BIOMÉRIEUX

BIONUMERICS®

version 8 - PLUGINS

Mycobacterium tuberculosis complex genotyping plugin

Contents

NOTES

SUPPORT BY APPLIED MATHS, A BIOMÉRIEUX COMPANY

While the best efforts have been made in preparing this manuscript, no liability is assumed by the authors with respect to the use of the information provided.

Applied Maths, a bioMérieux company, will provide support to research laboratories in developing new and highly specialized applications, as well as to diagnostic laboratories where speed, efficiency and continuity are of primary importance. Our software thanks its current status for a part to the response of many customers worldwide. Please contact us if you have any problems or questions concerning the use of BIONUMERICS®, or suggestions for improvement, refinement or extension of the software to your specific applications:

Applied Maths NV
Keistraat 120
9830 Sint-Martens-Latem
Belgium
PHONE: +32 9 2222 100
FAX: +32 9 2222 102
E-MAIL: BE-DAU-INFO@biomerieux.com
URL: https://www.bionumerics.com

Applied Maths, Inc.
11940 Jollyville Road, Suite 115N
Austin, Texas 78759
U.S.A.
PHONE: +1 512-482-9700
FAX: +1 512-482-9708
E-MAIL: US-DAU-INFO@biomerieux.com

LIMITATIONS ON USE

The BIONUMERICS® software, its plugin tools and their accompanying guides are subject to the terms and conditions outlined in the License Agreement. The support, entitlement to upgrades and the right to use the software automatically terminate if the user fails to comply with any of the statements of the License Agreement. No part of this guide may be reproduced by any means without prior written permission of the authors.

BIONUMERICS® uses following third-party software tools and libraries:

- Python 3.8 release from the Python Software Foundation, https://www.python.org/
- Xerces library for XML input and output from the Apache Software Foundation, https://xerces.apache.org/
- NCBI toolkit version 2.11.0, https://www.ncbi.nlm.nih.gov/BLAST/
- SRA Toolkit, https://ncbi.github.io/sra-tools/
- Boost c++ libraries, https://www.boost.org/
- Samtools for interacting with SAM / BAM files, https://www.htslib.org/download/
- 7-Zip (7za.exe), https://www.7-zip.org/
- Zlib library, https://zlib.net/
- Pigz for parallel gzip compression, https://zlib.net/pigz/
- Cairo 2D graphics library version 1.12.14, https://cairographics.org/
- Crypto++ library version 5.5.2, https://www.cryptopp.com/
- OpenSSL library, https://www.openssl.org/
- libSVM library for Support Vector Machines, https://www.csie.ntu.edu.tw/~cjlin/libsvm/
- SQLite version 3.7.17, https://www.sqlite.org/
- pymzML Python module version 2.4.7, https://github.com/pymzml/pymzML
- NumPy Python library version 1.19.1, https://www.numpy.org/
- BioPython Python library version 1.78, https://www.biopython.org/
- pyodbc Python module version 4.0.30, https://pypi.org/project/pyodbc/
- jinja2 Python library version 2.11.2, https://pypi.org/project/Jinja2/
- MarkupSafe Python library version 1.1.1, https://pypi.org/project/MarkupSafe/
- regex Python library version 2.5.91, https://pypi.org/project/regex/
- Chromium Embedded Framework, https://bitbucket.org/chromiumembedded/cef/wiki/Home
- SPAdes genome assembler version 3.15.3, https://bioinf.spbau.ru/spades *
- SKESA version 2.3.0, https://github.com/ncbi/SKESA/releases
- Unicycler version 0.5.0, https://github.com/rrwick/Unicycler/releases *
- Velvet for Windows, source code can be downloaded from https://www.bionumerics.com/download/open-source
- Bowtie2 version 2.2.5 (https://bowtie-bio.sourceforge.net/bowtie2/index.shtml)*
- SNAP version 2.0.0, https://www.microsoft.com/en-us/research/project/snap/
- RAxML version 8.2.11, https://github.com/stamatak/standard-RAxML/releases

- FastTree version 2.1.10, https://www.microbesonline.org/fasttree/
- CFSAN SNP pipeline version 2.2.0, https://github.com/CFSAN-Biostatistics/snp-pipeline *
- Prokka version 1.14.5, https://github.com/tseemann/prokka *
- sourmash version 4.1.0, https://github.com/dib-lab/sourmash **
- SeqSero2 for Windows, source code can be downloaded from https://www.bionumerics.com/download/open-source
- Fastp version 0.22.0, https://github.com/OpenGene/fastp

*: On Calculation Engine only **: See license conditions below

Sourmash license conditions:

Chapter 1

Introduction

The *MTBC genotyping plugin* for BIONUMERICS consists of four functionalities: ***Species***, ***Lineage***, ***Spoligotype*** and ***Resistance*** prediction. All functionalities start from raw sequence reads generated by WGS technologies such as Illumina or IonTorrent.

- *In silico* ***spoligotyping*** is performed using the nucleotide sequences of 43 spacers as references in a local mapping analysis (allowing for at most 1 mismatch between the reference spacer and the reads). Based on the spoligotyping octal code, the corresponding SB number is automatically retrieved from the *Mycobacterium bovis* spoligotype database (Mbovis.org).
- For ***species confirmation***, reads are broken down in kmers and compared with kmer lists of 16S rDNA sequences of 166 reference *Mycobacterium* species (see 7.1).
- For ***lineage*** and ***resistance*** prediction, sequence reads are first mapped onto the H37Rv reference genome with Bowtie 2 (version 2.2.5), followed by the detection of single nucleotide polymorphisms, insertions and deletions. Using this strategy, the tool is able to classify *Mycobacterium* strains belonging to the *Mycobacterium* complex in 8 known lineages (including Indo-Oceanic, East-Asian, East-African Indian, Euro-American, West-Africa 1/2 and a separate *M. bovis* clade) and 55 sub-lineages based on 62 known single nucleotide polymorphisms (SNPs) described by F. Coll ([1]). Lastly, resistance prediction is based on a list of mutations in known resistance genes (or promoter regions of these genes). On top of that, all these resistance genes can be screened for unknown mutations.

The pipeline is integrated within the BIONUMERICS client software, and implemented on a scalable high-throughput calculation environment. Results can be obtained within 2 to 60 minutes upon submission (depending on the coverage) and are stored into different experiment types and information fields inside your database.

Disclaimer: The results generated by the *MTBC functional genotyping plugin* are for research purposes only and not intended for clinical decision making.

The *MTBC functional genotyping plugin* is supported in the **BIONUMERICS-SEQ** and **BIONUMERICS-SUITE** configurations.

Chapter 2

Starting and setting up BIONUMERICS

2.1 Startup program

Make sure the latest version of BIONUMERICS is installed (https://www.bionumerics.com/download/software). The installation manual can be downloaded from https://www.bionumerics.com/download/manuals.

When BIONUMERICS is launched from the Windows start panel or when the BIONUMERICS shortcut ( ) on your computer's desktop is double-clicked, the **Startup program** is run. This program shows the *BIONUMERICS Startup* window (see Figure 2.1).

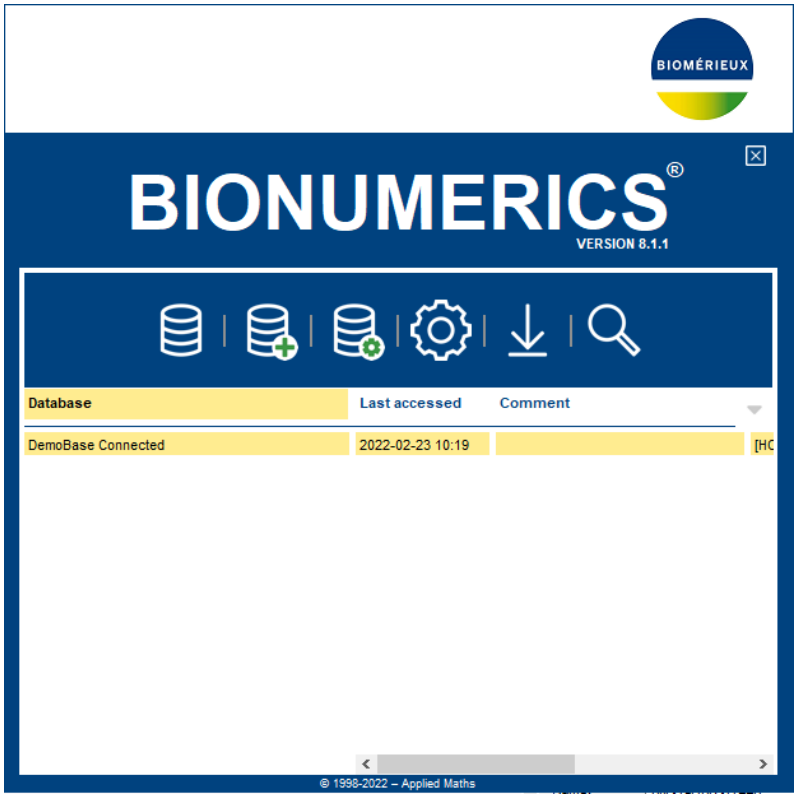

Figure 2.1: The *BIONUMERICS Startup* window.

A new BIONUMERICS database is created from the Startup program by pressing the button.

An existing database is opened in BIONUMERICS with or by simply double-clicking on a database name in the list.

2.2 Installation of the WGS tools plugin

The *MTBC genotyping plugin* requires the installation of the *WGS tools plugin* for assisting with import and launching jobs on the cloud-based computer environment (called from here forward the *Calculation Engine*). This plugin also allows you to submit other jobs, such as reference mapping for wgSNP analysis, de novo assembly and allele calling jobs for wgMLST analysis (see Figure 3.1). Tutorials for these analyses can be found on our website: https://www.bionumerics.com/tutorials.

Installing a plugin in a BIONUMERICS database is done from the *Plugins and Scripts* dialog box (see Figure 2.2), which can be called from the *Main* window by selecting ***File*** > ***Install / remove plugins...*** ( ).

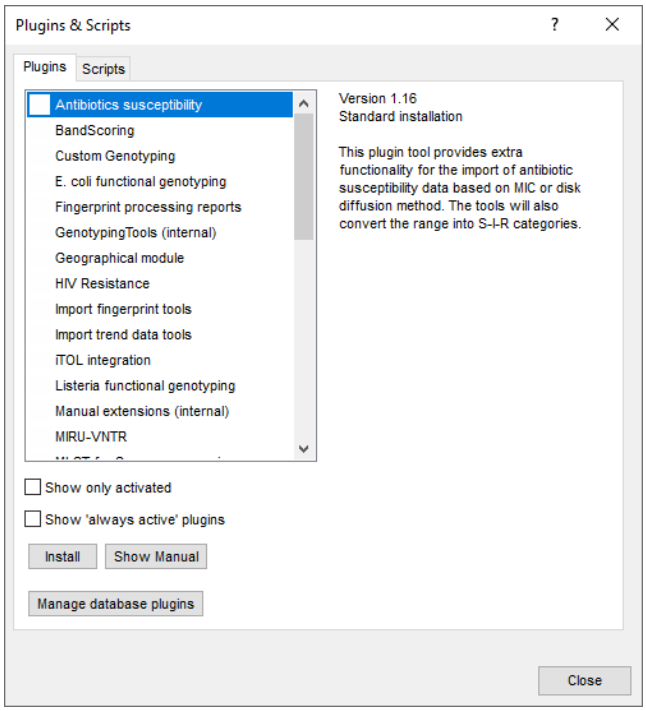

Figure 2.2: The *Plugins and Scripts* dialog box.

When a particular plugin is selected from the list of plugins, a short description appears in the right panel. If the selected plugin is documented, pressing <***Show Manual***> will open its manual in the *Help* window.

2.1 Select the *WGS tools plugin* from the list and press the <***Install***> button.

BIONUMERICS asks the user to confirm the installation of the *WGS tools plugin*. After confirmation, the plugin installation starts and the *WGS tools installation* wizard is shown.

2.2 Leave the option ***Use default Cloud Calculation Engine*** enabled and press <***Next***> in the *Calculation engine URL* wizard page.

2.3 Check the ***Enable running jobs on Cloud Calculation Engine*** and enter the credentials for your calculation engine project in the *Organism and project* wizard page (see Figure 2.3).

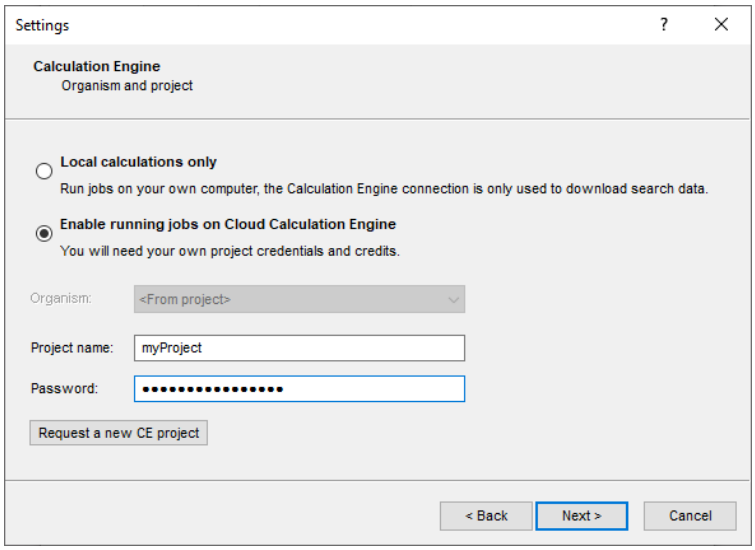

Figure 2.3: The *Organism and project* wizard page in the *WGS tools installation* wizard.

2.4 Press <***Next***> to start the synchronization with the specified allele database.

A confirmation dialog is displayed when the synchronization has been completed.

2.5 Press <***OK***>.

When the *WGS tools plugin* installation is complete, you will be prompted to restart the database. The *Plugins and Scripts* dialog box can be closed by pressing the <***Close***> button and the database via ***File*** > ***Exit***.

Open the database again from the *BIONUMERICS Startup* window. A ***WGS tools*** menu item is now available in the *Main* window (see Figure 2.4).

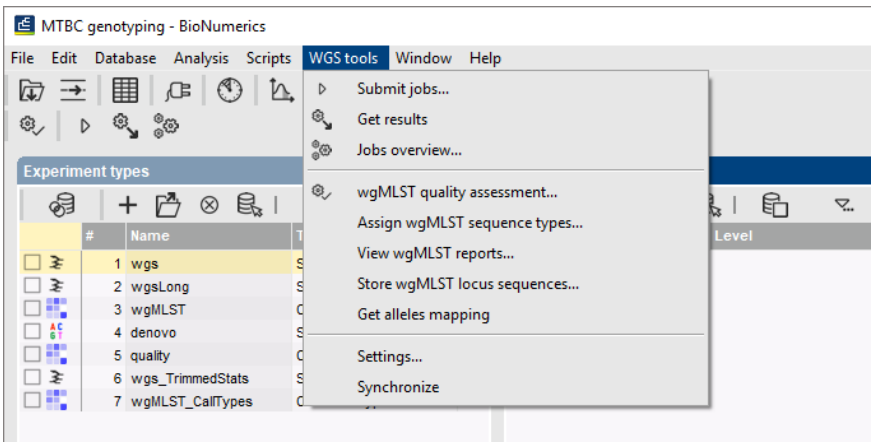

Figure 2.4: The ***WGS tools*** menu items.

