Merlin's view for creating a new database.

Then, go to the '*Project*' menu and select '*Create project*'. A new window will appear where the user shall choose a name for the project and its internal database, as well as click on the '*Is this a metagenomics project?*' checkbox to avoid Merlin to treat the new project as a single organism one. To select the input file with aminoacid sequences in the FASTA format, it must have the 'faa' extension so that Merlin can recognize it. After this, click '*Ok*' and the new project with all needed information is created (Figure 3). A project can also be created without indicating the input file, but then the user has to go to the '*Set genome file*' option in the '*Project*' menu so that he is able to perform further operations.

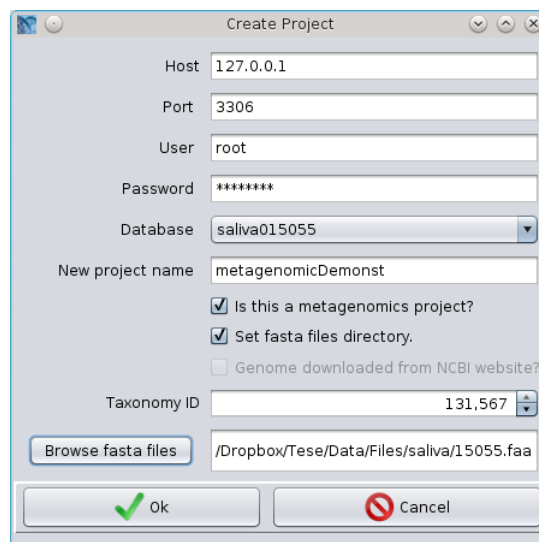


Figure 3: Merlin's view for creating a new project.

2.2 Load KEGG data to internal database

To load KEGG data, go to the '*Database*' menu and click on '*Load KEGG data*' option. Then, do not select any organism and click '*Ok*' so that all information stored in KEGG about pathways, enzymes, reaction and compounds is loaded into the internal database selected for the project (Figure 4).

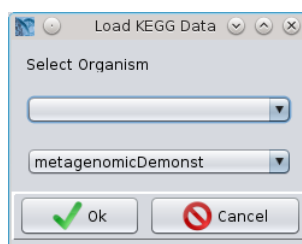


Figure 4: Merlin's view for loading KEGG data into the internal database.

2.3 Annotation of the metagenome

The annotation of a sample is done through a homology search using either BLAST or HMMER. There are two ways to execute this operation: locally and remotely. The remote version enables the choice between HMMER and BLAST, while the local implementation is possible only with BLAST. Of course this requires a local version of the BLAST program installed in the computer but details on this topic will be described next.

2.3.1 Local BLAST

The local Blast version was implemented to enable some speed improvements in big samples such as a metagenomics one. To setup a standalone version of BLAST, the user must follow the steps provided <http://www.ncbi.nlm.nih.gov/books/NBK52640/>. An important one is to modify the existing \$PATH variable with the path to the new BLAST bin directory, otherwise Merlin will not recognize the program. Merlin was tested with the ncbi-blast-2.2.28 version, but no problems might be found with other versions. Since the BLAST output only provides a list of the homologues with their respective score and e-value for each gene, it is necessary to retrieve more detailed information of each homologue to fill the Merlin's internal database tables. Regarding this task, a parser of the Uniprot database was developed,

which means that for now, it is only possible to execute this operation against that database.

For that, the user must download the reference database (either UniProtKB/Swiss-Prot, UniprotKB/TrEMBL or both) from the ftp uniprot website and configure it according the standalone BLAST setup instructions. The corresponding text file (www.uniprot.org/downloads) also needs to be downloaded for the homologue retrieval step.

After the required configuration of the Blast and local databases, Merlin is able to perform the operation by clicking in the '*Enzymes*' menu and '*BLAST annotation (local)*' option (Figure 5). First, the program blasts the genes against the local database and the results are stored in a temporary directory. Then, the retrieval of homologues information is done by parsing the text file that contains all the information of each record in the database. For each homologue the following fields are saved: UniprotID, Uniprot status, Full name, Ecnnumber, Gene name, Organism, Full taxonomy and Sequence. Afterwards, all this information is loaded into the project database.

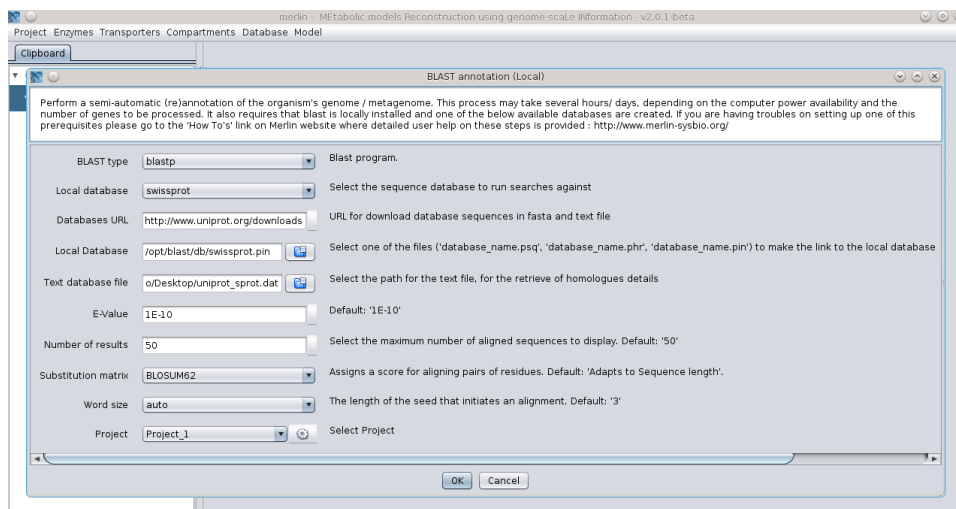


Figure 5: Merlin's Local Blast operation view.

Concerning the local Blast operation view, the first option refers to the Blast program used. In this case, the BLASTp is used, since the predicted ORFs come as amino acid sequences. The user can then choose the local database to perform the annotation, either the SwissProt or the whole UniprotKB (merge of the UniProtKB/Swiss-Prot and the UniProtKB/TrEMBL databases). The user may have to choose a trade-off between sensitivity and time. If SwissProt is selected, which is a small database, the

annotation can be very fast despite the loss of information. On the other hand, if the user chooses to use the entire Uniprot (huge database), some problems with the computation times will may occur. However the user will gain on sensibility, i.e more results will be reached in the annotation.

