User Manual

MutaNET

Version 2.0

MutaNET comes with a next generation sequencing (NGS) pipeline that calls mutations based on paired–end NGS reads, an automated mutation analysis tool and various file converters and mergers. The mutation analysis feature considers the coding region, protein domains, regulation and transcription factor binding site information, and can be used to analyse the potential impact of mutations on genes of interest.

This manual gives further information on the features, input files, results, and settings. Additionally, it provides example work flows. The software and instructions were tested on Ubuntu 16.10, Mac OS El Capitan, Windows 8 and Windows 10.

Contents

1 Quickstart

1.1 Starting MutaNET

MutaNET comes as **Python 3 source code** as well as an **executable** for Windows.

Executable (Windows) To start MutaNET, double–click on the executable **MutaNET32.exe** or **MutaNET64.exe**, depending on whether you have a 32–bit or 64–bit Windows installation. If you are not sure, choose the 32–bit executable. Make sure that the executable remains in the same directory as the config.yaml file. Otherwise the user interface will not start.

From Source Open a command prompt or terminal and execute the following command:

```
python3 source_folder_path/mutaNET.py
```

or on **Windows** depending on your Python installation:

```
python source_folder_path/mutaNET.py
```

Source_folder_path is the path to the folder containing the source code of MutaNET. This requires Python 3 to be installed.

The installation manual explains how to install Python 3, as well as programs required for the NGS pipeline of MutaNET on **Windows**, **Linux** and **Mac OS X**.

1.2 Example Data Sets

When starting MutaNET for the first time, the file paths for small example data sets for the NGS pipeline, mutation analysis and file converters are already loaded to allow quick testing. Keep in mind that for the NGS pipeline extra programs need to be installed.

The settings for the example data can be restored any time by clicking on **Settings → Restore default settings**. The data sets can be found in the **example_data** folder, if you want to have a look at the file formats.

2 NGS Pipeline

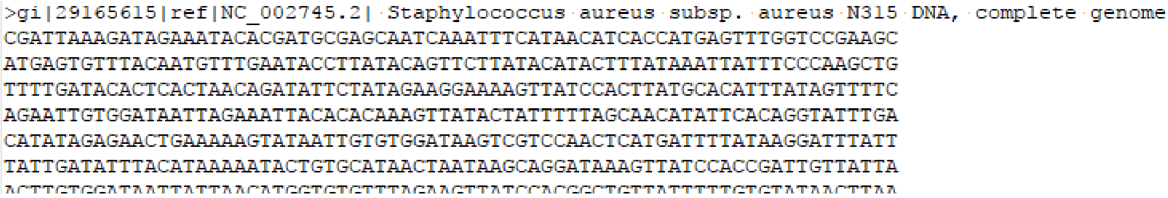
The next generation sequencing (NGS) pipeline aligns a set of paired–end NGS reads to a reference genome, performs quality control and mutation calling using Burrows-Wheeler-Aligner (BWA) [1], SAMTools [2] and VarScan [3].

2.1 Input Files

2.1.1 Reference Genome File

The reference genome file is in .fasta format and consists of a header line followed by the entire genome sequence. The header begins with **'>'** and then gives information about the genome.

Example:



```
>gi|29165615|ref|NC_002745.2| Staphylococcus aureus subsp. aureus N315 DNA, complete genome
CGATTAAAGATAGAAATACACGATGCGAGCAATCAAATTTCATAACATCACCATGAGTTTGGTCCGAAGC
ATGAGTGTTTACAATGTTTGAATACCTTATACAGTTCTTATACATACTTTATAAATTATTTCCCAAGCTG
TTTTGATACACTCACTAACAGATATTCTATAGAAGGAAAAGTTATCCACTTATGCACATTTATAGTTTTC
AGAATTGTGGATAATTAGAAATTACACACAAAGTTATACTATTTTTAGCAACATATTCACAGGTATTTGA
CATATAGAGAACTGAAAAAGTATAATTGTGTGGATAAGTCGTCCAACTCATGATTTTATAAGGATTTATT
TATTGATATTTACATAAAAATACTGTGCATAACTAATAAGCAGGATAAAGTTATCCACCGATTGTTATTA
ACTTGTGGATAATTATTAACATGGTGTGTTTAGAAGTTATCCACGGCTGTTATTTTTGTGTATAACTTAA
```

2.1.2 NGS Reads Directory

The directory contains .fastq files with paired–end NGS reads. The .fastq files must be paired, meaning that for each set of reads there needs to be a .fastq file for the plus strand and for the minus strand. The file names consist of three parts:

name_number.suffix

For each **name**, there needs to be two files with two different **numbers**. The **suffix** needs to contain **'fastq'**. The files can be compressed, i.e. **.fastq.gz**, as long as all files are compressed and have the same suffix.