Seven experiment types are created in the database (see Figure 2.5):

- **wgs**: This sequence read set experiment type is the experiment type which will contain the links to the short read sequence read files (see 3.1).
- The remaining 6 experiments: since these experiments are not used for *MTBC* genotyping

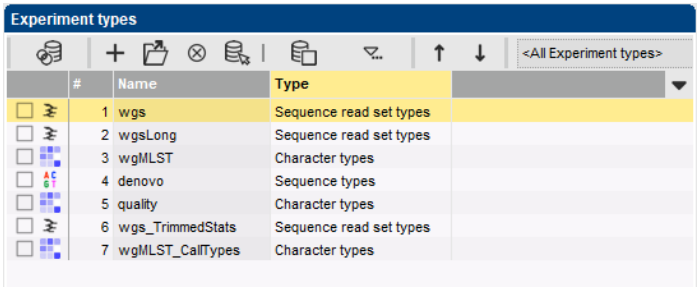

Figure 2.5: Experiment types created by the plugin.

analysis, these will not be covered in this manual. Please refer to the *WGS tools plugin* manual if interested in their further definition.

2.3 Installation of the MTBC genotyping plugin

To store the results of your genotyping jobs, you will have to create some new information fields and character/sequence experiments.

3.1 Make sure the *Database entries* panel is the active panel in the *Main* window and select ***Edit*** > ***Information fields*** > ***Add information field...***.

3.2 Enter a name, e.g. ***Species confirmation*** and choose ***Space*** next to ***Optimize storage for*** (see Figure 2.6). The latter is important because otherwise results can get truncated if they contain too much characters. All other settings can be left default.

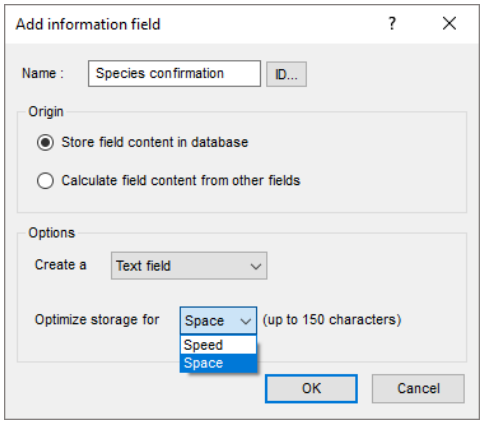

Figure 2.6: Adding a new information field with the ***Space*** option.

3.3 Repeat the previous step for the following information fields. Make sure you optimize the storage for ***Space*** each time:

- Unknown genotype
- Resistance summary
- Spoligotype (octal)
- SB number
- Lineage number

- Lineage name

Proceed as follows to install the *MTBC functional genotyping plugin*:

3.4 Call the *Plugins and Scripts* dialog box from the *Main* window with ***File*** > ***Install / remove plugins...*** ( ).

3.5 Select *MTBC functional genotyping* from the list and press the <***Install***> button.

3.6 Confirm the installation of the plugin.

3.7 In the *General tab* of the wizard (see Figure 2.7) choose ***wgs*** as ***WGS experiment type*** and the ***Info fields*** that will appear in the report. The ***Key*** is default shown in the report. Check for example the ***Species confirmation*** and ***Spoligotype (octal)*** information field or any other field that was added.

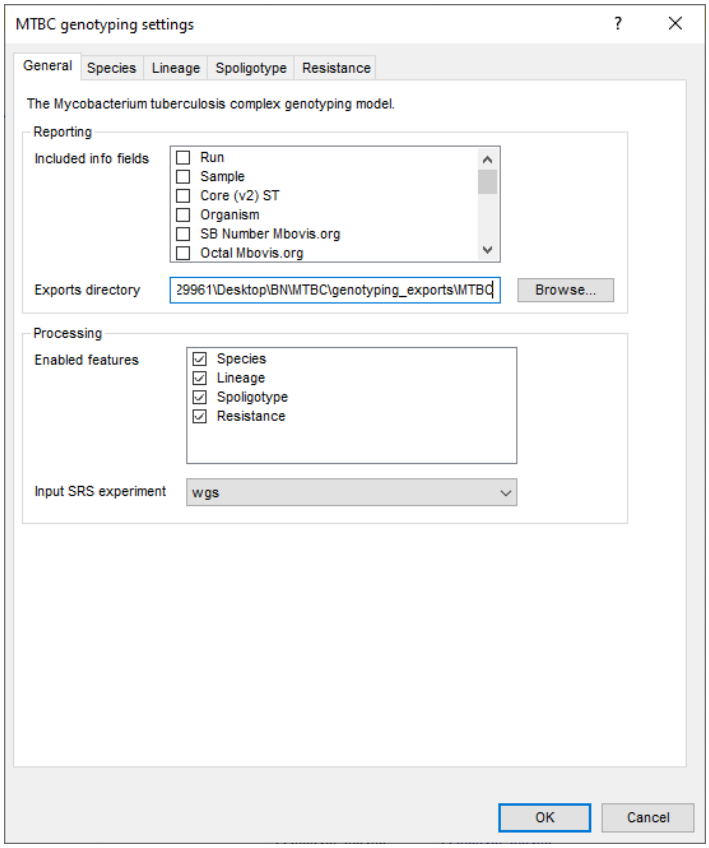

Figure 2.7: The *General tab*.

3.8 In the *Species tab* of the wizard choose the ***Species field*** for the storage of the species confirmation results and change the detection parameters if wanted (see Figure 2.8).

3.9 In the *Lineage tab* choose the fields for the storage of the ***Lineage number*** and ***Lineage name*** and change the detection parameters if wanted (see Figure 2.9).

3.10 In the *Spoligotype tab* select <***Create***> or select an existing experiment from the ***Spoligo presence/absence experiment type*** experiment type list, choose the fields for the storage of the octal code and SB number and modify the detection parameters if wanted (see Figure 2.10).

3.11 If <***Create***> was selected in the previous step, a dialog box appears prompting for the ***Spoligotype absence/present*** experiment type name. Accept the default proposed name or enter e.g. **Spoligotype** and click <***OK***>.

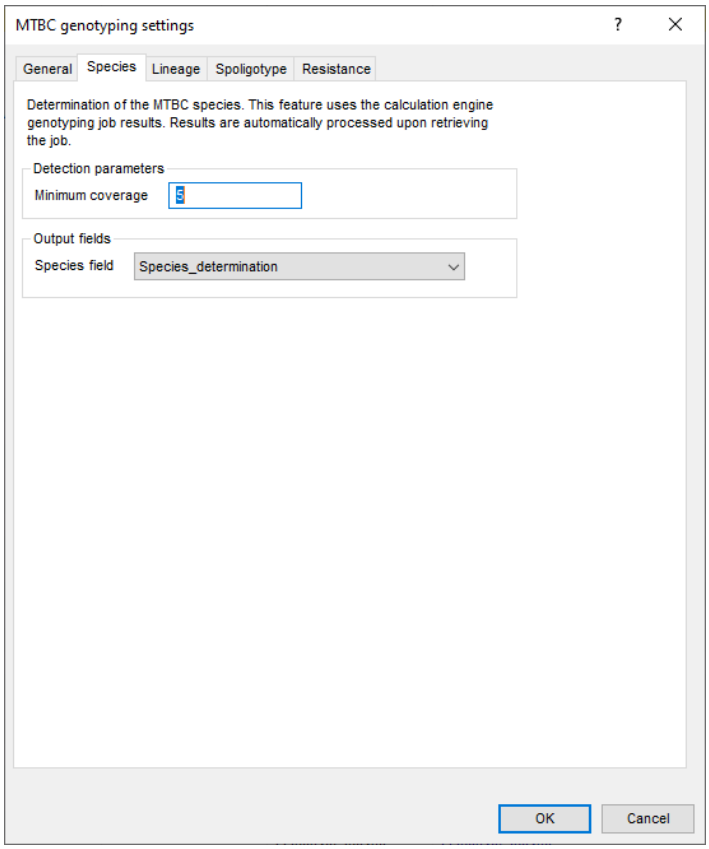

Figure 2.8: The *Species tab*.

3.12 In the *Resistance tab* (see Figure 2.11), select the ***Resistance database***. Currently, only one database is available: ***Curated database (version 6.0)***.

3.13 Select <***Create***> next to the three ***Resistance results*** experiment types or select existing experiments from the experiment types lists (see Figure 2.11).

3.14 Choose the ***Resistance summary*** information field for the storage of the antibiotics for which known resistance-related mutations (or indels) were found.

3.15 Choose the ***Unknown genotype*** information field for the storage of the antibiotics for which unknown mutations or indels (which are not included in the resistance database) were found in known resistance-related genes.

3.16 Check ***Detect genomic variants*** if you want to store all the mutations in resistance-related genes in an experiment type. Select <***Create***> or select an existing experiment from the ***Nucleotide variants*** and ***Amino acid variants*** experiment type lists.

3.17 Click <***OK***>.

3.18 If <***Create***> was selected in the previous step, a dialog box appears prompting for the experiment type name(s). Accept the default proposed names or choose your own and click <***OK***>.

When the *MTBC functional genotyping* installation is complete, you will be prompted to restart the database. The *Plugins and Scripts* dialog box can be closed by pressing the <***Close***> button and the database via ***File*** > ***Exit***.

Open the database again from the *BIONUMERICS Startup* window. A ***MTBC*** menu item is now available in the *Main* window (see Figure 2.12).

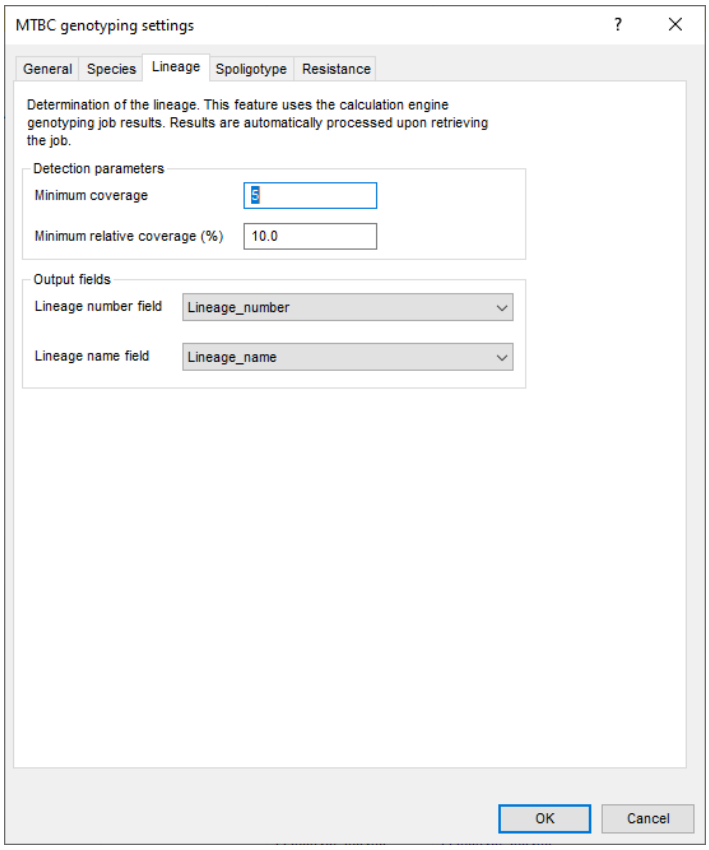

Figure 2.9: The *Lineage* tab.

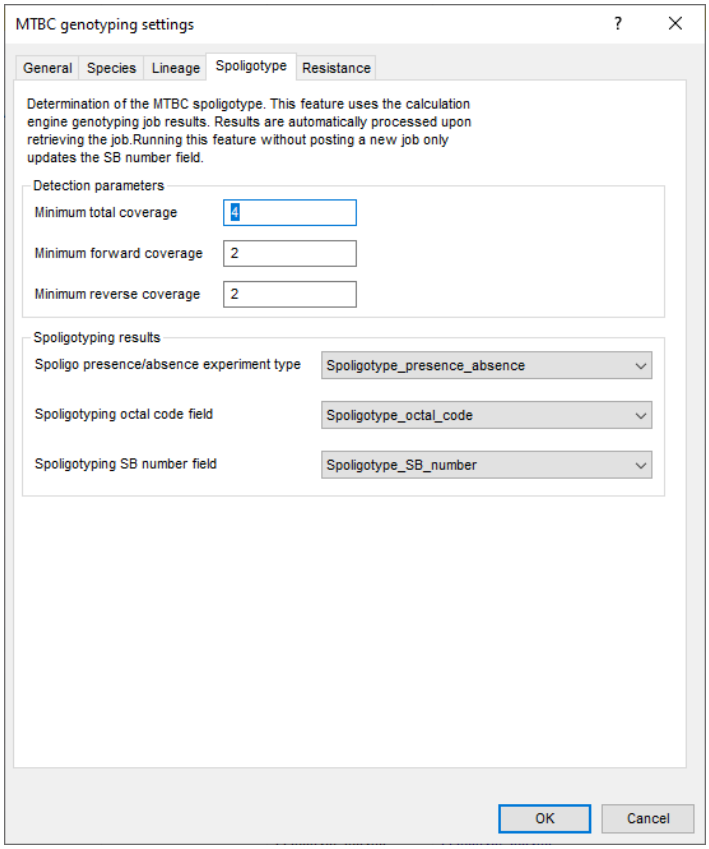

Figure 2.10: The *Spoligotype* tab.