The user has to select the directory where the local database was created and the database file for the homologues information retrieval explained above. The normal BLAST parameters, such as minimum e-value, number of results and substitution matrix can also be specified by the user.

2.3.2 Remote search

This operation is performed using the Entrez Programming Utilities API provided by NCBI and enables to search the sequences against databases such as NCBI non-redundant database. By clicking on '*Enzymes*' menu and '*BLAST annotation (remote)*' the user can specify the parameters of the BLAST he wants (Figure 6). The HMMER annotation can be accessed in the '*Enzymes*' menu and '*HMMER annotation*' option.

BLAST annotation (Remote)

Perform a semi-automatic (re)annotation of the organism's genome. This process may take several hours, depending on the web-server availability.

BLAST type	blastp	Blast program. Default: 'blastp'
Genetic Code	Standard	Genetic Code. Used for blastX, else ignored.
E-Value	1E-30	Default: '1E-30'
Adjust E-Value	<input checked="" type="checkbox"/>	Automatically adjust e-value for smaller sequences search
Remote database	nr	Select the sequence database to run searches against
Number of results	100	Select the maximum number of aligned sequences to display. Default: '100'
Substitution matrix	AUTOSELECTION	Assigns a score for aligning pairs of residues. Default: 'Adapts to Sequence length'.
Word size	auto	The length of the seed that initiates an alignment. Default: '3'
Organism		(Optional) Enter organism ncbi taxonomy id
Project	metagenomicDemonst	Select Project

OK Cancel

Figure 6: Merlin Remote Blast operation view.

2.3.3 Enzyme assignments

Merlin uses a routine to assign EC numbers to each gene g . These are done by assigning a weight to the number of times each EC number ec is found within the list of homologues for each gene (frequency), as well as to the

taxonomy of the organisms to which such homologues belong to [3]. The equation 1 describes how the scoring process is done, where the influence of the frequency score ($Score_f$) and the taxonomy score ($Score_t$) depend on an α parameter that controls the weight to give to each for the overall score:

$$Score_{ec}^g = \alpha \times Score_f + (1 - \alpha) \times Score_t \quad (1)$$

For single genome projects the default value of α is 0.5, meaning that the same weight is given to the frequency and taxonomy score. Since the taxonomy score is used to favor homologies with records of closely related taxonomies to the organism being studied, this score does not make sense for metagenomics projects, where the focus stands on the whole community instead of a single organism.

Therefore, for metagenomics projects the α default value is set to 1, making the EC number scores calculations only based on the frequency score. This score counts the number of occurrences of an EC number ec within all homologues of that gene, and divides this value by the total number of homologous genes n , as described in equation 2.

$$Score_f^{ec} = \sum_{i=1}^n \frac{(\nu_{ec_i})}{n} \quad (2)$$

where:

$$\nu_{ec_i} = \begin{cases} 1, & \text{if } ec \text{ exists in record } i \\ 0, & \text{otherwise} \end{cases}$$

The score will always have a numeric value between 0 and 1. A minimum score threshold is defined to automatically accept the annotations. For single genome projects this value is set by default to 0.5, which means that scores smaller than this value will not be annotated, despite the possibility of the user to manually curate it and accept the result.

In metagenomics, given the high amount of genes to be annotated, it is typically not feasible for the user to manually check and curate enzymes annotation, so all the enzymes in the metagenome will normally be automatically processed. Since metagenomics harbors a massive amount of genes that are poorly characterized, it is expected that in some cases, the assigned

EC numbers have low confidence scores. Given these facts, the default minimum score threshold for metagenomics projects is set to 0.3 (the user is still able to set the value to a more fitted one for each specific project).

The information about the homology search can be accessed in the '*Homology Data*' entity on the left side of the main window (Figure 10).

2.3.4 Parallel annotations

Since the input file in *Merlin* for metagenomics datasets is composed of a high number of predicted genes, it can be useful for the user to split it into several files and run the similarity searches for each of them in different computers with *Merlin* installed. Once this is done, the user may want to gather all the annotations again into one project with its respective database. Next, this process is described in detail:

1. The user saves a backup of the database to export to another computer in the '*Database*' menu and '*Save Database Backup*' option (Figure 7) (Note that this option is also helpful to avoid loss of information in case of any problem with the project or the database).
2. The outputted file, that comes in the Structured Query Language (SQL) format, is copied into the computer who hosts the database that will merge the results and the user can create a new project using the newly created database from that file (in the '*Database*' menu and '*New Database from SQL file*' as shown Figure 8).
3. Given two projects, the user can merge their databases in the '*Database*' menu in the '*Merge databases*' option (Figure 9) being the annotation of all genes of the metagenomics sample together in the same project.

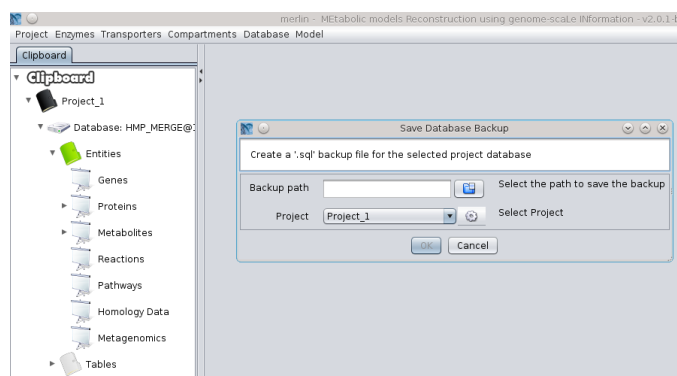


Figure 7: *Merlin*'s view for saving a backup of the project database.

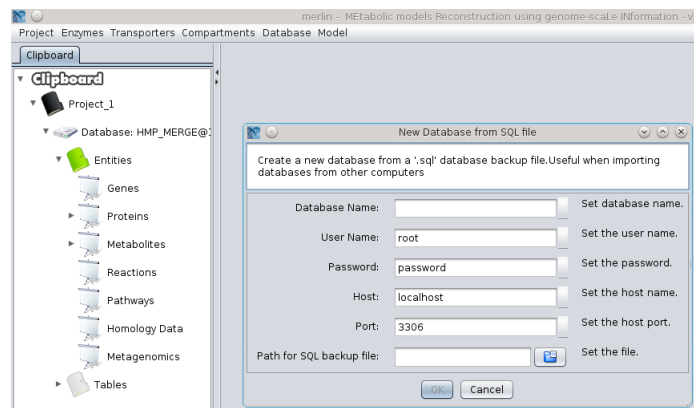


Figure 8: *Merlin's* view for creating a new database from a backup SQL file.