Example File Names:

```
name1_1.fastq    or    name1_1.fastq.gz
name1_2.fastq          name1_2.fastq.gz
name2_A.fastq          name2_1.fastq.gz
name2_B.fastq          name2_2.fastq.gz
```

2.2 Results

The NGS pipeline produces a SNP and an indel variant calling format (.vcf) file for each pair of input read files, as well as a tab–separated (.tsv) file containing the mutations of all .vcf files. See Section 4.1 for an example .vcf file and Section 3.1.2 for an example mutation .tsv file.

If intermediate results are not deleted via the advanced settings, the results also include **.sai**, **.sam**, **.bam** and **.mpileup** files for each pair of input read files.

2.3 Advanced Settings

The advanced settings can be opened via **Settings → Advanced NGS settings** in the menu.

BWA executable:

Path to the Burrows-Wheeler-Aligner executable, which is required for aligning the reads to the reference genome. If you have not installed BWA yet, see the installation guide via **Help → Open installation guide** for instructions for Windows, Mac OS X and Linux.

SAMTools executable:

Path to the SAMTools executable, which is required for indexing and quality control. If you have not installed SAMTools yet, see the installation guide via **Help → Open installation guide** for instructions for Windows, Mac OS X and Linux.

VarScan executable:

Path to the VarScan executable, which is required for variant calling. If you have not installed VarScan yet, see the installation guide via **Help → Open installation guide** for instructions for Windows, Mac OS X and Linux.

SAMTools mapping quality:

Reads with mapping quality below this threshold are ignored.

Default: 30

VarScan SNP calling p-value:

P–value threshold for calling SNPs and indels.

Default: 0.05

Clean intermediate results:

If enabled, deletes the intermediate results of the pipeline to save disc space. The .vcf files with the called SNPs and indels are not deleted, and neither is the .tsv file with the merged mutation information.

Open result directory:

If this setting is enabled, the NGS pipeline result directory is opened in the operating system's file explorer after completing all analysis steps.

Save:

After making sure the new settings are valid, the advanced settings are saved to the user configuration file and the settings window is closed.

Cancel:

All changes to the advanced settings are discarded and the settings window is closed.

2.4 Runtime Examples

Computer Windows 10 laptop, 64–bit, Intel i7-7700HQ 2.8Ghz CPU (only 1 core used)

Step	**121.4kb .fastq**	**76.686kb .fastq**
input pre–processing	0.0s	0.0s
BWA (mapping)	45.0s	43.5s
BWA (to .sam)	13.4ss	9.7s
SAMTools (to .bam + indexing)	30.9s	13.8s
SAMTols (quality control)	10.9s	2.4s
SAMTools (generating .mpileup)	71.6s	21.4s
VarScan (SNP + indel calling)	94.9s	38.6s
writing output	0.3s	0.0s
total	**267.0s**	**129.5s**

3 Automated Mutation Analysis

The mutation analysis takes a set of genes and a set of mutations, maps the mutations onto different gene regions, computes various statistics and can be used to perform optional analysis steps to obtain further results.

Genes of Interest Analysis:
Given a file with names and/or locus tags of genes of interest, analyses the potential impact of mutations on this set of genes. In particular, it is possible to specify up to 10 sub-categories for these special interest genes in the genes of interest input file.

Coding Region Analysis:
Analyses the potential impact of mutations in coding regions by computing a score based on a given amino acid substitution matrix. Additionally, if **genes of interest analysis** is enabled, the scores of mutations in these special interest genes and in the remaining genes are statistically analysed.

Protein Domain Analysis:
Maps mutations to a given set of protein domains. This allows a better estimation of the potential impact of these mutations on protein function.

Additionally, some protein domain files, e.g. from UniProt, already contain known mutations and their effects as protein domains. If that is the case, a list of all such mutations that also occur in the input mutation set will be among the result files.

Regulation Analysis:
Builds a gene regulatory network (GRN) of the organism based on given regulation information. This allows a better estimation of the potential global impact of mutations.