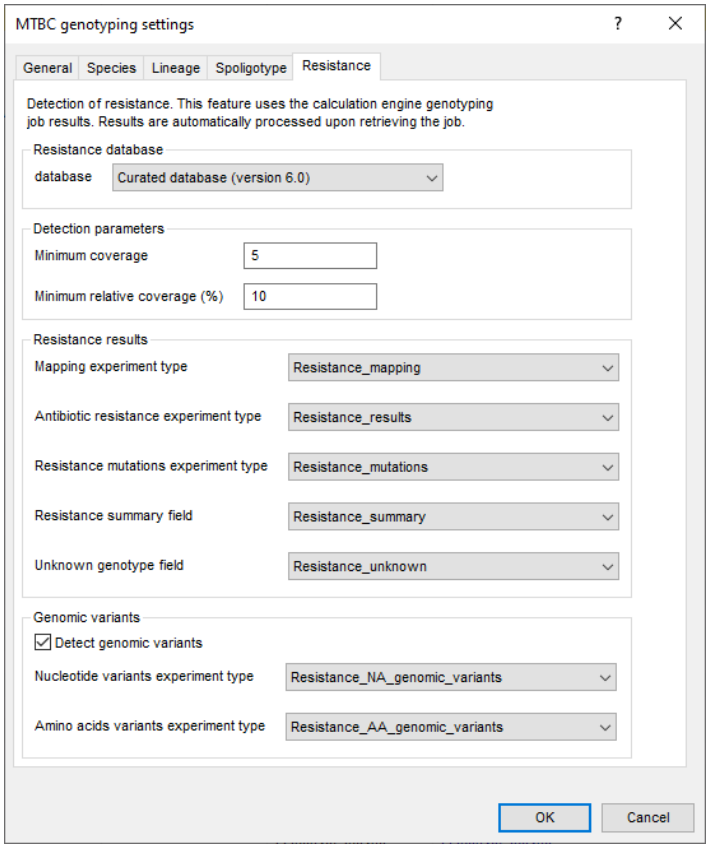

Figure 2.11: The *Resistance* tab.

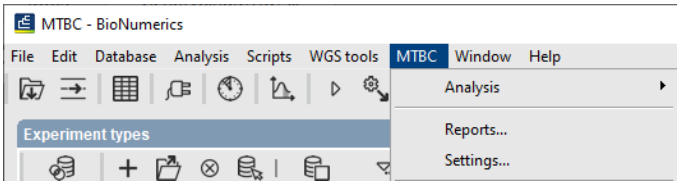

Figure 2.12: New menu items after installation of the plugin.

Chapter 3

MTBC genotyping in BIONUMERICS

To analyze samples with the *MTBC functional genotyping plugin*, following three steps need to be performed:

- Import sequence read sets (Illumina or IonTorrent FASTQ files with extension .fastq.gz (see 3.1))
- Submit an MTBC genotyping job to the CE (see 3.2)
- Import results (see 3.2)

3.1 Importing sequence read sets

To initiate analyses on your samples the raw sequence reads must be imported, preferably as links. This import option is only available after installation of the *WGS tools plugin* (See 2.2). The data *link* simply implies a link to an online repository (NCBI, EMBL-EBI, Amazon (S3) or BaseSpace) or local file server, and enables easy communication with the calculation engine. Another obvious advantage of this method is the disc space saved.

1.1 Select ***File*** > ***Import...*** (, **Ctrl+I**) to open the *Import data* wizard and to start the import of the sequence read sets.

1.2 Choose ***Import sequence read set data as links*** and click <***Next***> to start the *Import sequence read sets as links* wizard.

Following tutorials are available on our website covering the data and meta data import options:

- https://www.bionumerics.com/tutorial/importing-fastq-files-and-fastq-file-links.
- https://www.bionumerics.com/tutorial/importing-links-data-stored-sra-ena-amazon-s3-and-basespace.
- https://www.bionumerics.com/tutorial/adding-entry-information.

3.2 Submitting an MTBC genotyping job to the CE and import of results

Once the *MTBC functional genotyping plugin* is installed and the settings have been specified, an MTBC genotyping job (spoligotyping, resistance/lineage prediction and species prediction) can be

submitted.

2.1 Select a single entry in the *Database entries* panel by holding the **Ctrl**-key and left-clicking on the entry. Alternatively, use the **space bar** to select a highlighted entry or click the ballot box next to the entry.

2.2 In order to select a group of entries, hold the **Shift**-key and click on another entry.

2.3 Select ***WGS tools*** > ***Submit jobs...*** ( ) to open the *Submit jobs* dialog box.

From the *Submit jobs* dialog box, one can define which algorithms need to be run on the selected samples and as such, define and launch the related jobs on the calculation engine.

2.4 Select ***Calculation Engine***.

2.5 Check the box next to ***MTBC genotyping*** (see Figure 3.1). Note that this option is only available after successful installation of the plugin and closing and reopening of the database after installation.

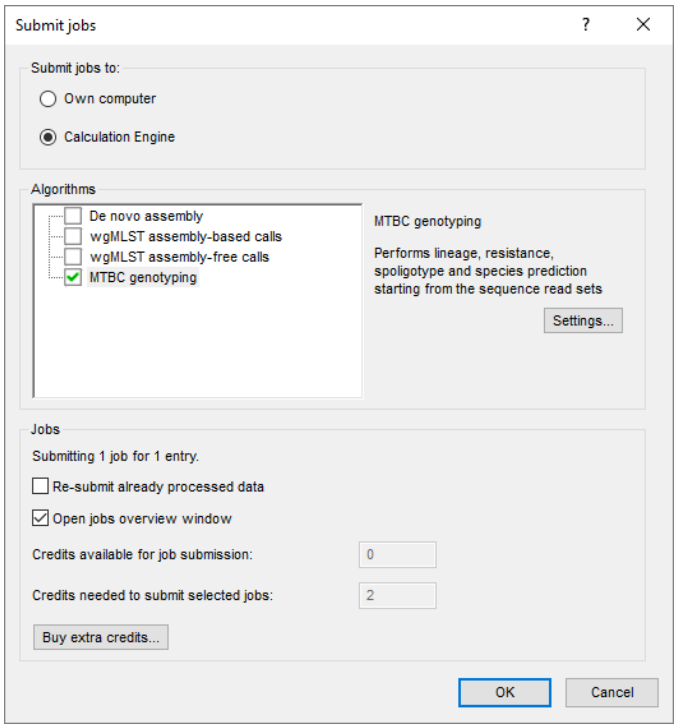

Figure 3.1: Submitting jobs to the calculation engine.

At this stage, you can also still change the settings by clicking the <***Settings***> button next to ***MTBC genotyping***.

2.6 Uncheck all other boxes if you do not want to perform any additional analyses (e.g. wgMLST).

Jobs that already have been submitted and have been imported successfully, will not be re-launched for analysis, unless the check box in front of ***Re-submit already processed data*** in the ***Jobs*** part is checked.

By default, the *Job overview* window will be opened after submission of the jobs. However, this can be changed by unchecking the option ***Open jobs overview window***.

To analyze one sample with the MTBC genotyping tool, you need 2 credits. The number of credits required to run the selected jobs for the selected entries can be consulted at the bottom of the

Submit jobs dialog box.

2.7 Click <***OK***> to launch the MTBC genotyping jobs and open the *Job overview* window for the calculation engine (see Figure 3.2).

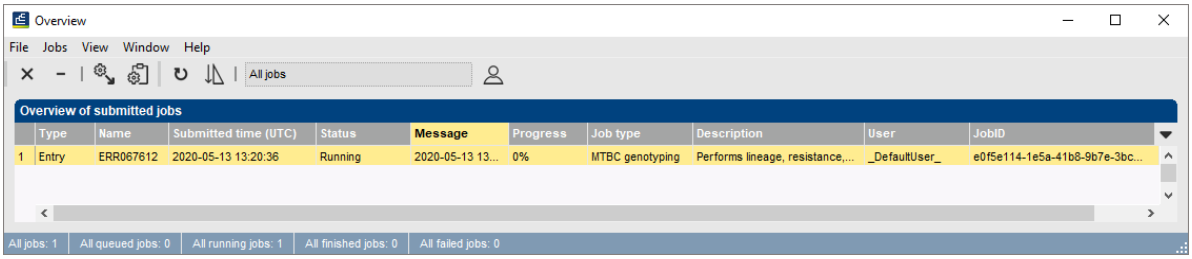

Figure 3.2: Job overview.

In the *Job overview* window you can see the status of the submitted jobs. The *Job overview* window can be opened from the *Main* window with ***WGS tools*** > ***Jobs overview...*** ( ).

2.8 Finished jobs can be imported with a manual action (***Jobs*** > ***Get results*** ( )) or through an automatic update: select ***File*** > ***Settings***, check both options and specify an interval (e.g. 10 min).

The job results can also be imported starting from the entry selection in the *Main* window:

2.9 Make an entry selection in the *Database entries* panel and select ***WGS tools*** > ***Get results*** ( ).

The job log files are saved in the *Job log* panel of the *Entry* window. Double-click on an entry in the *Database entries* panel to open the *Entry* window and to consult this information.

Once the results are imported, the corresponding jobs and their underlying data sets are automatically deleted from the calculation engine and as such, from the *Job overview* window.

3.3 Re-analysis of your samples with other settings

If you want to redo the lineage or resistance prediction of your samples with other settings, you do not need to submit a new job to the CE. This means that no additional credits will be charged:

1. Make new information fields to store the new lineage or resistance results (optional step).
2. Open the *Settings dialog box* with ***MTBC*** > ***Settings...*** in the *Main* window.
3. Select the new information fields and/or create new character experiments to store the new resistance results (optional step).
4. Select other settings.
5. Resistance and lineage prediction can be re-analyzed locally by clicking in the *Main* window on ***MTBC*** > ***Analysis*** > ***Resistance*** or ***MTBC*** > ***Analysis*** > ***Lineage*** respectively. If you want to redo both analysis, select ***MTBC*** > ***Analysis*** > ***All enabled***.

The options ***MTBC*** $>$ ***Analysis*** $>$ ***Resistance/lineage/all enabled*** can only be used when there is already a mapping present for the selected entries (this will only be the case if you have already submitted an MTBC genotyping job for this sample to the CE). Please note that only the mapping generated by the MTBC genotyping plugin can be used and not the standard mapping.

If you did not create/select new experiment types in the settings, results in the current experiment types will be overwritten when you re-analyze your samples.

If you click on ***MTBC*** $>$ ***Analysis*** $>$ ***Resistance***, and you also selected new, empty fields for lineage results, the lineage results obtained with the original settings will be used. If you also want to analyze the lineage with new settings, you have to click on ***MTBC*** $>$ ***Analysis*** $>$ ***Lineage*** or ***MTBC*** $>$ ***Analysis*** $>$ ***All enabled***.

Chapter 4

MTBC genotyping settings

The settings for the analyses can still be adjusted after installation of the plugin by clicking in the *Main* window on ***MTBC*** > ***Settings***. Alternatively, the settings can be changed during job submission by clicking on the <***Settings***> button next to the ***MTBC genotyping*** option (see Figure 3.1).

4.1 General settings

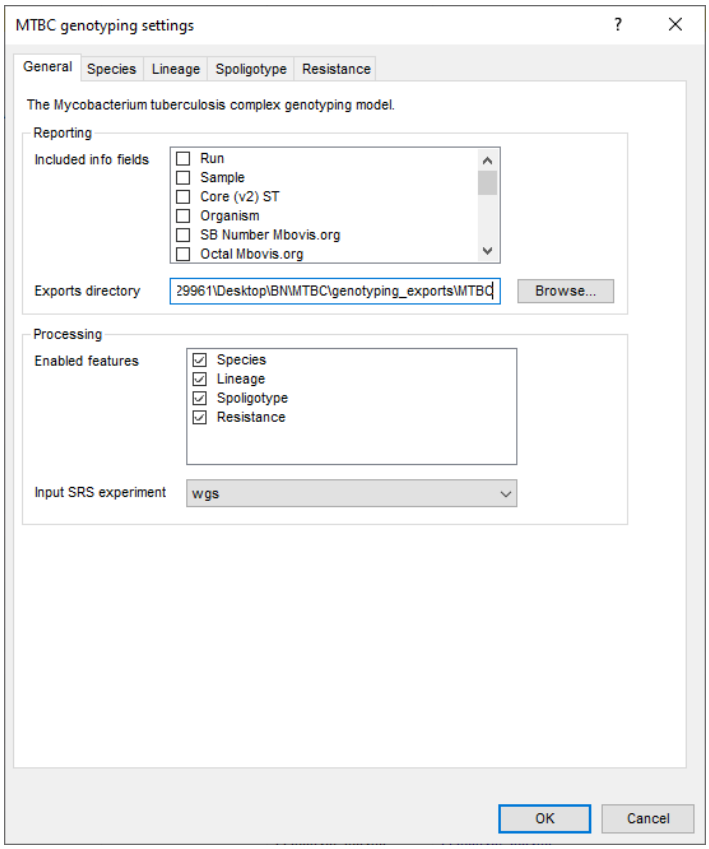

Figure 4.1: The *General tab*.

Input SRS experiment: Choose the experiment that contains your whole genome sequencing data.

When you have installed the *WGS tools plugin*, a ***wgs*** experiment type (sequence read set type) is automatically created. You can store all your short read sequence read data (NGS data) in here.

Included info fields: all information fields from your database (visible in the *Main* window) will be listed here. Check the ones you would like to see in the final report (see 5).

4.2 Settings for species prediction within the *Mycobacterium* genus

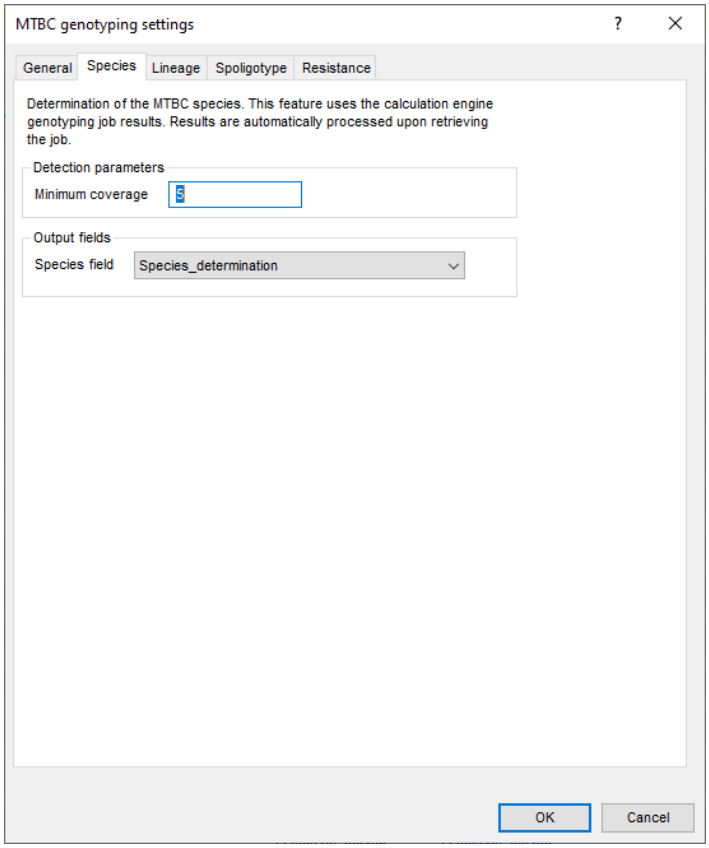

Figure 4.2: The *Species* tab.

Minimum coverage: Minimum count for the kmer to be reliably present in the data. If at least 95% of the kmers related to a species 16S rDNA sequence have a coverage equal to or greater than the minimum coverage, we consider the species to be present and list it in the output information field. As a consequence, multiple species can be listed in this field.