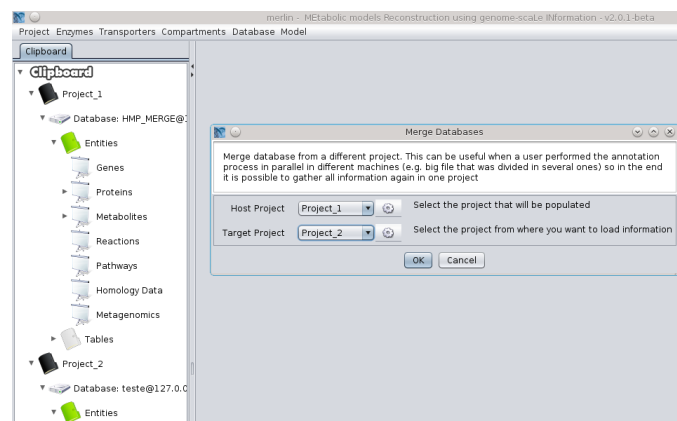


Figure 9: *Merlin's* view for merge databases from two projects.

2.4 Data integration

This operation is essential so that all the homology results are integrated with the KEGG information loaded before. Without this step, the functional metagenomics analysis can not be performed. The user must click on the 'Integration' button on the bottom right corner on the main window of the 'Homology Data' entity (Figure 10).

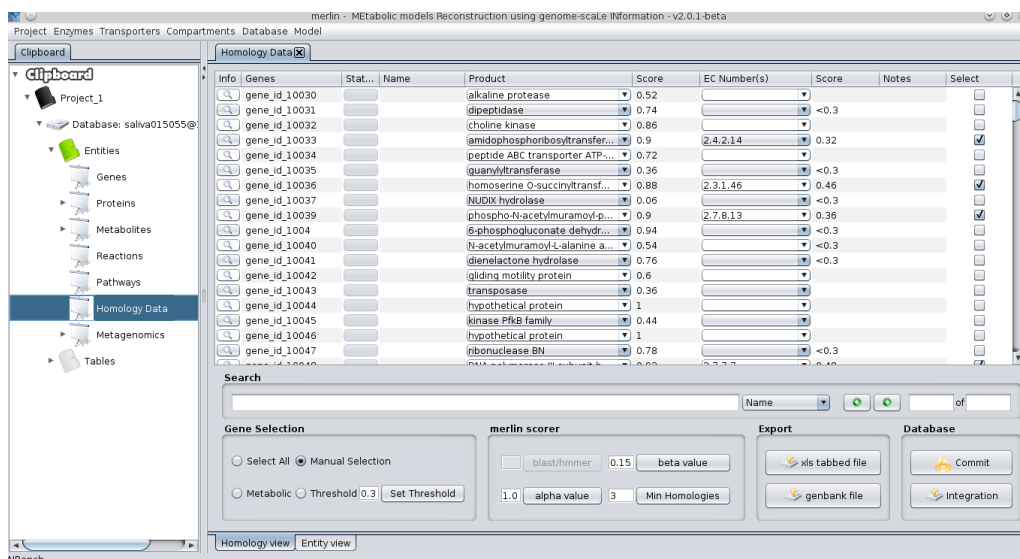


Figure 10: Merlin's view for Homology data information and database integration.

Then, a small window will appear when the user is able to choose about the preferences on the integration. The default values can be maintained as long as in the last combo box the option '*Integration*' is chosen (Figure 11).

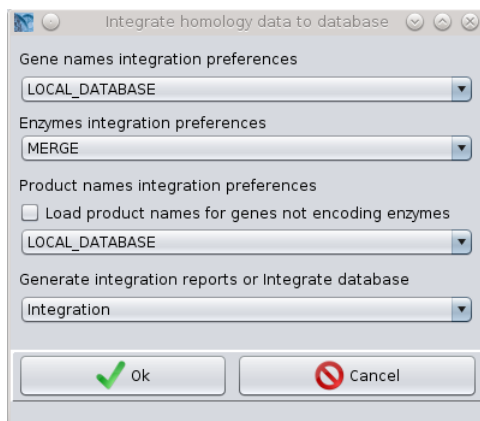


Figure 11: Homology data integration options.

2.5 Taxonomic composition

2.5.1 Methodology

The methodology for taxonomic characterization employed in Merlin purposes to assign a taxonomic label to each gene, as well as to describe the overall community composition. Thus, it classifies each gene at the phylum and genus level based on the list of homologues obtained from the homology search. Afterwards, given a classification for each gene, it calculates the proportions of each taxon in the whole set of genes.

A routine was developed to assign a phylum and genus to each gene. The assignments are performed giving a weight to the number of times each phylum and genus are found within the gene homologues list. A gene is only classified if it contains a minimum number of homologues and the phylum and genus scores are higher than two defined thresholds.

Therefore, for each gene g , given an ordered set of N homologues (higher than k , the minimum number of homologues allowed), the phylum scores ($Score_{phylum}$) is calculated according the equation 3. The ideal value for $Score_{phylum}$ is 1, which would mean that every homologue in the set is of the same phylum.

$$Score_{phylum}^g = \frac{Score_{bestph}}{Score_{max}} \quad (3)$$

The best phylum score ($Score_{bestph}$) for each gene represents the phylum candidate from the whole list of homologues that achieved the best score to be tested (equation 4). Theoretically, it would be expected that the phylum that occurs most in the list homologues would be the one selected as the candidate. However, it was decided to privilege the first five homology hits, since those are likely to be more taxonomically related to the gene being tested:

$$Score_{bestph}^g = \sum_{i=1}^N f_{bestph_i} \quad (4)$$

where:

$$f_{bestph_i} = \begin{cases} 2.0, & \text{if homologue in position } i \text{ belongs to phylum } bestph \text{ and } i \leq 5 \\ 0.5, & \text{if homologue in position } i \text{ belongs to phylum } bestph \text{ and } i > 5 \\ 0, & \text{otherwise} \end{cases}$$

In the cases where two or more candidate phyla have the same score, the routine chooses the taxon that comes first in the homologues list as the best candidate for a given gene.