If **genes of interest analysis** is enabled, the sub–network of these special interest genes and their direct and indirect regulators is constructed. Furthermore, a table with all non–synonymous mutations occurring in this sub–network will be among the result files.

Transcription Factor Binding Site Analysis:
Analyses the potential impact of mutations in transcription factor binding sites given a set of transcription factor binding site (TFBS) information and a set of aligned transcription factor (TF) sequences.

3.1 Input Files

The tab–separated (.tsv) files contain a header with column names in the first line. The order of these columns does not matter, nor are additional columns problematic.

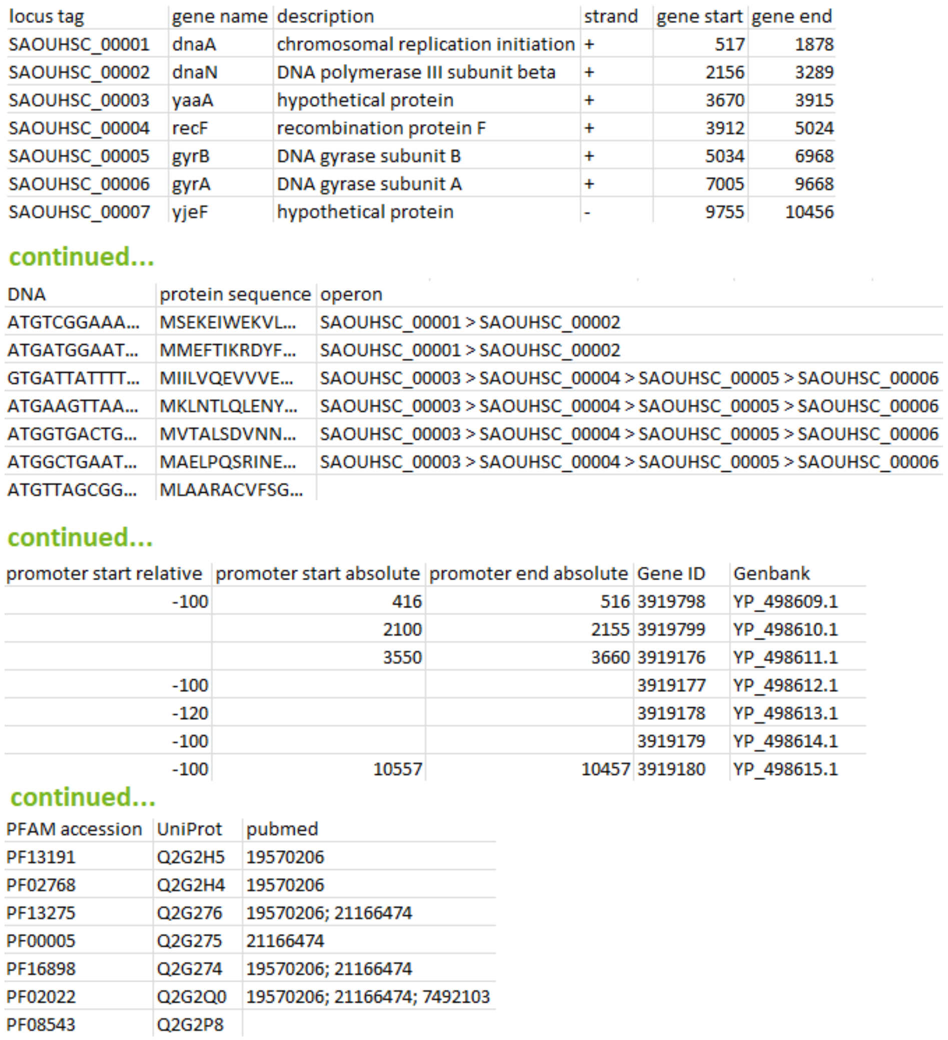
The genome positions in the input files are 1–based, meaning that the first base in the genome has position 1. Furthermore, all positions are given on the plus strand and are automatically adjusted for the respective strand by the analysis. Unknown fields can be left empty or marked with symbols such as `/`, `-`, `?`, `*`.

3.1.1 Gene File

The gene file is tab–separated and is always required for the mutation analysis. Each line, apart from the header, represents a gene. The table below describes the different columns and specifies if they are required to be in the file or not.