If you do not get a result for the species confirmation, it could help to lower the minimum coverage settings. If you get too much results (which is sometimes the case for mycobacterial samples not belonging to the MTBC complex), it could help to raise the minimum coverage settings (e.g. to 10).

Species field: Information field containing predicted species: e.g. MTBC complex, *Mycobacterium abscessus*, *Mycobacterium caprae*, etc.

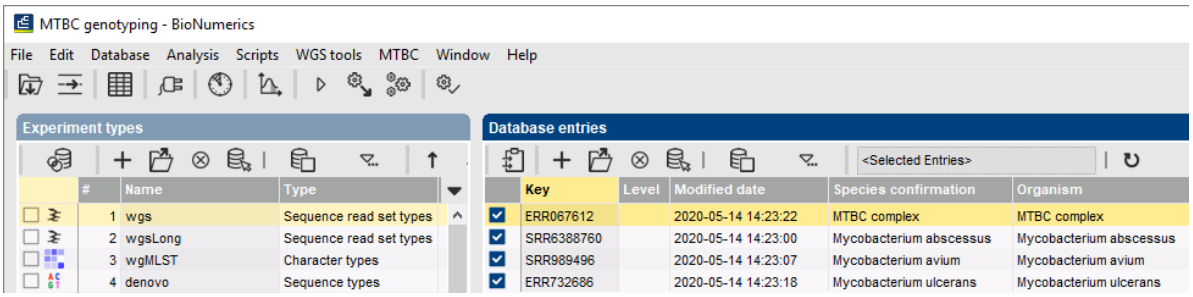

Figure 4.3: Example output of species prediction.

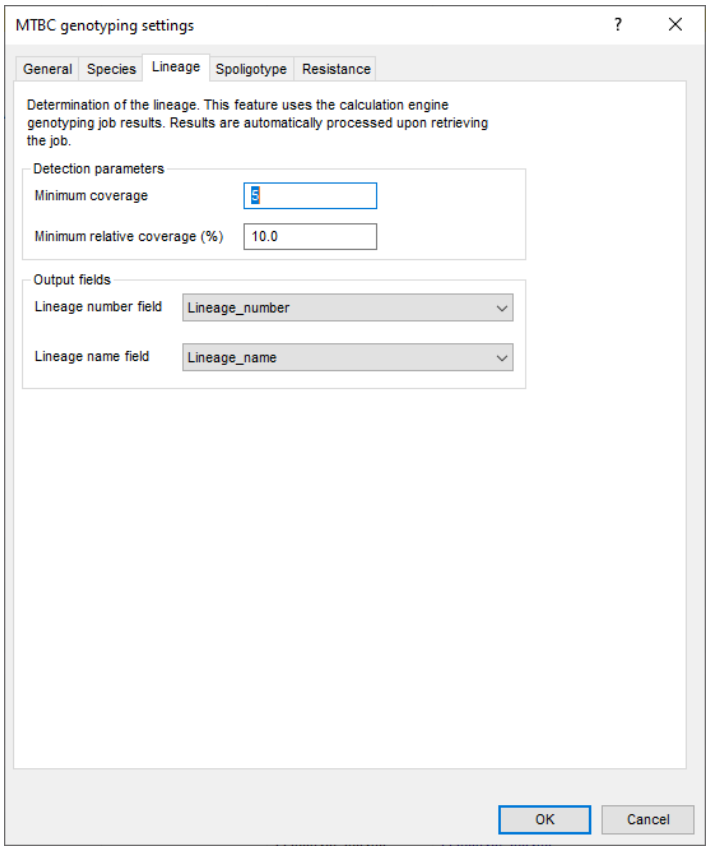

Figure 4.4: The *Lineage* tab.

4.3 Settings for lineage prediction of MTBC species

Minimum coverage: Minimum amount of reads (absolute coverage) that need to contain the specific lineage SNP.

Minimum relative coverage: the minimum percentage of reads that need to contain the specific lineage SNP at a certain genomic position. The relative coverage is calculated as the amount of reads with the specific lineage SNP divided by the total amount of reads at that position.

Lineage field: information field containing lineage number (e.g. 2 / 2.2 / 2.2.1).

Lineage name: Information field containing lineage name (e.g. East-Asian / East-Asian (Beijing)).

Lineage name and number are based on the detection of lineage-specific SNPs (defined by Coll and coworkers ([1])) that have an absolute and relative coverage higher than or equal to the specified minima.

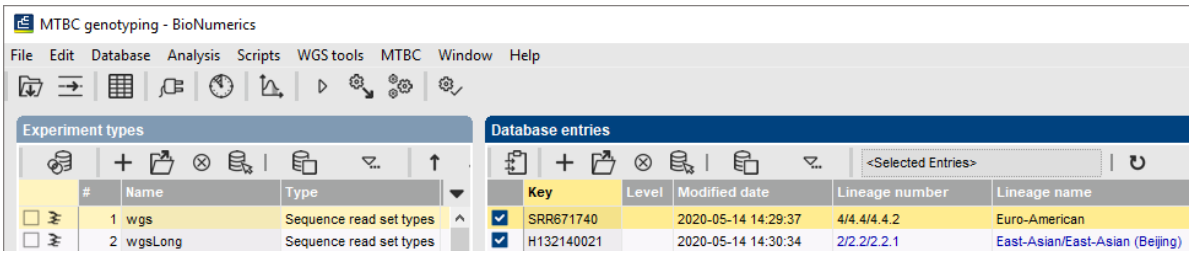

Figure 4.5: Example output of lineage prediction.

4.4 Settings for *in silico* Spoligotyping

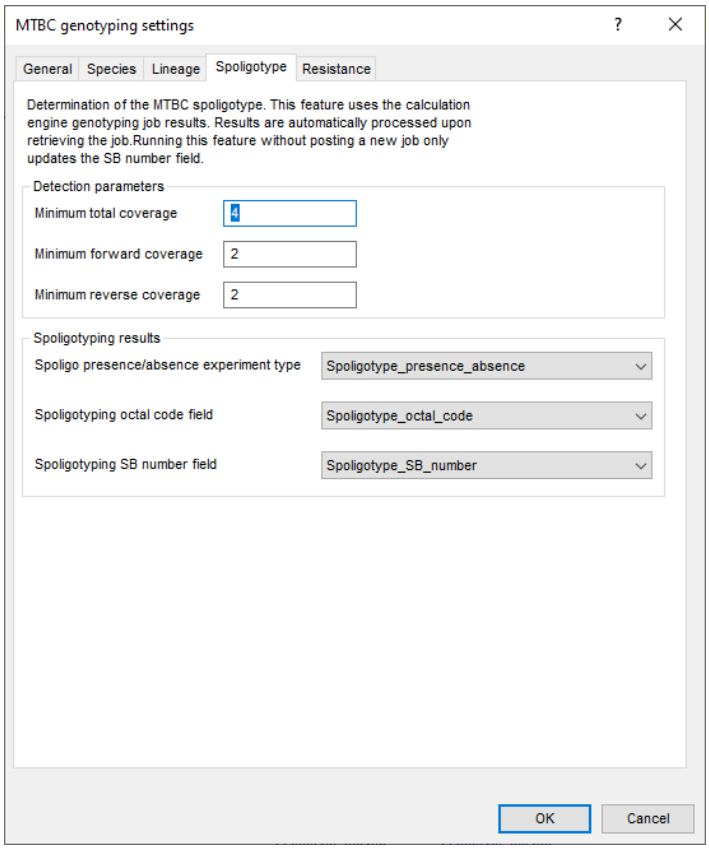

Figure 4.6: The *Spoligotype* tab.

A spacer (see 7.2) is considered present when all of its 21 bp kmers or all 21 bp kmers of a 1 bp variant of this sequence have sufficient forward coverage (determined by ***Minimum forward coverage*** parameter), reverse coverage (determined by ***Minimum reverse coverage*** parameter and total coverage (determined by ***Total coverage*** parameter).

The prediction of the presence/absence of the spacers in BIONUMERICS highly depends on the coverage settings. Using too high minimal reverse or forward coverages can lead to false negatives (the spacer is present in the genome but not picked up by the plugin) while too low coverages can lead to false positives. Depending on the settings, the *in silico* spoligotype can thus differ from the spoligotype derived from wet lab experiments.

Spoligo presence/absence experiment type: character experiment in which the binary code (for each of the 43 spacers: absence or presence) is stored (see Figure 4.7 and Figure 4.8).

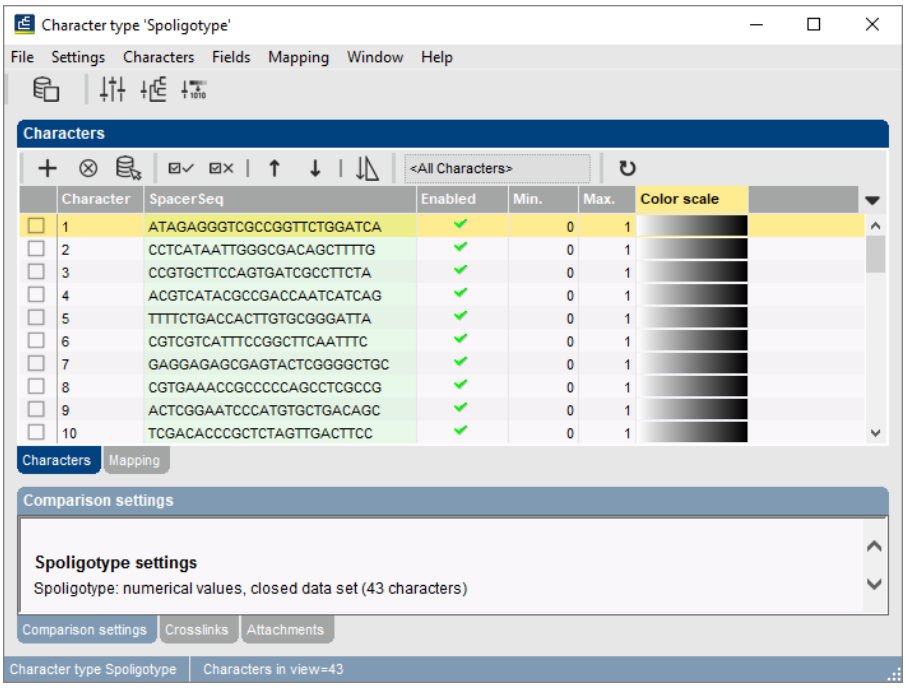

Figure 4.7: Spoligo presence/absence experiment type. This experiment contains 43 spacer sequences.

Figure 4.8: Example output of spoligotyping (spoligo presence/absence experiment type) for a sample with spoligotype 000000000003771. White = spacer is absent, black = spacer is present.

Spoligotyping octal code field: Information field containing the octal spoligotype code (e.g. 000000000003771) (see Figure 4.9).

Spoligotyping SB number field: Information field containing the spoligotype SB number (e.g. SB0040) (see Figure 4.9).

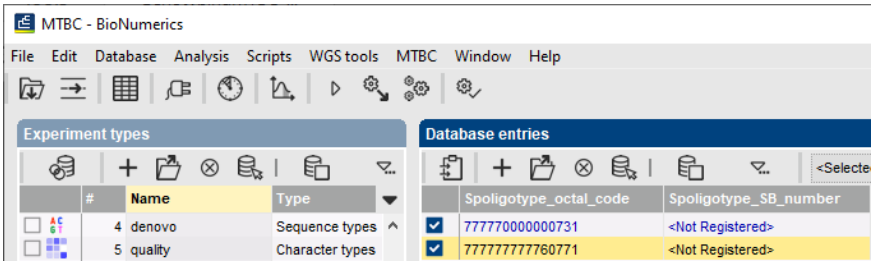

Figure 4.9: Example output of the spoligotyping octal code field and SB number field.

4.5 Settings for resistance prediction

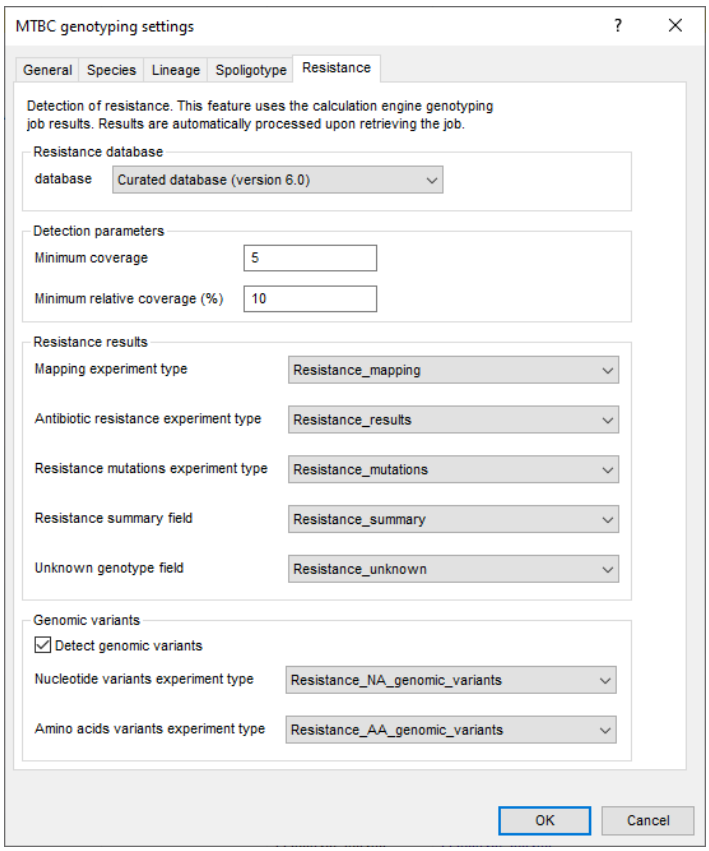

Figure 4.10: The *Resistance* tab.

The ***Resistance knowledgebase*** consists of a list of genomic mutations (single nucleotide polymorphisms, SNPs), insertions and deletions related to phenotypic resistance. Currently, there is one resistance database available:

Curated (version 6.0): This is the curated resistance knowledge base (KB) of bioMerieux and contains mutations related with resistance to Fluoroquinolone (general class), Moxifloxacin, Ofloxacin, levofloxacin, Capreomycin, Isoniazid, Amikacin, Rifampicin, Pyrazinamide, Ethionamide, Ethambutol, Kanamycin, Para-aminosalycilic acid, Streptomycin, Linezolid, Bedaquiline and Clofazimine. Initially, the library contained all resistance-related mutations extracted from Pointfinder, tbProfiler, MUBII, Biogram, ReseqTB, Mykrobe, PhyResSe and kvarQ. However, several mutations that are most likely not related to resistance according to our validation study and a subsequent literature review were removed from this list. Validation consisted of the comparison of *in silico* resistance prediction results and phenotypic drug susceptibility results for $>$1700 publically available MTBC samples.