The maximum score ($Score_{max}$) shown in equation 5 represents the highest possible score for gene g given the number of homologues:

$$Score_{max}^g = \sum_{i=1}^N f_i \quad (5)$$

where:

$$f_i = \begin{cases} 2.0, & \text{if } i \leq 5 \\ 0.5, & \text{otherwise} \end{cases}$$

This routine explained above describes the methodology developed in this work for phylum scoring. For genus assignment, the procedure is exactly the same, with exception for the minimum score threshold. In the end, a gene will be assigned with a taxonomic label only if it fulfills the following criteria:

- The number of homologues is higher than the minimum number required k (default value is 5), otherwise the routine will not even perform the phylum/genus scores calculations.
- The phylum score $Score_{phylum}^g$ is higher than the phylum threshold (default value is 0.5).
- The genus score $Score_{genus}^g$ is higher than the genus threshold (default value is 0.3).
- The phylum and genus are congruent, that is, the selected genus belongs to the selected phylum.

2.5.2 Merlin's operation mode for taxonomy

The described methodology was integrated in *Merlin* in a user-friendly interface as depicted in Figure 12. This visualization panel is accessed by clicking on the '*Taxonomy*' sub-view under the '*Metagenomics*' entity. It comprises taxonomic information for all genes, including the selected phylum and genus for each gene and their scores. Information for those genes that do not present a minimum number of homologues, the scores are lower than the thresholds and the phylum/genus are incongruent is also provided.

Info	Genes	Phylum	Phylum Score	Genus	Genus Score	Phylum/Genus are concordant
	gene_id_1006	Bacteroidetes/Chlorobi gr...	1.0	Prevotella	1.0	true
	gene_id_10060	No homologues enough				
	gene_id_10061	Proteobacteria	1.0	Haemophilus	0.6769	true
	gene_id_10062	Proteobacteria	1.0	Haemophilus	0.7344	true
	gene_id_10063	Bacteroidetes/Chlorobi gr...	0.9846	Prevotella	0.8154	true
	gene_id_10064	Bacteroidetes/Chlorobi gr...	1.0	Prevotella	0.8333	true
	gene_id_10065	Bacteroidetes/Chlorobi gr...	1.0	Prevotella	0.8154	true
	gene_id_10066	Bacteroidetes/Chlorobi gr...	1.0	Prevotella	0.8308	true
	gene_id_10067	Proteobacteria	0.9846	Aggregatibacter	0.2154(< 0.3)	no minimum score
	gene_id_10068	Firmicutes	1.0	Veillonella	0.4308	true
	gene_id_10069	Firmicutes	1.0	Veillonella	0.7949	true
	gene_id_1007	Bacteroidetes/Chlorobi gr...	1.0	Prevotella	0.7231	true
	gene_id_10070	Bacteroidetes/Chlorobi gr...	0.975	Prevotella	0.9	true
	gene_id_10071	Firmicutes	1.0	Streptococcus	1.0	true
	gene_id_10072	Proteobacteria	1.0	Haemophilus	0.3692	true
	gene_id_10073	Firmicutes	0.6923	Campylobacter	0.3077	false
	gene_id_10074	Firmicutes	1.0	Lachnoanaerobaculum	0.381	true
	gene_id_10075	No homologues enough				
	gene_id_10076	No homologues enough				
	gene_id_10077	Proteobacteria	1.0	Haemophilus	0.6462	true

Search
 Name of

Set parameters
 Minimum number of homologues:
 Minimum phylum score:
 Minimum genus score:

Recalculate

Export

Metagenomics taxonomy view Entity view

Figure 12: Merlin's view for taxonomy information of metagenomic datasets.

As the above figure shows, the user can easily change the values for some parameters of the scoring algorithm: the '*Minimum number of homologues*' text box allows to set a new value for the required number of homologues that each gene must have for the algorithm to perform the calculations; the '*Minimum phylum score*' and '*Minimum genus score*' text boxes can be altered to set the minimum scores for which a gene gets a valid taxonomic assignment. The scores can be re-calculated and the main table updated by clicking on the '*Taxonomic composition*' button.

The '*info*' column provides a button for each gene that allows to access detailed information about the phylum/genus scores for a given gene by showing all taxonomic elements used for that specific classification in a pop-up window (Figure 13).

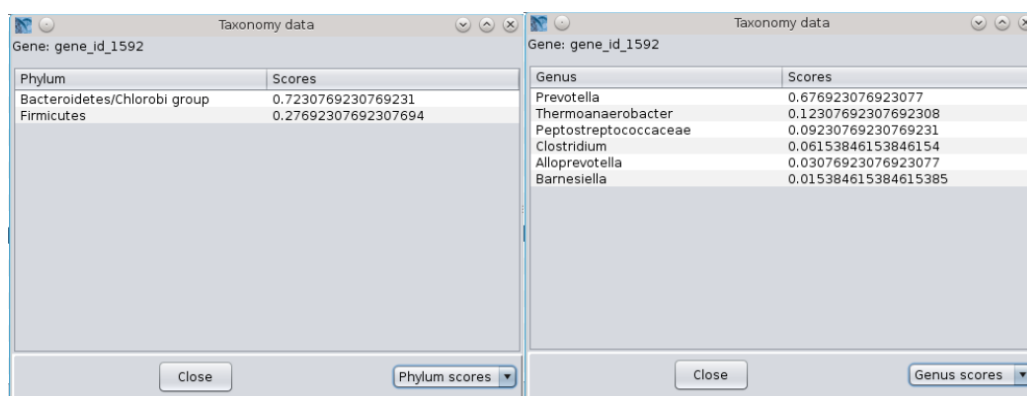


Figure 13: Detail windows for the '*Taxonomy*' main view (Phylum scores on the left, Genus scores on the right).

By clicking on the '*Entity view*' tab in the panel, the user is able to see the main statistics of the scoring algorithm (e.g number of genes that did not achieve the minimum scores) as well as the overall community composition, where the percentages of the phylum and genus are discriminated (Figure 14). This information can also be exported to a text file by clicking on the '*text file*' button in the main window.