Column Name	**Description**	**Required**
chromosome	Name of the chromosome on which the gene is located.	no
locus tag	Unique locus tag of the gene. This field cannot be empty.	yes
gene name	General name of the gene.	no
description	Function of the gene or corresponding protein.	no
strand	The DNA strand the gene is on.	yes
gene start	Starting position of the coding region on the plus strand.	yes
gene end	Position of the last base in the coding region on the plus strand, which means that this position is included in the coding region.	yes
DNA	DNA sequence of the coding region on the respective strand. Allowed letters are **A**, **C**, **G**, **T**, **N**.	yes
protein sequence	Amino acid sequence of the protein. This should correspond to the DNA sequence.	yes
operon	Locus tags of the genes in the operon that includes this gene. They are separated by either **' > '** or **' < '**, indicating the order of the genes in the operon. Operons that contain both directions are ignored.	no
promoter start relative	Promoter region start relative to the coding region start. If the absolute start and end positions are not given, this is used to set the promoter region ranging from the relative start to the coding region start. An example can be found in Section 3.3.	no
promoter start absolute	Start position of the promoter region in the genome on the plus strand.	no
promoter end absolute	Position of the last base in the promoter region in the genome on the plus strand.	no
Gene ID	Accession number of the gene in NCBI Gene.	no
Genbank	Accession number of the gene in NCBI Genbank.	no
PFAM	Accession number of the protein in PFAM.	no
UniProt	Accession number of the protein in UniProt.	no
Pubmed	NCBI Pubmed IDs associated with the gene or corresponding protein. They are separated by **'; '**.	no

Example:



locus tag	gene name	description	strand	gene start	gene end
SAOUHSC_00001	dnaA	chromosomal replication initiation	+	517	1878
SAOUHSC_00002	dnaN	DNA polymerase III subunit beta	+	2156	3289
SAOUHSC_00003	yaaA	hypothetical protein	+	3670	3915
SAOUHSC_00004	recF	recombination protein F	+	3912	5024
SAOUHSC_00005	gyrB	DNA gyrase subunit B	+	5034	6968
SAOUHSC_00006	gyrA	DNA gyrase subunit A	+	7005	9668
SAOUHSC_00007	yjeF	hypothetical protein	-	9755	10456

continued...

DNA	protein sequence	operon
ATGTCGGAAA...	MSEKEIWEKVL...	SAOUHSC_00001 > SAOUHSC_00002
ATGATGGAAT...	MMEFTIKRDYF...	SAOUHSC_00001 > SAOUHSC_00002
GTGATTATTTT...	MIILVQEVVVE...	SAOUHSC_00003 > SAOUHSC_00004 > SAOUHSC_00005 > SAOUHSC_00006
ATGAAGTTAA...	MKLNTLQLENY...	SAOUHSC_00003 > SAOUHSC_00004 > SAOUHSC_00005 > SAOUHSC_00006
ATGGTGACTG...	MVTALSDVNN...	SAOUHSC_00003 > SAOUHSC_00004 > SAOUHSC_00005 > SAOUHSC_00006
ATGGCTGAAT...	MAELPQSRINE...	SAOUHSC_00003 > SAOUHSC_00004 > SAOUHSC_00005 > SAOUHSC_00006
ATGTTAGCGG...	MLAARACVFSG...	

continued...

promoter start relative	promoter start absolute	promoter end absolute	Gene ID	Genbank
-100	416	516	3919798	YP_498609.1
	2100	2155	3919799	YP_498610.1
	3550	3660	3919176	YP_498611.1
-100			3919177	YP_498612.1
-120			3919178	YP_498613.1
-100			3919179	YP_498614.1
-100	10557	10457	3919180	YP_498615.1

continued...

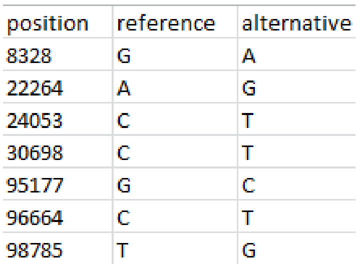
PFAM accession	UniProt	pubmed
PF13191	Q2G2H5	19570206
PF02768	Q2G2H4	19570206
PF13275	Q2G276	19570206; 21166474
PF00005	Q2G275	21166474
PF16898	Q2G274	19570206; 21166474
PF02022	Q2G2Q0	19570206; 21166474; 7492103
PF08543	Q2G2P8	

3.1.2 Mutation File

The mutation file is tab–separated and is always required for the mutation analysis. Each line, apart from the header, represents a mutation.