Minimum coverage: Minimum amount of reads that need to contain the specific resistance SNP. For amino acids, the lowest coverage within the codon is considered.

Minimum relative coverage: Minimum percentage of reads that need to contain the specific resistance SNP (amino acid or nucleic acid) at a certain genomic position. The relative coverage is calculated as the amount of reads with the specific resistance SNP divided by the total amount of reads at that position.

Mapping experiment type: Sequence experiment in which the mapped sequence of your sample will be stored. This sequence has the same length as the H37Rv reference genome (4411532 bp)

(see Figure 4.11).

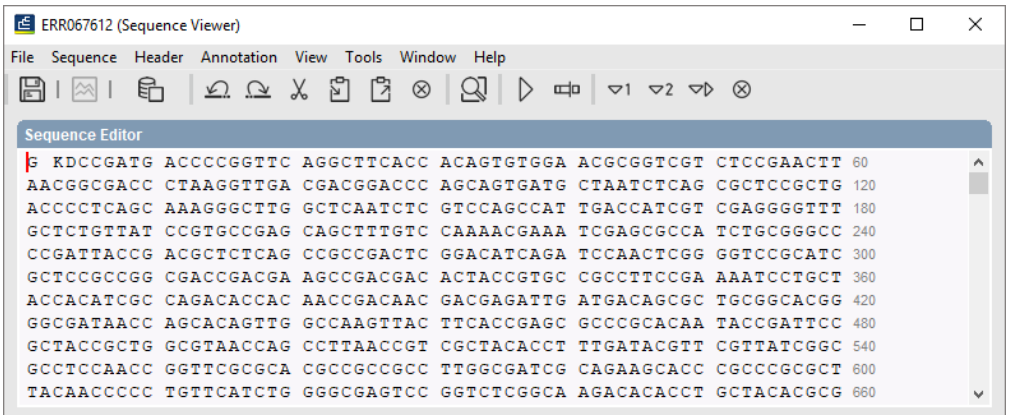

Figure 4.11: Example output of mapping experiment type for sample ERR067612.

Antibiotic resistance experiment type: character experiment containing all antibiotics from the selected resistance KB (see Figure 4.12).

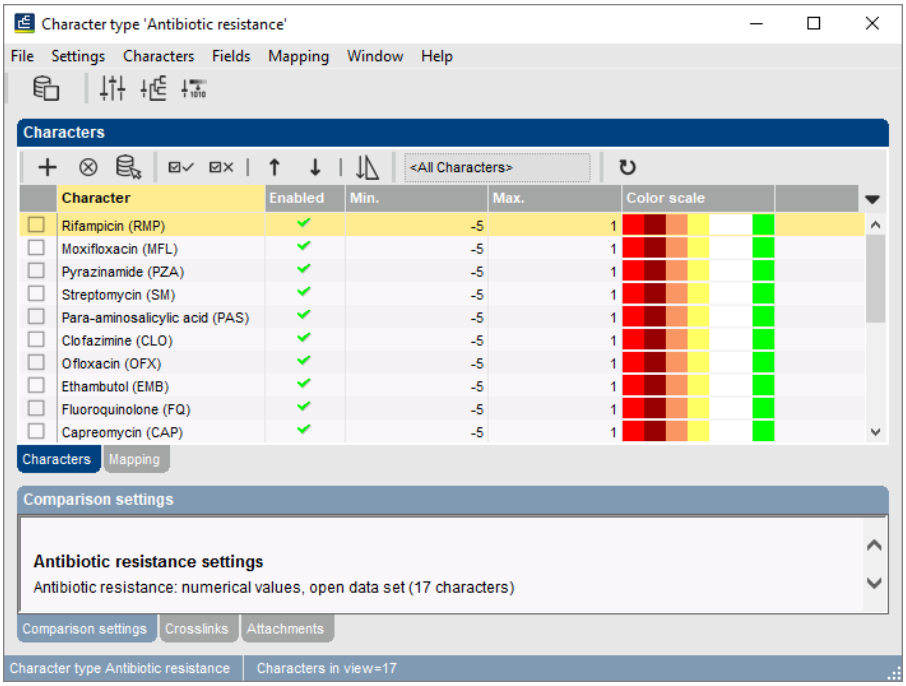

Figure 4.12: Antibiotic resistance experiment type.

The summary resistance call for an antibiotic can be (see Figure 4.13):

- ***Resistant (R; -5)***: at least 1 known resistance-related mutation or indel from the selected resistance KB was found with sufficient coverage.
- ***Unknown (U; -3)***: there are no known resistance-related mutations from the selected resistance KB found with sufficient coverage but there is at least 1 unknown mutation or indel found in a resistance-related gene. For positions already included in the resistance KB, every (non-synonymous) mutation or indel with sufficient coverage (as defined in the settings) leads to the 'unknown' status. For positions in resistance-related genes not yet included in the resistance KB, only majority/consensus mutations with more than 75% relative coverage

lead to the 'unknown' status. Lastly, if no bases are covered in resistance related genes for a particular antibiotic, the outcome is also 'unknown'.

- ***Failed (F; -2)***: no known or unknown mutations with sufficient coverage were found but also not all positions in the resistance-related genes were covered.
- ***Susceptible (S; +1)***: there are no known or unknown mutations found and all bases of the resistance-related genes were covered sufficiently.


ERR067612

Character	Value	Mapping
Rifampicin (RMP)	-5	R
Moxifloxacin (MFL)	-3	U
Pyrazinamide (PZA)	-5	R
Streptomycin (SM)	-5	R
Para-aminosalicylic ...	-3	U
Clofazimine (CLO)	1	S
Ofloxacin (OFX)	-3	U
Ethambutol (EMB)	-5	R
Fluoroquinolone (FQ)	-3	U
Capreomycin (CAP)	-3	U
Levofloxacin (LFX)	-3	U
Bedaquiline (BED)	1	S
Isoniazid (INH)	-3	U
Amikacin (AMK)	1	S
Kanamycin (KAN)	1	S
Ethionamide (ETH)	1	S
Linezolid (LZD)	1	S

Press Insert to add character

Figure 4.13: Example output of antibiotic resistance experiment type for sample ERR067612. U = Unknown, R = Resistant, F = Failed, S = Susceptible.

Sufficient coverage means that the absolute coverage is greater than or equal to the minimum coverage and the relative coverage is greater than or equal to the minimum relative coverage, as specified in the settings.

If not all positions from the resistance-related genes are sufficiently covered (an X sign will be present in the QC passed column in the quality metrics section of the summary report), but at least one resistance-related mutation with sufficient coverage was found or an unknown mutation was found, the call will be resistant (-5) or unknown (-3), respectively. If both unknown and known resistance mutations were found, the call will be resistant (-5).

Resistance mutation experiment type: character experiment containing all resistance-related positions from the selected resistance KB (see Figure 4.14).

Nomenclature of the characters from the curated resistance list (Figure 4.14):
Name:ProteinPosition:GenomicPosition:MutType.

- ***Name***: Name of the locus (or promotor).
- ***NucleotidePosition***: In case of insertions/deletions or promotor sequences.
- ***ProteinPosition***: Amino acid position in the protein product in case of coding gene sequences (this will be empty in case of e.g. promotor sequences or rRNA coding sequences).
- ***GenomicPosition***: Nucleotide position in H37Rv (only for non-coding sequences, insertions and deletions).
- ***MutType***: AA = amino acid substitution, NA = nucleotide substitution, IS = internal stop-codon, INS = insertion, DEL = deletion.

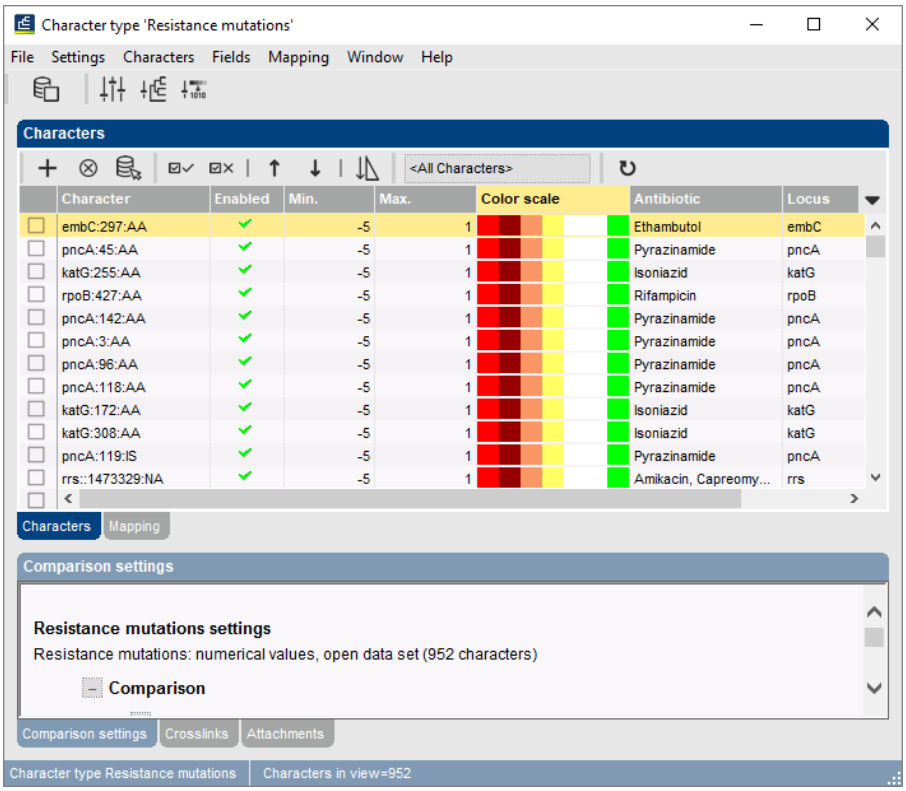

Figure 4.14: Resistance mutation experiment type.

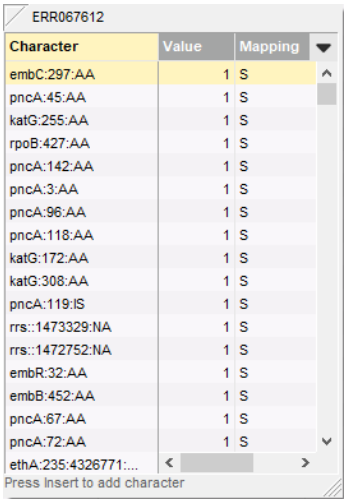

Figure 4.15: Example output of resistance mutations experiment type for sample ERR067612.

The resistance call for a resistance-related position can be (Figure 4.15):

- ***Resistant (R; -5)***: at least 1 known resistance-related mutation or indel from the selected resistance KB was found with sufficient coverage on this position.
- ***Unknown (U; -3)***: there are one or more variant calls with sufficient coverage at this position, but not one from the selected KB.
- ***Susceptible (S; +1)***: there are no known or unknown mutations with sufficient coverage at this position.

- ***No call (empty cell)***: if there is no base call with sufficient coverage.
- ***Missing (F; -1)***: if there is no coverage at all at this position.

Resistance summary field: Information field containing all antibiotics (abbreviations) for which resistance-related mutations (with sufficient coverage) from the selected resistance KB were found in resistance-related genes (see Figure 4.16).

Unknown genotype field: Information field containing all antibiotics (abbreviations) for which only unknown mutations (with sufficient coverage) were found in resistance-related genes (see Figure 4.16).

An additional information field called ***MDR_or_XDR*** is generated automatically during the first import of results (see Figure 4.16). MDR: multidrug resistance is defined as predicted resistance to INH and RMP. XDR: extensively drug resistance is defined as predicted resistance to INH, RMP, (CAP, KAN or AMK) and 1 of the FQ (Moxifloxacin, Ofloxacin, Levofloxacin or fluoroquinolone in general). No: not MDR or XDR.

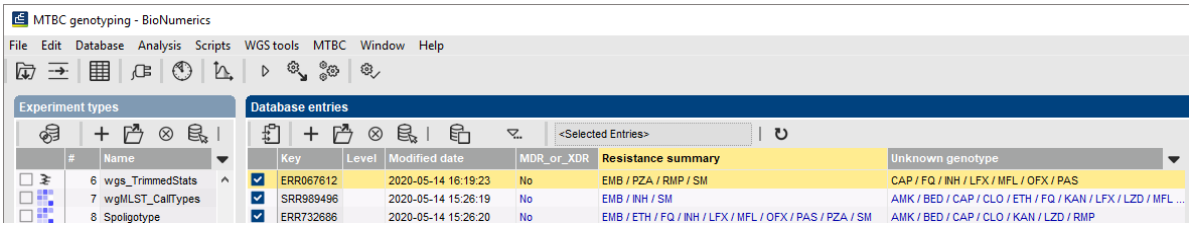

Figure 4.16: Example output of resistance summary and unknown genotype field.

Detect genomic variants: Check this box if you want to analyze and store all genomic variants in resistance-related genes in separate experiment types. If you do not need this information, it is advised to uncheck this box to speed up import of the results.

Nucleotide variants experiment type: Character experiment that contains all nucleotide mutations in resistance-related genes that have sufficient coverage. These include (new) mutations which are not in the selected resistance KB (see Figure 4.17).

The resistance call for a resistance-related position can be (Figure 4.18):

- A/C/T/G: nucleotide substitution.
- Multiple variants: there are multiple bases with sufficient coverage.
- Insertion: an insertion is present at this position.
- - : a deletion is present at this position.

Amino acid variants experiment type: Character experiment that contains all amino acid mutations in resistance-related genes that have sufficient coverage. These include (new) mutations which are not in the selected resistance KB.

The resistance call for a resistance-related position can be:

- An amino acid (one letter abbreviation).
- Multiple variants: there are multiple bases with sufficient coverage.
- Insertion: an insertion is present at this position.
- - : a deletion is present at this position.