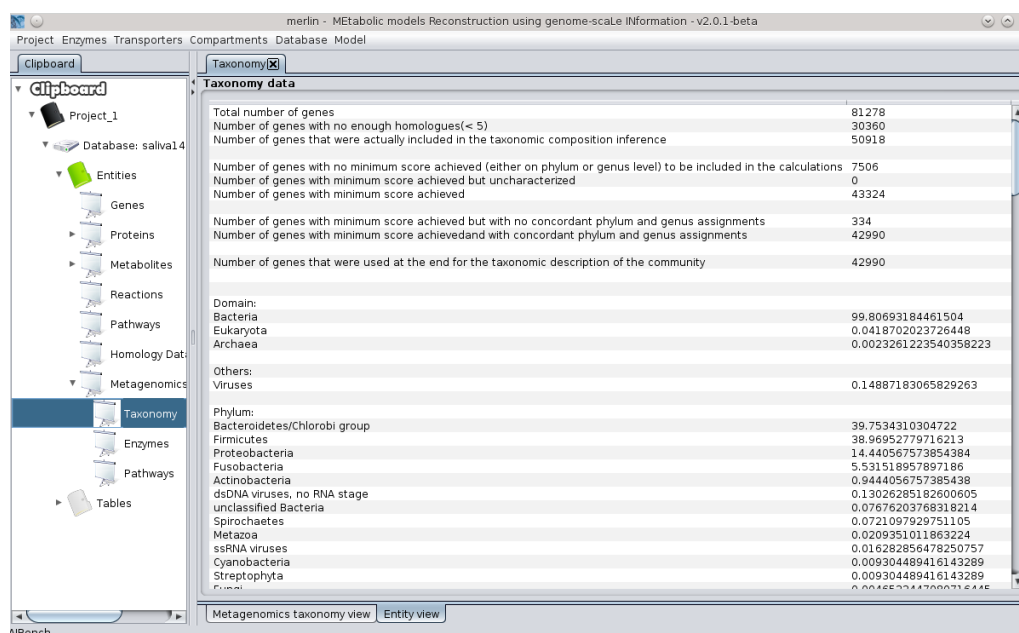


Figure 14: Statistics and overall community composition displayed in Merlin.

2.6 Enzyme analysis

2.6.1 Methodology

This routine aims to characterize the metabolic enzymes that are present in the metagenome and assign an abundance according to the number of times each of them is encoded in the whole set of genes. Furthermore, it tries to associate the taxonomic genus that encodes each enzyme. This operation is highly dependent on the annotation and the taxonomy, thus it is desirable that these previous steps work well to get better results.

From the set of all enzymes loaded from KEGG, this procedure selects the ones with a complete EC number that have at least one annotated gene in the metagenome, being automatically assumed that those enzymes are present in the community. Regarding the enzyme abundance calculation *Abundance*, given the set of all N annotated metabolic genes (Ω_n) and a set of T genes (Ω_t) encoding for the EC number ec , the routine calculates enzyme abundance according to the equation 6:

(6)

Afterwards, this method checks on the genes encoding an EC number if they have a taxonomic genus assignment from the previous taxonomy routine. If so, it is assumed that a specific genus encodes that EC number in the microbial community. On the other hand, if none of the genes encoding an EC number has a genus assignment (e.g. no minimum number of homologues, no minimum score), that enzyme is treated as present but with no taxonomic information regarding it.

2.6.2 Merlin's operation mode for enzymes

The visualization panel for metagenomics enzymes is integrated in Merlin in the '*Enzymes*' sub-view under the '*Metagenomics*' entity. The main view encompasses information of all encoded enzymes in the dataset. For each enzyme, the number of genes encoding it, the underlying reactions and their abundances are displayed. Furthermore, information is provided concerning the number of genes with taxonomic genera and the number of genes without them (Figure 15). Since taxonomic information is required to execute this operation, the user must execute the taxonomy routine first by clicking on

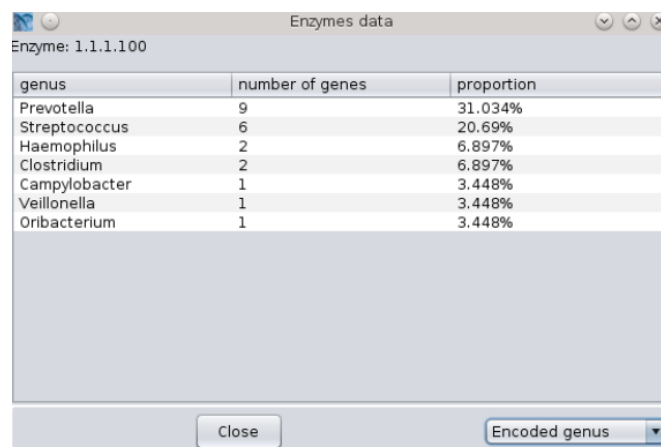
the '*Taxonomy*' sub-view described earlier, otherwise Merlin will throw an error.