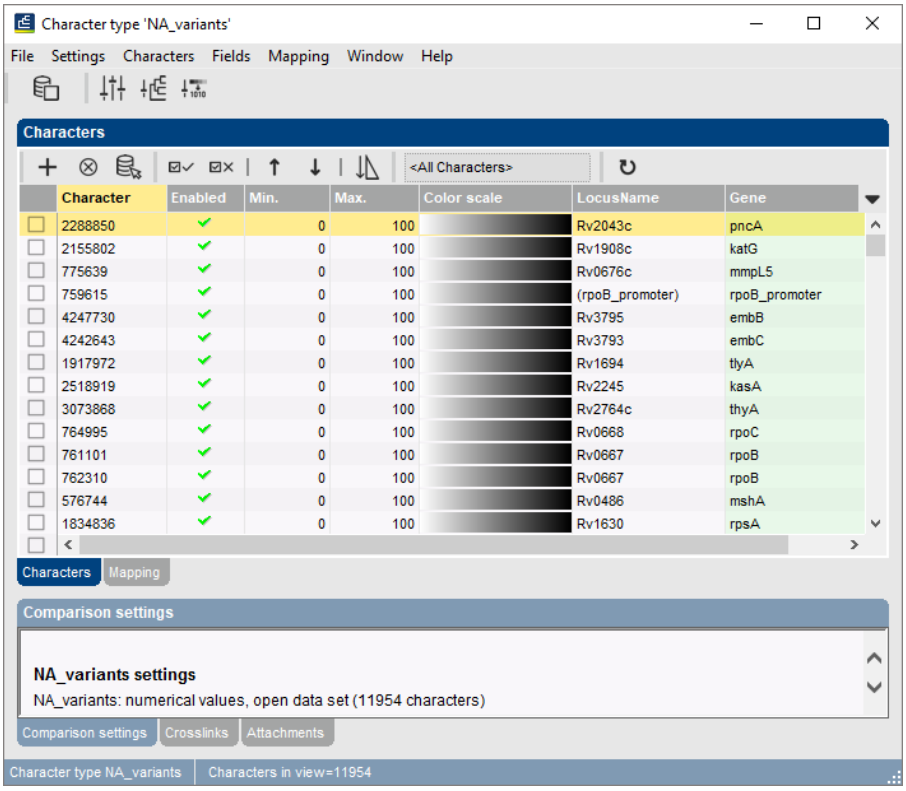

Figure 4.17: Nucleotide variants experiment type. Character: genomic position (H37Rv numbering).

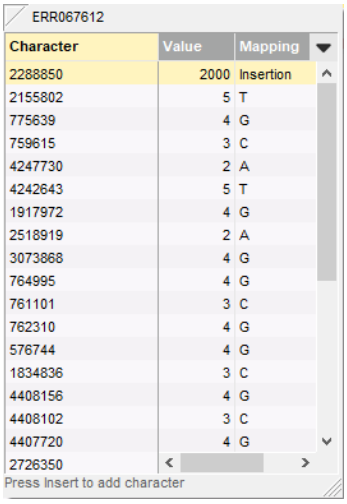

Figure 4.18: Example output of nucleotide variants experiment type for sample ERR067612.

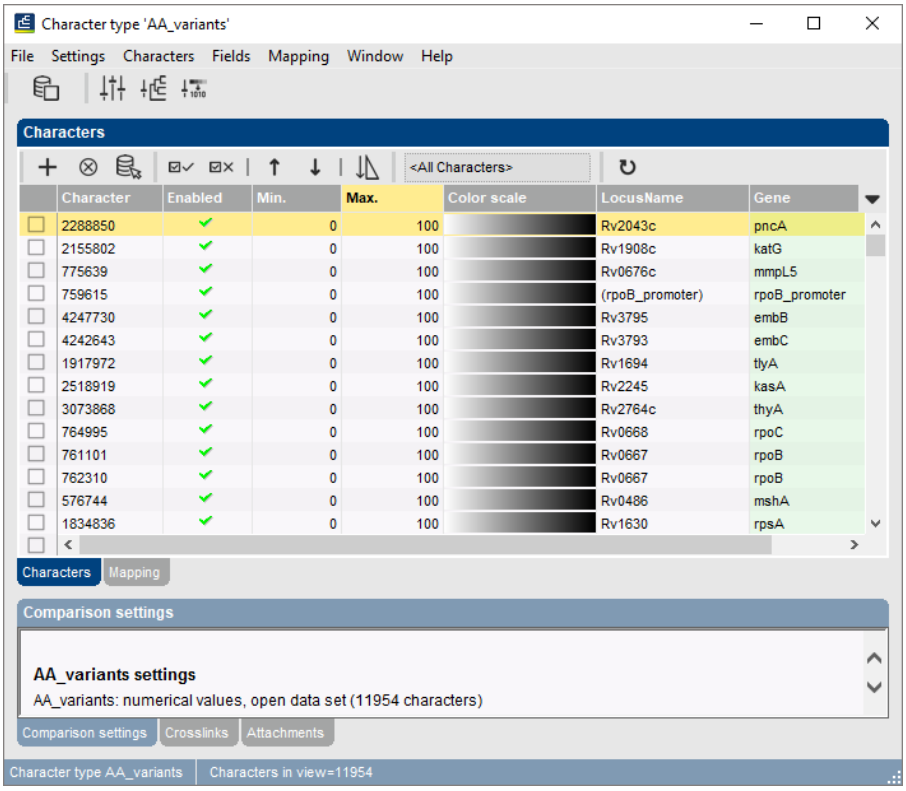

Figure 4.19: Amino acid variants experiment type. Character: genomic position (H37Rv numbering).

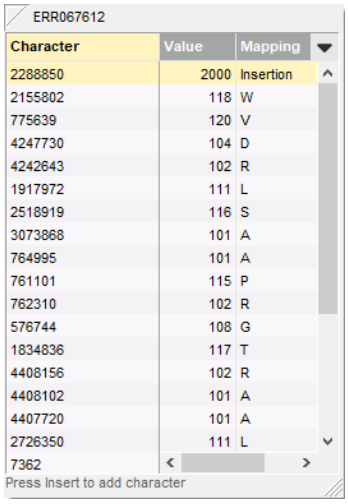

Figure 4.20: Example output of amino acid variants experiment type for sample ERR067612.

Chapter 5

MTBC genotyping reports

0.1 To generate a report, select the entries of interest and click on ***MTBC*** > ***Reports...***.

5.1 Summary section

Date: Date in MM/DD/YYYY on which the entry was last analysed with the plugin.

Name: The content for this entry in the display field, by default the key.

Info fields: The content of info fields in the database for this entry (determined by the user in the general settings of the plugin).

Predicted lineage: List of lineages for which lineage-related mutations (based on the SNP list of Coll ([1])) were found with sufficient coverage (determined by the user in the settings of the plugin).

Predicted resistance: All antibiotics for which at least 1 known resistance-related mutation or indel from the selected resistance KB was found with sufficient coverage.

Predicted susceptibility: All antibiotics for which all bases of the resistance-related genes were covered sufficiently and no known or unknown mutations (with sufficient coverage) were found.

Unknown: There are no known resistance-related mutations from the selected resistance KB found with sufficient coverage but there is at least 1 unknown mutation or indel found in resistance-related genes (only majority/consensus mutations compared to H37Rv are considered here).

5.2 Resistance section

5.2.1 Mutations

This section lists the mutations at positions for which a point mutation is known to cause resistance (according to the selected KB).

When you click on a header of a column, you sort all rows according to this column.

Antibiotic: Name of the antibiotic.

Locus: Name of the locus.

Codon position: In case of mutations in protein coding sequences, this number refers to the amino acid position in the protein.

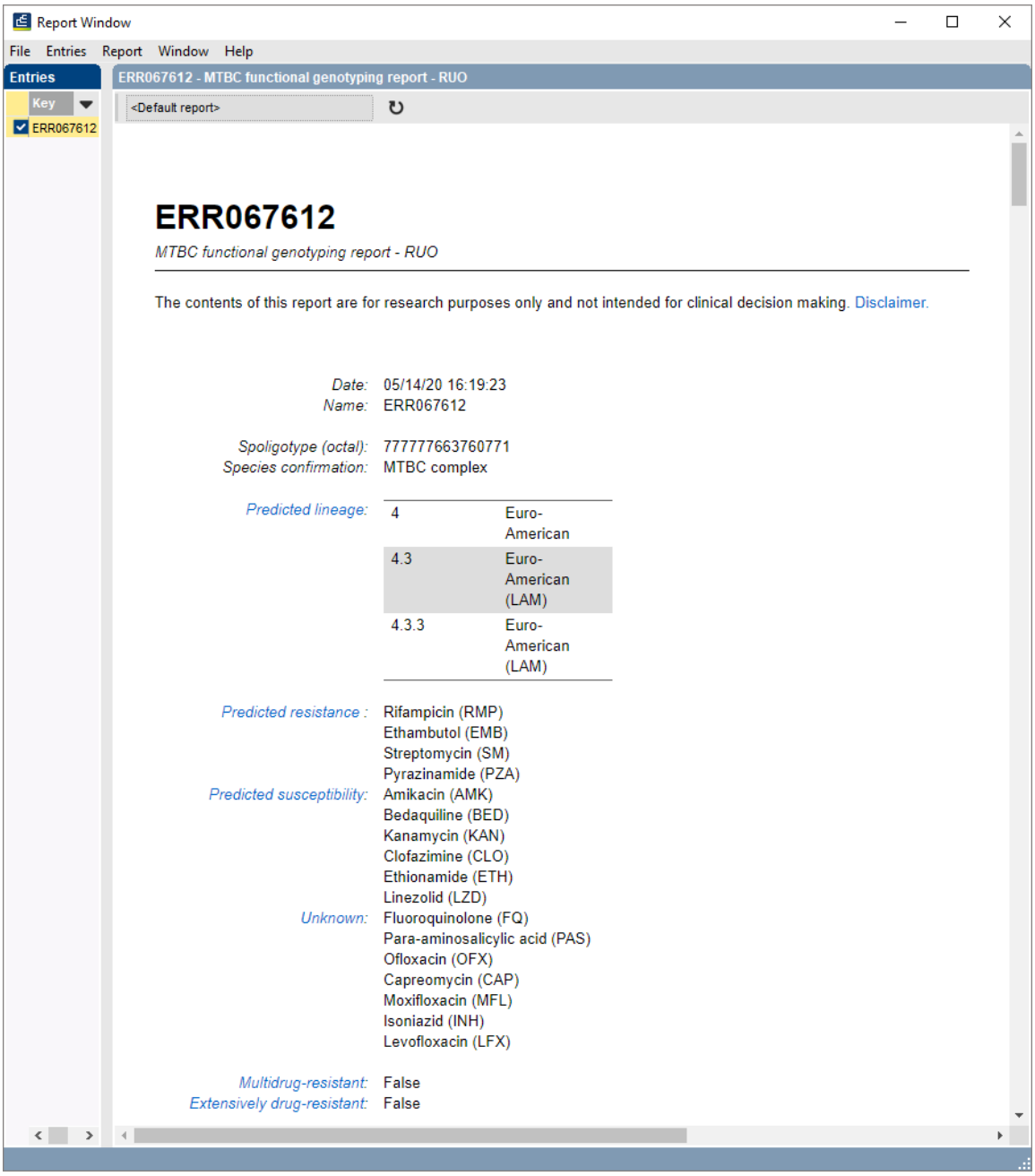

Figure 5.1: Summary page.

Resistance

Mutations

Antibiotic	Locus	Codon position	Genomic position	WT	Call	Conclusion	Coverage	Relative coverage	Notes
Ethambutol	embB	378		E	A	Unknown	248	97.25%	atypical
Fluoroquinolone	gyrB	538		Q	K	Unknown	34	12.19%	atypical
Isoniazid	ahpC_promoter		2726139	C	T	Resistant	248	99.60%	
Isoniazid	inhA	78		V	A	Resistant	179	98.90%	
Isoniazid	katG	315		S	T	Resistant	182	98.38%	
Streptomycin	rpsL	43		K	R	Resistant	281	99.29%	

Figure 5.2: Example output of MTBC report: resistance - mutations section for sample ERR551070.

Genomic position: In case of mutations in non-protein coding sequences (e.g. promotor sequences), this number refers to the nucleotide position in the genome of *M. tuberculosis* H37RV (NC_000962.3).

WT: Base or amino acid occurring at that specific position in the genome of *M. tuberculosis* H37Rv (NC_000962.3). If the gene is located at the negative strand, the base at the negative strand is displayed.

Call: Amino acid call in your sample in case of mutations in protein coding sequences and base calls in case of promotor regions or other noncoding DNA sequences (e.g. rrs). If the gene is located at the positive strand, the base at the positive strand is displayed. If the gene is located at the negative strand, the base at the negative strand is displayed.

Conclusion:

- ***Resistant***: When a specific mutation from the resistance KB was found.
- ***Unknown***: When there is an unknown mutation at a genomic position included in the selected resistance KB (the specific call is different than the one(s) included in the selected KB). e.g. ribD G8R is known to be involved in resistance to para-aminosalicylic acid. If your sample contains the mutation ribD G8R, the report will indicate resistance for PAS. However, if your sample contains the mutation ribD G8S, the conclusion will be ***Unknown***.

Coverage: Amount of reads containing the specific mutation. For an amino acid, the coverage is the minimum of the coverages of the three positions within the codon.

Relative coverage: Amount of reads containing the specific mutation at the genomic position under consideration divided by the total amount of reads mapped to that position. For an amino acid, the relative coverage is the minimum of the relative coverages of the three positions within the codon. Relative coverages that deviate a lot from 100% can be an indication of mixed infections or contaminations.

Notes:

- ***Atypical***: in case of unknown mutations.

5.2.2 Premature stop codons

This section lists the mutations at positions for which a premature stopcodon is known to cause resistance (according to the selected KB).

Premature stop codons

Antibiotic	Locus	Codon position	WT	Call	Conclusion	Coverage	Relative coverage	Notes
Streptomycin	gid	127	Q	Stop	Resistant	140	97.90%	

Figure 5.3: Example output of MTBC report: resistance - premature stop codons section for sample ERR553228.

Antibiotic: Name of the antibiotic.

Locus: Name of the locus.

Codon position: Amino acid position in the protein.

Wild type: Amino acid occurring at that specific position in the genome of *M. tuberculosis* H37Rv (NC_000962.3).

Call: Amino acid call in your sample. STOP = stopcodon.

Conclusion:

- ***Resistant***: In case a nonsense mutation (stopcodon; *) is present.
- ***Unknown***: In case a mutation is present at this position, but it is not a nonsense mutation.

Coverage: the coverage is the minimum of the coverages of the three positions within the codon.

Relative coverage: The relative coverage is the minimum of the relative coverages of the three positions within the codon.

5.2.3 Insertions/deletions

This section lists positions for which indels are known to cause resistance (according to the selected KB).

Insertions/deletions

Antibiotic	Locus	Codon position	Genomic position	WT	Call	Conclusion	Notes
Pyrazinamide	pncA	131	2288850		G	Resistant	

Figure 5.4: MTBC report: resistance - insertions/deletions section.

Antibiotic: Name of the antibiotic.

Locus: Name of the locus.

Codon position: This number refers to the amino acid position in the protein in case of protein coding sequences.

Genomic position: Start position of the region in the genome of *M. tuberculosis* H37Rv (NC_000962.3), for which insertions or deletions are related to resistance (defined in selected resistance KB).

WT: Base occurring at that specific position in the genome of *M. tuberculosis* H37Rv (NC_000962.3) in case of deletions. If the gene is located at the positive strand, the base at the positive strand is displayed. If the gene is located at the negative strand, the base at the negative strand is displayed. In case of insertions, nothing will be displayed.

Call: In case of a deletion, the call will be '-'. In case of insertions, the call will be the inserted sequence. If the gene is located at the positive strand, the base at the positive strand is displayed. If the gene is located at the negative strand, the base at the negative strand is displayed.

Conclusion:

- ***Resistant***: When a specific insertion/deletion included in the selected resistance KB was found.
- ***Unknown***: when there is an unknown insertion/deletion in a region known to be related to resistance (as defined in the selected resistance KB).
- ***Insertion with equal length in KB***: The sequence inserted at this position has the same length as a sequence in the resistance knowledge base (at the same position) but has a different DNA sequence.

- ***Larger deletion***: the deletion in your sample is larger than the known resistance-related deletion from the selected KB at this position.
- ***Deletion close, but not on this genomic position***: a deletion was found at a short distance (1 bp downstream or upstream) from a known resistance-related deletion (that is, included in the selected KB).

5.2.4 Quality metrics

This list contains quality metrics of putative and confirmed resistance-related loci (extracted from FIND/ReseqTB).

Quality metrics I

Antibiotic	Locus	Gene	QC Passed	Size	Number of bases covered	Average coverage	Start	Stop
INH	(ahpC_promoter)	ahpC_promoter	V	82	82	22.74	1	82
AMK, KAN	(eis_promoter)	eis_promoter	V	55	55	38.27	1	55
EMB	(embA_promoter)	embA_promoter	V	17	17	29.53	1	17
ETH, INH	(fabG1_promoter)	fabG1_promoter	V	101	101	45.33	1	101
	(inhA_promoter)	inhA_promoter	V	855	855	28.90	1	855
INH	(katG_promoter)	katG_promoter	V	13	13	24.15	1	13
PZA	(pncA_promoter)	pncA_promoter	V	13	13	27.85	1	13
LZD	(rrl)	rrl	V	3138	3138	38.59	1	3138
AMK, CAP, KAN, SM	(rrs)	rrs	V	1537	1537	45.34	1	1537
CAP	(tlyA_promoter)	tlyA_promoter	V	101	101	16.73	1	101
FQ, LFX, OFX	Rv0005	gyrB	V	2028	2028	36.57	1	2028
FQ, LFX, MFL, OFX	Rv0006	gyrA	V	2517	2517	30.70	1	2517
	Rv0129c	fbpC	V	1023	1023	38.30	1	1023
	Rv0340		V	540	540	17.51	1	540
	Rv0341	iniB	X	1440	1371	15.75	1	1440
EMB, INH	Rv0342	iniA	V	1923	1923	30.11	1	1923

Figure 5.5: Example output of MTBC report: resistance - Quality metrics I section for sample ERR551067.

Antibiotics: Abbreviation of the antibiotics for which resistance is linked to this gene.

Locus: Name of the locus.

Gene: Name of the gene.

Size: Length of the locus (expressed as amount of base pairs).

QC passed: When all positions within the locus are covered with sufficient coverage, a green ”V” sign will be present. Otherwise, a red ”X” will be present.

Number of bases covered: number of positions within the locus with sufficient coverage.

Average coverage: Average base coverage of all bases included in the locus.

Start: First base of the locus with sufficient coverage.

Stop: Last base of the locus with sufficient coverage.

Unresolved bases: Amount of ambiguous bases in the mapped consensus sequence (e.g. M, Y, N,. . .).

Start codon: States whether there is a start codon (either ATG, GTG or TTG) present or not.

Stop codon: States whether there is a stop codon present or not.

Quality metrics II

Antibiotic	Locus	QC Passed	Unresolved bases	Start Codon	Stop Codon	Internal stop codon
INH	(ahpC_promoter)	V	0			
AMK, KAN	(eis_promoter)	V	0			
EMB	(embA_promoter)	V	0			
ETH, INH	(fabG1_promoter)	V	0			
	(inhA_promoter)	V	0			
INH	(katG_promoter)	V	0			
PZA	(pncA_promoter)	V	0			
LZD	(rrl)	V	0			
AMK, CAP, KAN, SM	(rrs)	V	0			
CAP	(tlyA_promoter)	V	0			
FQ, LFX, OFX	Rv0005	V	0	Yes	Yes	No
FQ, LFX, MFL, OFX	Rv0006	V	0	Yes	Yes	No
	Rv0129c	V	0	Yes	Yes	No
	Rv0340	V	1	Yes	Yes	No
	Rv0341	X	37	Yes	Yes	No
EMB, INH	Rv0342	V	0	Yes	Yes	No

Figure 5.6: Example output of MTBC report: resistance - Quality metrics II section for sample ERR551067.

Internal stop codon: States whether there is a premature (internal) stop codon present inside the locus.

Quality metrics III

Antibiotic	Locus	QC Passed	NA mutations	AA mutations	Unknown mutations	Number of deletions	Unknown deletions	Number of insertions	Unknown insertions
INH	(ahpC_promoter)	V	0		0	0	0	0	0
AMK, KAN	(eis_promoter)	V	0		0	0	0	0	0
EMB	(embA_promoter)	V	0		0	0	0	0	0
ETH, INH	(fabG1_promoter)	V	0		0	0	0	0	0
	(inhA_promoter)	V	0		0	0	0	0	0
INH	(katG_promoter)	V	0		0	0	0	0	0
PZA	(pncA_promoter)	V	0		0	0	0	0	0
LZD	(rrl)	V	0		0	0	0	0	0
AMK, CAP, KAN, SM	(rrs)	V	0		0	0	0	0	0
CAP	(tlyA_promoter)	V	0		0	0	0	0	0
FQ, LFX, OFX	Rv0005	V	1	0	0	0	0	0	0
FQ, LFX, MFL, OFX	Rv0006	V	3	3	3	0	0	0	0
	Rv0129c	V	0	0	0	0	0	0	0
	Rv0340	V	0	0	0	0	0	0	0
	Rv0341	X	0	0	0	4	4	0	0
EMB, INH	Rv0342	V	0	0	0	0	0	0	0

Figure 5.7: Example output of MTBC report: resistance - Quality metrics III section for sample ERR551067.

NA mutations: Amount of nucleotide mutations in the mapped consensus sequence of your sample compared to the sequence of H37Rv (this includes both known and unknown mutations; ambiguous bases are not considered).

AA mutations: Amount of amino acid mutations compared to the translated sequence of H37RV (this includes both known and unknown mutations).

Unknown mutations: Amount of nucleotide or amino acid mutations (in case of protein coding sequences) that are not present in the selected resistance KB.

Number of deletions: Amount of deletions in the mapped consensus sequence of your sample compared to the sequence of H37Rv (this includes both known and unknown deletions). A deletion that spans multiple bases is only counted once.

Unknown deletions: Amount of deletions that are not present in the selected resistance KB.

Number of insertions: Amount of insertions in the mapped consensus sequence of your sample compared to the sequence of H37Rv (this includes both known and unknown insertions).

Unknown insertions: Amount of insertions that are not present in the selected resistance KB.

5.2.5 Genomic variants

This section lists the mutations and indels with sufficient coverage in putative and confirmed resistance-related loci (extracted from FIND/ReseqTB).

Genomic variants

Antibiotic	Locus	Gene	Size	Position	Genomic position	Type	WT	Call	Relative coverage	Strand
	Rv1592c		1341	963	1792778	Mutation	A (E)	G (E)	100.00%	neg
	Rv1592c		1341	964	1792777	Mutation	A (I)	G (V)	100.00%	neg
	Rv2247	accD6	1422	600	2521342	Mutation	T (D)	C (D)	100.00%	pos
INH	Rv2428	ahpC	588	254	2726446	Mutation	T (V)	G (G)	19.23%	pos
	Rv2846c	efpA	1593	271	3154361	Mutation	A (S)	C (R)	12.77%	neg
EMB	Rv3795	embB	3297	918	4247431	Mutation	G (M)	A (I)	100.00%	pos
EMB	Rv3795	embB	3297	1004	4247517	Mutation	A (N)	C (T)	16.67%	pos
EMB	Rv3793	embC	3285	2781	4242643	Mutation	C (R)	T (R)	96.30%	pos
EMB	Rv3793	embC	3285	2941	4242803	Mutation	G (V)	C (L)	100.00%	pos
FQ, LFX, MFL, OFX	Rv0006	gyrA	2517	61	7362	Mutation	G (E)	C (Q)	100.00%	pos
FQ, LFX, MFL, OFX	Rv0006	gyrA	2517	284	7585	Mutation	G (S)	C (T)	95.83%	pos
FQ, LFX, MFL, OFX	Rv0006	gyrA	2517	2003	9304	Mutation	G (G)	A (D)	100.00%	pos
FQ, LFX, OFX	Rv0005	gyrB	2028	1641	6880	Mutation	G (L)	A (L)	85.71%	pos
	Rv0341	iniB	1440	1268	410629	Mutation	A (E)	G (G)	33.33%	pos

Figure 5.8: Example output of MTBC report: resistance - genomic variants section for sample ERR551067.

Locus: Name of the locus.

Gene: Name of the gene.

Size: Length of the locus (expressed as amount of base pairs).

Position: This number refers to the nucleotide position in the promotor/gene, both in case of protein-coding and non-coding sequences.

Genomic position: Position in the genome of *M. tuberculosis* H37Rv (NC_000962.3).

Type: Mutation, insertion, deletion or missing in case there is no coverage.

WT: Base occurring at that specific position in the genome of *M. tuberculosis* H37Rv (NC_000962.3). If the gene is located at the negative strand, the base at the negative strand is displayed. In case of coding sequences, the amino acid call is displayed between brackets.

Call: Base call at that specific position in your sample. If the gene is located at the negative strand, the base at the negative strand is displayed. In case of coding sequences, the amino acid call is displayed between brackets. * = stopcodon.

Coverage: Relative coverage of the mutation.

Strand:

- Pos: Gene is located at the positive strand.
- Neg: Gene is located at the negative strand.

5.3 Lineage section

5.3.1 Mutations

Lineage

Mutations:

Lineage	Locus	Position	WT	Call	Coverage	Relative coverage	Notes
4 Euro-American	Rv0835	171	C	T	46	100.00%	
4.1 Euro-American	Rv0058	2262	G	A	23	100.00%	

Figure 5.9: Example output of MTBC report: lineage - mutations section for sample ERR551067.

Lineage: Lineage number and name as defined in 4.3.

Locus: Name of locus in which a lineage-related mutation was found.

Position: Nucleotide position in the gene.

Wild type: Base occurring at that specific position in the genome of *M. tuberculosis* H37Rv (NC_000962.3). If the gene is located at the negative strand, the base at the negative strand is displayed.

Call: Base call at that specific position in your sample. If the gene is located at the negative strand, the base at the negative strand is displayed.

Coverage: Amount of reads containing the specific mutation.

Relative coverage: Amount of reads containing the specific mutation at the genomic position under consideration divided by the total amount of reads mapped to that position. Relative coverages that deviate a lot from 100% can be an indication of a mixed infection or contamination.

Notes:

- ***Atypical***: there is a mutation at a known lineage related position, but the call itself is unknown (not included in the SNP list of Coll and coworkers ([1])).

5.3.2 Quality metrics

The terminology is the same as for the quality metrics section of resistance (see 5.2.4)

5.4 Export functions

The report can be exported in several formats:

When you click on ***Report*** $>$ ***Print***, you can generate a pdf file of the report for the highlighted entry.

If you click on ***Report*** $>$ ***Export current report***, separate tsv files for the highlighted entry will be generated in the export folder of the database.

If you want to export the results of multiple entries at once, select the entries in the *Report window* and click on ***Entries*** $>$ ***Export selected***.

Chapter 6

Tips and tricks

6.1 Making character views

If you are interested in only 1 particular antibiotic, you can restrict the view of the mutations to only those related to this antibiotic:

1.1 Double-click on the Resistance Mutation experiment type in the *Experiment types* panel of the *Main* window.

1.2 Press **Ctrl+Shift+F** or select ***Characters*** > ***Find character...*** ( , **Ctrl+Shift+F**).

1.3 Enter e.g. “streptomycin” and click on <***Select all***>.

1.4 Click in the menubar on ***Characters*** > ***Character views*** > ***Manage user defined views*** or click on the toolbar button <***All characters***> and choose <***Manage user defined views***>. Click <***Add***> and enter a name (e.g. “Streptomycin”). The view of the characters will be automatically switched to the selection. If you want to see all characters again, you can click on ***Characters*** > ***Character views*** > ***All characters*** (see Figure 6.1).

1.5 If you open a new comparison with the selected entries, you can now select ***Streptomycin*** from the aspect list next to the name of the experiment type (see Figure 6.2).

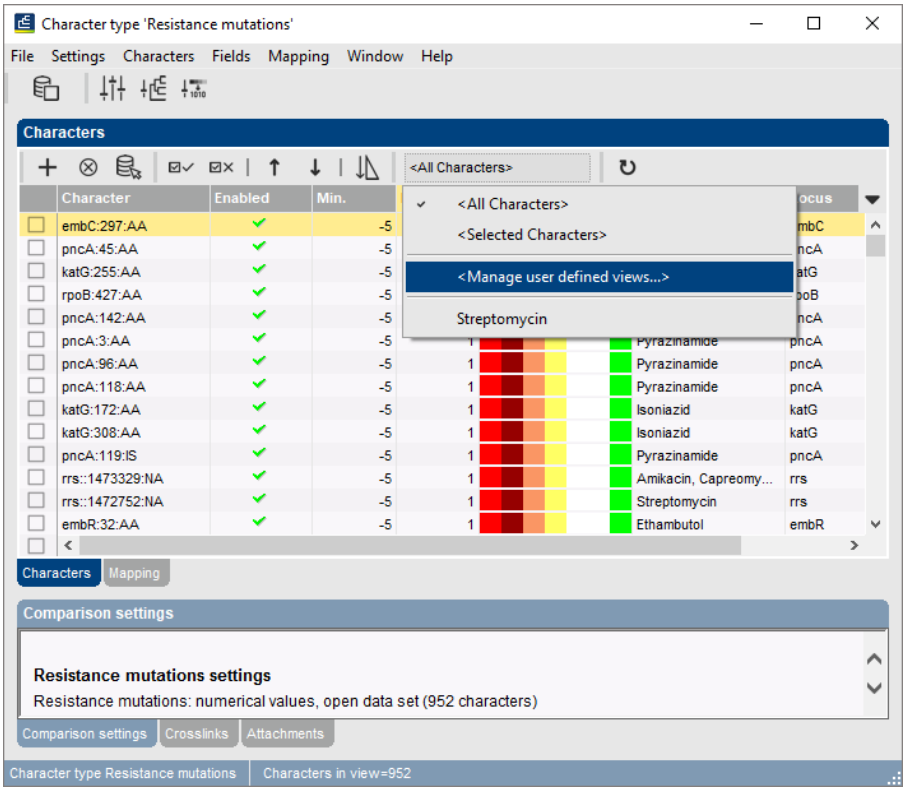

Figure 6.1: Manage user defined views in a character experiment type.