Info	Names	ECnumber	Nº of reacti...	Nº of genes	Abundance	Nº of genes encoded by genus	Nº of genes with no genus
alcohol:NAD+ ...	1.1.1.1	19	3	4.0E-4	No genus minimum score	3 (100.0%)	
(3R)-3-hydroxy...	1.1.1.100	11	29	0.0035	22 (75.862%)	7 (24.138%)	
dTDP-6-deoxy...	1.1.1.133	1	17	0.0021	16 (94.118%)	1 (5.882%)	
(S)-3-hydroxyb...	1.1.1.157	3	4	5.0E-4	2 (50.0%)	2 (50.0%)	
Transferred to...	1.1.1.158	0	30	0.0036	22 (73.333%)	8 (26.667%)	
7alpha-hydrox...	1.1.1.159	1	1	1.0E-4	1 (100.0%)	0 (0.0%)	
(R)-pantoate...	1.1.1.169	1	3	4.0E-4	2 (66.667%)	1 (33.333%)	
S-amino-6(5-...	1.1.1.193	1	12	0.0015	11 (91.667%)	1 (8.333%)	
IMP:NAD+ ovid...	1.1.1.205	2	42	0.0051	37 (88.095%)	5 (11.905%)	
morphine:NAD...	1.1.1.218	2	1	1.0E-4	1 (100.0%)	0 (0.0%)	
UDP-alpha-D-g...	1.1.1.22	1	5	6.0E-4	4 (80.0%)	1 (20.0%)	
L-histidinol:NA...	1.1.1.23	3	10	0.0012	9 (90.0%)	1 (10.0%)	
shikimate:NAD...	1.1.1.25	1	15	0.0018	12 (80.0%)	3 (20.0%)	
4-phosphono...	1.1.1.262	3	5	6.0E-4	4 (80.0%)	1 (20.0%)	
2-C-methyl-D-e...	1.1.1.267	1	26	0.0032	24 (92.308%)	2 (7.692%)	
(S)-lactate:NA...	1.1.1.27	3	4	5.0E-4	4 (100.0%)	0 (0.0%)	
GDP-beta-L-fu...	1.1.1.271	3	7	8.0E-4	7 (100.0%)	0 (0.0%)	
(R)-2-hydroxyc...	1.1.1.272	5	1	1.0E-4	1 (100.0%)	0 (0.0%)	
(R)-lactate:NA...	1.1.1.28	1	1	1.0E-4	1 (100.0%)	0 (0.0%)	
D-glycerate:N...	1.1.1.29	2	4	5.0E-4	3 (75.0%)	1 (25.0%)	
L-homoserine:...	1.1.1.3	2	15	0.0018	12 (80.0%)	3 (20.0%)	
(R)-3-hydroxya...	1.1.1.36	2	1	1.0E-4	No genus minimum score	1 (100.0%)	
(S)-malate:NA...	1.1.1.37	1	5	6.0E-4	3 (60.0%)	2 (40.0%)	
(S)-malate:NA...	1.1.1.38	2	14	0.0017	11 (78.571%)	3 (21.429%)	
(R,R)-butane...	1.1.1.4	1	1	1.0E-4	1 (100.0%)	0 (0.0%)	
(S)-malate:NA...	1.1.1.40	2	17	0.0021	17 (100.0%)	0 (0.0%)	
isocitrate:NAD...	1.1.1.41	1	1	1.0E-4	No genus minimum score	1 (100.0%)	
isocitrate:NAD...	1.1.1.42	3	3	4.0E-4	2 (66.667%)	1 (33.333%)	

Figure 15: Merlin's view for metagenomic enzymes information.

The '*info*' column provides detailed information about the selected enzyme in three different ways: in a table, the most important one, it shows the genera encoding the enzyme (Figure 16); another table displays the genes (locus tag) encoding the enzyme as well as their taxonomic assignments; the last one exhibits the reactions assigned to the selected enzyme (Figure 17)

The '*Enzymes coverage*' button allows the user to export the enzymes coverage (presence/absence) to a tab-separated text file. In the '*Entity view*' tab in the down side of the panel the main statistics of the enzymes are shown (e.g number of enzymes from each class)(Figure 18).

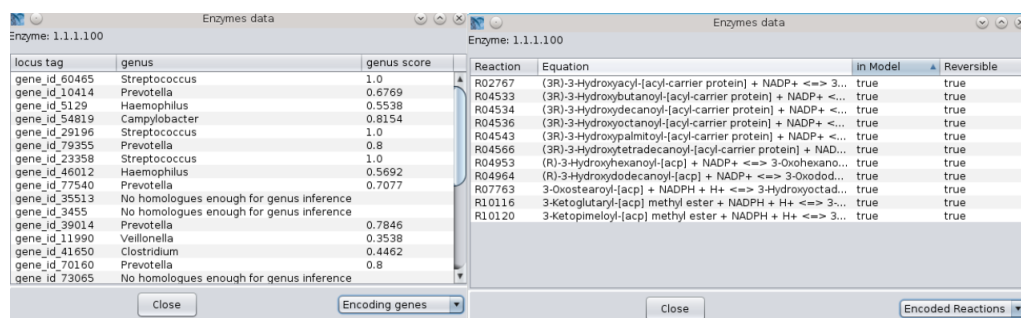


Enzyme: 1.1.1.100

genus	number of genes	proportion
Prevotella	9	31.034%
Streptococcus	6	20.69%
Haemophilus	2	6.897%
Clostridium	2	6.897%
Campylobacter	1	3.448%
Veillonella	1	3.448%
Oribacterium	1	3.448%

Buttons: Close, Encoded genus

Figure 16: Merlin's detailed information for different genera encoding a selected enzyme.



Enzyme: 1.1.1.100

locus tag	genus	genus score
gene_id_60465	Streptococcus	1.0
gene_id_10414	Prevotella	0.6769
gene_id_5129	Haemophilus	0.5538
gene_id_54819	Campylobacter	0.8154
gene_id_29196	Streptococcus	1.0
gene_id_79355	Prevotella	0.8
gene_id_23358	Streptococcus	1.0
gene_id_46012	Haemophilus	0.5692
gene_id_77540	Prevotella	0.7077
gene_id_35513	No homologues enough for genus inference	
gene_id_3455	No homologues enough for genus inference	
gene_id_39014	Prevotella	0.7846
gene_id_11990	Veillonella	0.3538
gene_id_41650	Clostridium	0.4462
gene_id_70160	Prevotella	0.8
gene_id_73065	No homologues enough for genus inference	

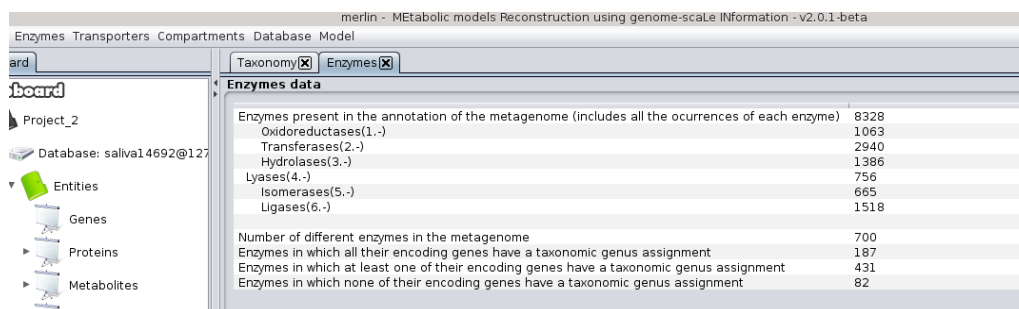
Buttons: Close, Encoding genes

Enzyme: 1.1.1.100

Reaction	Equation	In Model	Reversible
R02767	(3R)-3-Hydroxyacyl-[acyl-carrier protein] + NADP+ <=> 3...	true	true
R04533	(3R)-3-Hydroxybutanoyl-[acyl-carrier protein] + NADP+ <...	true	true
R04534	(3R)-3-Hydroxydecanoyl-[acyl-carrier protein] + NADP+ <...	true	true
R04536	(3R)-3-Hydroxyoctanoyl-[acyl-carrier protein] + NADP+ <...	true	true
R04543	(3R)-3-Hydroxypalmitoyl-[acyl-carrier protein] + NADP+ <...	true	true
R04566	(3R)-3-Hydroxytetradecanoyl-[acyl-carrier protein] + NAD...	true	true
R04953	(R)-3-Hydroxyhexanoyl-[acp] + NADP+ <=> 3-Oxohexano...	true	true
R04964	(R)-3-Hydroxydodecanoyl-[acp] + NADP+ <=> 3-Oxodod...	true	true
R07763	3-Oxostearoyl-[acp] + NADPH + H+ <=> 3-Hydroxyoctad...	true	true
R10116	3-Ketoglutaroyl-[acp] methyl ester + NADPH + H+ <=> 3...	true	true
R10120	3-Ketopimeloyl-[acp] methyl ester + NADPH + H+ <=> 3...	true	true

Buttons: Close, Encoded Reactions

Figure 17: Detail windows for the 'Enzymes' main view (Genes on the left, Reactions on the right).



merlin - Metabolic models Reconstruction using genome-scale INformation - v2.0.1-beta

Enzymes Transports Compartments Database Model

ard

Taxonomy Enzymes

board

Project_2

Database: saliv14692@127

Entities

- Genes
- Proteins
- Metabolites

Enzymes data

Enzymes present in the annotation of the metagenome (includes all the occurrences of each enzyme)	8328
Oxidoreductases(1.-)	1063
Transferases(2.-)	2940
Hydrolases(3.-)	1386
Lyases(4.-)	756
Isomerases(5.-)	665
Ligases(6.-)	1518
Number of different enzymes in the metagenome	700
Enzymes in which all their encoding genes have a taxonomic genus assignment	187
Enzymes in which at least one of their encoding genes have a taxonomic genus assignment	431
Enzymes in which none of their encoding genes have a taxonomic genus assignment	82

Figure 18: Statistics of the metagenomics enzymes entity.

2.7 Pathways characterization

2.7.1 Methodology

This section describes the methodology implemented for the classification of functional pathways in complex microbial communities. The main goal here focuses on finding the pathways that are effectively present in the community as a whole, and then find out which organisms may be involved.

When the user performs the loading of KEGG data, pathways information is integrated into the internal database. This information includes the complete enzymes within each pathway, as well as their reactions, amongst others. This is the basis for the routine for metagenomics pathway inference, that is divided in three main stages:

1. Test whether a pathway is effectively present;
2. If so, calculate its abundance;
3. Assign taxonomic information to pathways.

Concerning the first step, this method classifies a pathway as present using hypergeometric tests (equation 7). This test calculates the probability that the number of enzymes observed in an enzymes list that compose a pathway occurred by chance. Therefore, given a pathway pt with n enzymes (where n is higher than 3), its probability P is given by:

$$P_{pt} = \frac{\binom{E}{e} \binom{N-E}{n-e}}{\binom{N}{n}} \quad (7)$$

where N is the population size of the enzymes that compose all the pathways, E refers to all the enzymes observed in the metagenome and e indicates the number of encoded enzymes (successes) in pathway pt with n enzymes.

To test whether a pathway is statistically significant, *Merlin* compares the value of the p-value P with a threshold t defined by the user (default is 0.1). If P is smaller than t , it means that the probability of the observed situation be explained by chance is so low that the pathway is likely to be present, thus is classified accordingly:

$$f_w = \begin{cases} 1, & \text{if } P_{pt} \leq t \\ 0, & \text{otherwise} \end{cases}$$

where 1 indicates the presence of the pathway pt and 0 its absence.

Pathway abundance abd_p (equation 8), is calculated according to a methodology proposed by Abubucker et al [4], representing the average of the upper half enzyme abundances of the pathway, to be robust to low-abundance enzymes:

$$abd_p = \frac{1}{\frac{|p|}{2}} \sum_{i \in [p/2]} w_{i,p} \quad (8)$$

where $[p/2]$ stands for the most abundant half of enzymes.

The last stage of the routine tries to describe the genus that is more involved in each pathway. This step is only performed if the pathway is considered present (coverage greater than the threshold), otherwise it would not make sense to assign a genus to a pathway. To make sure that a given genus is operating a pathway, a conservative approach is followed where a genus is assigned to a pathway only if that genus encodes at least 75% of the annotated enzymes within the pathway. This information is pulled out from the enzymes routine, where the genera for each enzyme are discriminated. At the end, this method is able to provide which genera are executing each pathway as well as the opposite, which pathways each genus executes.

2.7.2 Merlin's operation mode for pathways

To access and execute the pathways inference routine in *Merlin*, the user must click on the '*Pathways*' view under the '*Metagenomics*' entity. The main view (Figure 19) shows the overall information for each pathway, such as the number of complete enzymes that compose it, the number of enzymes from each pathway that were annotated in the sample, the obtained p-value in the hypergeometric test, the pathway abundance and the number of taxonomic genera executing each pathway.