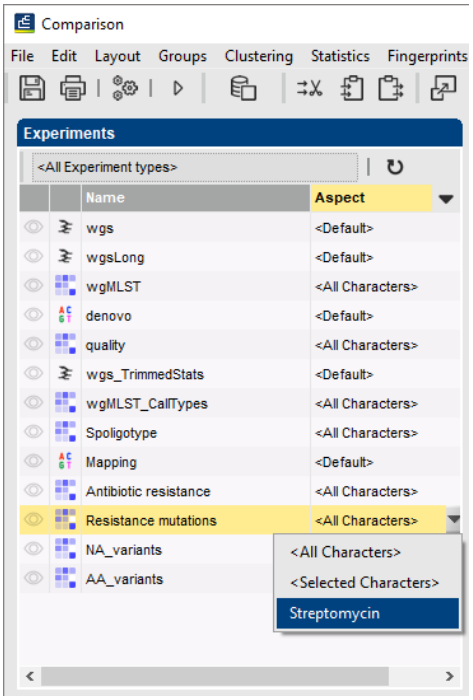

Figure 6.2: Restrict view to selection of mutations (e.g. only those related to Streptomycin resistance)

Chapter 7

Supplementary materials

7.1 List of 166 *Mycobacterium* species used as reference strains for species prediction

- *Mycobacterium conspicuum*
- *Mycobacterium hodleri* DSM44183
- *Mycobacterium malmoense* ATCC29571
- *Mycobacterium duvalii* ATCC43910
- *Mycobacterium moriokaense* DSM44221T
- *Mycobacterium tuberculosis* NCTC7416H37Rv
- *Mycobacterium frederiksbergense* DSM44346
- *Mycobacterium agri* DSM44515T
- *Mycobacterium austroafricanum* ATCC33464
- *Mycobacterium elephantis* AJ010747
- *Mycobacterium pulveris* DSM44222T
- *Mycobacterium heidelbergense* 2554/91
- *Mycobacterium intermedium* X67847
- *Mycobacterium avium* ATCC19698
- *Mycobacterium flavescens* ATCC14474
- *Mycobacterium gilvum* ATCC43909
- *Mycobacterium nonchromogenicum* ATCC19530
- *Mycobacterium chlorophenolicum* X79292
- *Mycobacterium szulgai* ATCC25799
- *Mycobacterium simiae* ATCC25275
- *Mycobacterium gordonae* ATCC14470

- *Mycobacterium shimoidei* ATCC27962
- *Mycobacterium genavense* X60070
- *Mycobacterium capraeg* M-1
- *Mycobacterium rhodesiae* DSM44223T
- *Mycobacterium parafortuitum* DSM43528
- *Mycobacterium botniense* E347
- *Mycobacterium goodie* M069
- *Mycobacterium marinum* ATCC927
- *Mycobacterium holsaticum* 1406
- *Mycobacterium smegmatis* ATCC19420
- *Mycobacterium komossense* ATCC33013
- *Mycobacterium chitae* ATCC19627
- *Mycobacterium thermoresistibile* ATCC19527
- *Mycobacterium aurum* ATCC23366
- *Mycobacterium gadium* ATCC27726
- *Mycobacterium aichiense* ATCC27280
- *Mycobacterium leprae*
- *Mycobacterium hiberniae* ATCC9874
- *Mycobacterium obuense* ATCC27023
- *Mycobacterium cookii* ATCC49103(T)=NZ2
- *Mycobacterium xenopi* typestrain:DSM43995
- *Mycobacterium kansasii* typestrain:DSM44162
- *Mycobacterium intracellulare* typestrain:DSM43223
- *Mycobacterium avium* typestrain:DSM44156
- *Mycobacterium vanbaalenii* typestrain:DSM7251=PYR-1
- *Mycobacterium confluentis* typestrain:DSM44017
- *Mycobacterium mageritense* typestrain:DSM44476
- *Mycobacterium saskatchewanense* NRCM00-250
- *Mycobacterium parmense* MUP1182
- *Mycobacterium immunogenum* DSM44764
- *Mycobacterium kubicae* CDC941078
- *Mycobacterium heckeshornense* S369

- *Mycobacterium doricum* FI-13295
- *Mycobacterium montefiorense* ATCCBAA-256
- *Mycobacterium pinnipedii* AF502574
- *Mycobacterium sherrisii* ATCCBAA-832
- *Mycobacterium celatum* L08169
- *Mycobacterium psychrotolerans* typestrain:WA101
- *Mycobacterium parascrofulaceum* ATCCBAA-614
- *Mycobacterium alvei* CIP103464
- *Mycobacterium tusciae* FI-25796
- *Mycobacterium branderi* ATCC51789
- *Mycobacterium brumae* ATCC51384
- *Mycobacterium engbaekii* ATCC27353
- *Mycobacterium lentiflavum* ATCC51985
- *Mycobacterium microti* ATCC19422
- *Mycobacterium poriferae* ATCC35087
- *Mycobacterium tokaiense* ATCC27282
- *Mycobacterium vaccae* ATCC15483
- *Mycobacterium neoaurum* ATCC25795
- *Mycobacterium asiaticum* ATCC25276
- *Mycobacterium chubuense* ATCC27278
- *Mycobacterium diernhoferi* ATCC19340
- *Mycobacterium fallax* ATCC35219
- *Mycobacterium gastri* ATCC15754
- *Mycobacterium phlei* ATCC11758
- *Mycobacterium scrofulaceum* ATCC19981
- *Mycobacterium africanum* ATCC25420
- *Mycobacterium palustre* E846
- *Mycobacterium terrae* M29568
- *Mycobacterium hassiacum* U49401
- *Mycobacterium triplex* 90-1019
- *Mycobacterium bohemicum* U84502
- *Mycobacterium novocastrense* 73

- *Mycobacterium pyrenivorans* DSM44605
- *Mycobacterium fortuitum* CIP104534
- *Mycobacterium houstonense* ATCC49403
- *Mycobacterium neworleansense* ATCC49404
- *Mycobacterium peregrinum* CIP105382
- *Mycobacterium septicum* DSM44393
- *Mycobacterium chelonae* CIP104535
- *Mycobacterium mucogenicum* ATCC49650
- *Mycobacterium porcinum* CIP105392
- *Mycobacterium senegalense* CIP104941
- *Mycobacterium wolinskyi* ATCC700010
- *Mycobacterium farcinogenes* NCTC10955
- *Mycobacterium boenickei* W5998
- *Mycobacterium brisbanense* W6743
- *Mycobacterium canariasense* 502329
- *Mycobacterium nebraskense* UNMC-MY1349
- *Mycobacterium abscessus* CCUG48898
- *Mycobacterium cosmeticum* LTA-388
- *Mycobacterium lacus* NRCM00-255
- *Mycobacterium rufum* JS14
- *Mycobacterium aromaticivorans* JS19b1
- *Mycobacterium fluoranthenivorans* typestrain:FA-4
- *Mycobacterium triviale* TMC1453
- *Mycobacterium pseudoshottsii* L15
- *Mycobacterium chimaera* typestrain:FI-0169
- *Mycobacterium florentinum* typestrain:F-I93171
- *Mycobacterium shottsii* M175
- *Mycobacterium arupense* AR30097
- *Mycobacterium colombiense* typestrain:10B
- *Mycobacterium abscessus* CIP108541
- *Mycobacterium phocaicum* CIP108542
- *Mycobacterium aubagnense* CIP108543

- *Mycobacterium conceptionense* CIP108544
- *Mycobacterium pallens* czh-8
- *Mycobacterium rutilum* czh-117
- *Mycobacterium monacense* B9-21-178
- *Mycobacterium crocinum* czh-42
- *Mycobacterium kumamotonense* CST7274(=GTC2729)
- *Mycobacterium salmoniphilum* ATCC13758
- *Mycobacterium stomatepiae* typestrain:DSM45059
- *Mycobacterium llatzerense* typestrain:MG13
- *Mycobacterium noviomagense* NLA000500338
- *Mycobacterium riyadhense* NLA000201958
- *Mycobacterium setense* ABO-M06
- *Mycobacterium bouchedurhonense* 4355387
- *Mycobacterium senuense* 05-832
- *Mycobacterium arosiense* T1921
- *Mycobacterium avium* ATCC49884
- *Mycobacterium seoulense* 03-19
- *Mycobacterium vulneris* NLA000700772
- *Mycobacterium mantenii* NLA000401474
- *Mycobacterium marseillense* 5356591
- *Mycobacterium paraseoulense* 31118
- *Mycobacterium hippocampi* BFLP-6
- *Mycobacterium paraffinicum* ATCC12670
- *Mycobacterium madagascariense* JCM13574
- *Mycobacterium murale* JCM13392
- *Mycobacterium algericum* Bejaia
- *Mycobacterium litorale* F4
- *Mycobacterium europaeum* DSM45397
- *Mycobacterium interjectum* HM037998
- *Mycobacterium iranicum* M05
- *Mycobacterium ulcerans* ATCC19423
- *Mycobacterium sphagni* typestrain:DSM44076

- *Mycobacterium fortuitum* typestrain:DSM44220
- *Mycobacterium heraklionense* NCTC13432
- *Mycobacterium parakoreense* 299
- *Mycobacterium koreense* 01-305
- *Mycobacterium shinjukuense* GTC2738
- *Mycobacterium longobardum* DSM45394
- *Mycobacterium immunogenum* typestrain:DSM45595
- *Mycobacterium yongonense* 05-1390
- *Mycobacterium haemophilum* DSM44634
- *Mycobacterium bourgelatii* 02-5273-A84
- *Mycobacterium fragae* HF8705
- *Mycobacterium timonense* CIP109830
- *Mycobacterium sediminis* YIMM13028
- *Mycobacterium kyorinense* KUM060204
- *Mycobacterium paragordonae* 49061
- *Mycobacterium celeriflavum* AFPC-000207
- *Mycobacterium paraense* IEC26

7.2 List of 43 spacer oligonucleotides used as reference sequences for *in silico* spoligotyping

>SpoligoSpacer_1
ATAGAGGGTCGCCGGCTCTGGATCA

>SpoligoSpacer_2
CCTCATGCTTGGGCGACAGCTTTTG

>SpoligoSpacer_3
CCGTGCTTCCAGTGATCGCCTTCTA

>SpoligoSpacer_4
ACGTCATACGCCGACCAATCATCAG

>SpoligoSpacer_5
TTTTCTGACCACTTGTGCGGGATTA

>SpoligoSpacer_6
CGTCGTCATTTCCGGCTTCAATTTC

>SpoligoSpacer_7

GAGGAGAGCGAGTACTCGGGGCTGC

>SpoligoSpacer_8
CGTGAAACCGCCCCCAGCCTCGCCG

>SpoligoSpacer_9
ACTCGGAATCCCATGTGCTGACAGC

>SpoligoSpacer_10
TCGACACCCGCTCTAGTTGACTTCC

>SpoligoSpacer_11
GTGAGCAACGGCGGCGGCAACCTGG

>SpoligoSpacer_12
ATATCTGCTGCCCGCCCGGGGAGAT

>SpoligoSpacer_13
GACCATCATTGCCATTCCCTCTCCC

>SpoligoSpacer_14
GGTGTGATGCGGATGGTCGGCTCGG

>SpoligoSpacer_15
CTTGAATAACGCGCAGTGAATTTCG

>SpoligoSpacer_16
CGAGTTCCCGTCAGCGTCGTAAATC

>SpoligoSpacer_17
GCGCCGGCCCGCGCGGATGACTCCG

>SpoligoSpacer_18
CATGGACCCGGGCGAGCTGCAGATG

>SpoligoSpacer_19
TAACTGGCTTGGCGCTGATCCTGGT

>SpoligoSpacer_20
TTGACCTCGCCAGGAGAGAAGATCA

>SpoligoSpacer_21
TCGATGTCGATGTCCCAATCGTCGA

>SpoligoSpacer_22
ACCGCAGACGGCACGATTGAGACAA

>SpoligoSpacer_23
AGCATCGCTGATGCGGTCCAGCTCG

>SpoligoSpacer_24
CCGCCTGCTGGGTGAGACGTGCTCG

>SpoligoSpacer_25
GATCAGCGACCACCGCACCCTGTCA

>SpoligoSpacer_26
CTTCAGCACCACCATCATCCGGCGC

>SpoligoSpacer_27
GGATTCGTGATCTCTTCCCGCGGAT

>SpoligoSpacer_28
TGCCCCGGCGTTTAGCGATCACAAC

>SpoligoSpacer_29
AAATACAGGCTCCACGACACGACCA

>SpoligoSpacer_30
GGTTGCCCCGCGCCCTTTTCCAGCC

>SpoligoSpacer_31
TCAGACAGGTTCGCGTCGATCAAGT

>SpoligoSpacer_32
GACCAAATAGGTATCGGCGTGTTCA

>SpoligoSpacer_33
GACATGACGGCGGTGCCGCACTTGA

>SpoligoSpacer_34
AAGTCACCTCGCCCACACCGTCGAA

>SpoligoSpacer_35
TCCGTACGCTCGAAACGCTTCCAAC

>SpoligoSpacer_36
CGAAATCCAGCACCACATCCGCAGC

>SpoligoSpacer_37
CGCGAACTCGTCCACAGTCCCCCTT

>SpoligoSpacer_38
CGTGGATGGCGGATGCGTTGTGCGC

>SpoligoSpacer_39
GACGATGGCCAGTAAATCGGCGTGG

>SpoligoSpacer_40
CGCCATCTGTGCCTCATACAGGTCC

>SpoligoSpacer_41
GGAGCTTTCCGGCTTCTATCAGGTA

>SpoligoSpacer_42
ATGGTGGGACATGGACGAGCGCGAC

>SpoligoSpacer_43
CGCAGAATCGCACCGGGTGCGGGAG

Chapter 8

Abbreviations

- AMK: amikacin
- BED: bedaquiline
- CAP: capreomycin
- CLO: clofazimine
- EMB: ethambutol
- ETH: ethionamide
- FQ: fluoroquinolone
- INH: isoniazid
- KAN: kanamycin
- KB: resistance knowledge base, a list with all genomic variants (mutations, insertions and deletions) that are related to phenotypic resistance
- LFX: levofloxacin, belongs to the class of FQ
- LZD: linezolid
- MFL: moxifloxacin, belongs to the class of FQ
- OFX : ofloxacin, belongs to the class of FQ
- PAS: para-aminosalicylic acid
- PZA: pyrazinamide
- RMP: rifampicin
- SM: streptomycin
- WGS: whole genome sequencing

Bibliography

[1] Francesc Coll, Ruth McNerney, José Afonso Guerra-Assuncao, Judith R Glynn, Joao Perdigao, Miguel Viveiros, Isabel Portugal, Arnab Pain, Nigel Martin, and Taane G Clark. A robust snp barcode for typing mycobacterium tuberculosis complex strains. *Nature communications*, 5:4812, 2014.

bioMérieux S.A. • 69280 Marcy l'Étoile • France • Tel.: + 33 (0)4 78 87 20 00 • Fax: +33 (0)4 78 87 20 90
www.biomerieux.com