A pathway is considered present if the column '*p-value*' displays a lower value than the threshold defined by the user in the '*Threshold for p-value*' text box. The user can set this parameter to be more or less stringent. Furthermore, he/she can also adopt a less conservative approach for genus assignment by

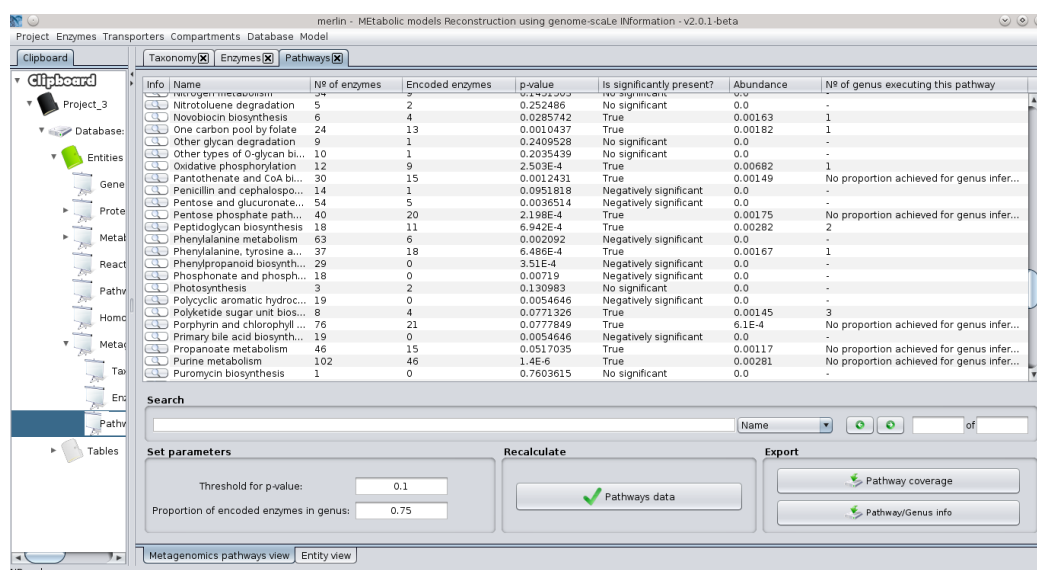


Figure 19: Merlin's view for metagenomic pathway information.

setting the '*Proportion of encoded enzymes in genus*' to a value lower than 0.75. It means that is possible to choose the proportion of the annotated genes in each pathway that a genus need to encode in order to assign that genus to a specific pathway. For example, if a proportion of 0.5 is selected and the genus *Escherichia* is being tested for pathway *p*, it is only necessary that this genus encodes half of the annotated enzymes to assume that *Escherichia* operates the pathway *p*. The main table can be re-calculated and updated with the new parameters by clicking on the '*Pathways data*' button.

The '*info*' column displays detailed information about the selected pathway, namely at the genus level, where it is shown the genus that executes that pathway (Figure 20), and at the enzyme/reactions level where encoded and no encoded enzymes, and encoded and no encoded reactions are shown, respectively (Figure 21).

Genus contributing for this pathway	Number of enzymes encoded	Is this genus executing this pathway?
Haemophilus	22	True
Prevotella	21	False
Campylobacter	20	False
Veillonella	20	False
Streptococcus	19	False
Fusobacterium	17	False
Bacillus	8	False
Clostridium	6	False
Bacteroides	6	False
Atopobium	5	False
Enterococcus	5	False
Megasphaera	5	False
Oribacterium	4	False
Selenomonas	4	False
Peptostreptococcus	3	False
Porphyromonas	3	False

Figure 20: Merlin's detailed information for genera operating a selected pathway.

Encoded Enzymes	Non encoded Enzymes
2.1.2.9	2.5.1.73
2.9.1.1	2.7.1.164
6.1.1.1	2.9.1.2
6.1.1.10	6.1.1.23
6.1.1.11	6.1.1.24
6.1.1.12	6.1.1.26
6.1.1.14	6.1.1.27
6.1.1.15	6.3.5.6
6.1.1.16	6.3.5.7
6.1.1.17	
6.1.1.18	
6.1.1.19	
6.1.1.2	
6.1.1.20	
6.1.1.21	
6.1.1.22	

Encoded Reactions	Non encoded Reactions
R03659	R08224
R03662	R08578
R03654	R03905
R03940	R03651
R03652	R03647
R03660	R08576
R03650	R08577
R08219	R04212
R03665	R08223
R03657	
R03648	
R03663	
R05578	
R02918	
R03646	
R03664	

Figure 21: Detail windows for the 'Pathways' main view (Enzymes on the left, Reaction on the right).

Regarding the export options, Merlin allows to export the pathways coverage (presence/absence) to a tab-separated text file by clicking on the '*Pathways coverage*' button . Selecting the '*Pathway/Genus info*' button, a list of pathways operated by each taxonomic genus is exported to a text file. In the '*Entity view*' tab in the down side of the panel the main statistics of the pathways routine are shown(Figure 22).

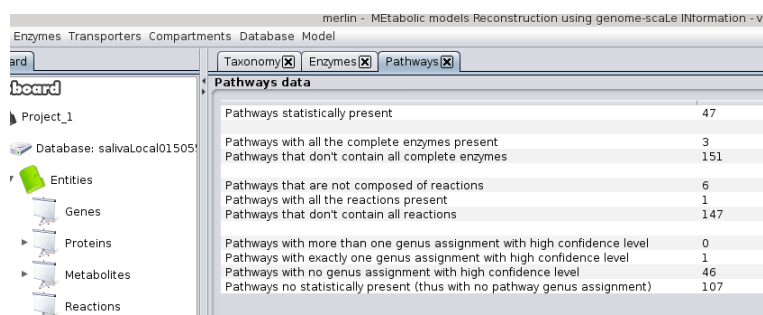


Figure 22: Statistics of the metagenomics pathways entity.

3 Using more memory

You can specify the amount of memory that Merlin can use. The default maximum amount of memory is 4GB but it is recommended to increase this value accordingly the computer capabilities. To do so, open the running script with a text editor and change the current memory specification (e.g. `MAXMEM="-Xmx4G"`) to a new value.

4 Contact

If you have any doubt and/or suggestion about metagenomic analysis with Merlin feel free to contact me: psbpedrobarbosa@gmail.com.

References

- [1] O. Dias, M. Rocha, F. Eugenio, and et al. "Merlin : Metabolic Models Reconstruction using Genome-Scale Information". In: *Computer Applications in Biotechnology* 11.1 (2010), pp. 120–125.
- [2] D Glez-Peña, M Reboiro-Jato, P Maia, and et al. "AIBench: a rapid application development framework for translational research in biomedicine." In: *Computer methods and programs in biomedicine* 98.2 (May 2010), pp. 191–203.
- [3] O. Dias. "Reconstruction of the Genome-scale Metabolic Network of *Kluyveromyces Lactis*". PhD thesis. University of Minho, 2013.

- [4] S. Abubucker, N. Segata, J. Goll, and et al. “Metabolic reconstruction for metagenomic data and its application to the human microbiome.” In: *PLoS computational biology* 8.6 (Jan. 2012), e1002358.