Column Name	Description
position	The genome position of the SNP, insertion or deletion on the plus strand. The positions are 1–based, meaning that the first base in the genome has position 1.
reference	The DNA base(s) on the plus strand in the reference genome.
alternative	The DNA base(s) on the plus strand in the mutated genome.

Example:



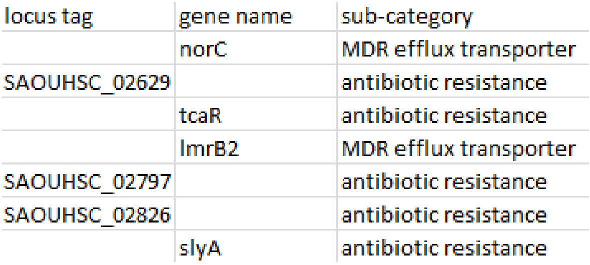
position	reference	alternative
8328	G	A
22264	A	G
24053	C	T
30698	C	T
95177	G	C
96664	C	T
98785	T	G

3.1.3 Genes of Interest File

The genes of interest file is tab–separated and is required for analysis specific to this set of genes, e.g. a set of antibiotic resistance genes. Each line, apart from the header, gives the locus tag or gene name of an a special interest gene.

Column Name	Description
locus tag	Unique locus tag of the gene.
gene name	General name of the gene.
sub–category	This column allows to specify further sub–categories for the genes of interest. There can be up to 10 sub–categories.

Example:

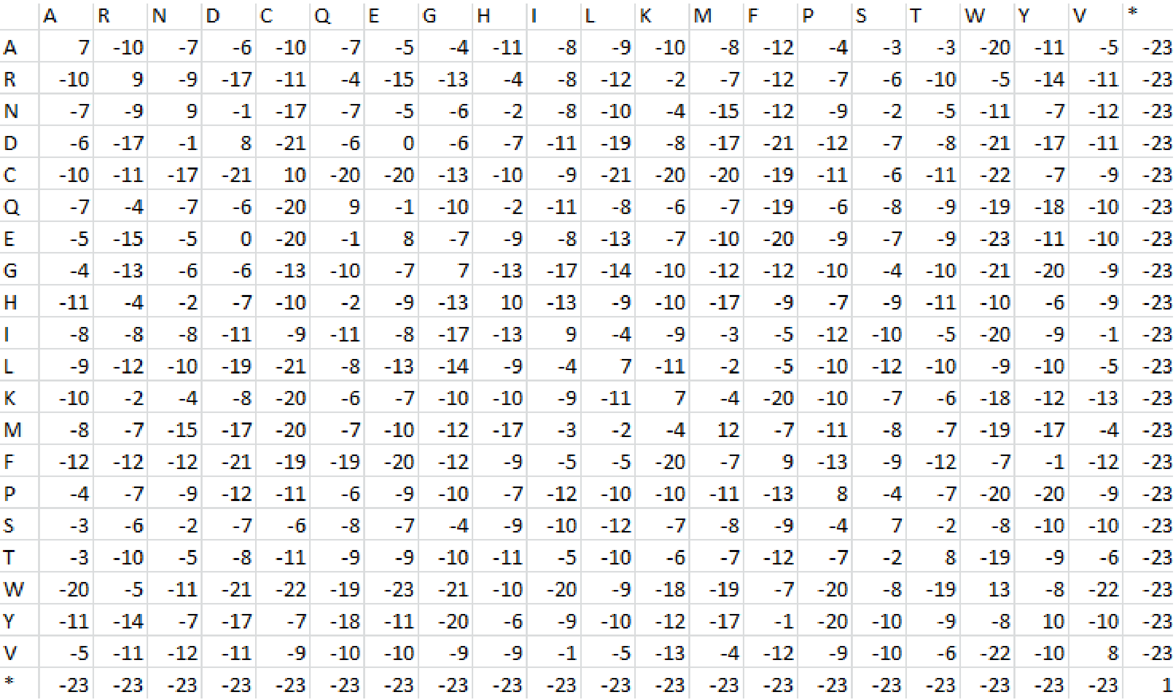


locus tag	gene name	sub-category
	norC	MDR efflux transporter
SAOUHSC_02629		antibiotic resistance
	tcaR	antibiotic resistance
	lmrB2	MDR efflux transporter
SAOUHSC_02797		antibiotic resistance
SAOUHSC_02826		antibiotic resistance
	slyA	antibiotic resistance

3.1.4 Substitution Matrix File

The amino acid substitution matrix file is tab–separated and is required for coding region analysis. The file needs to include a column and a row for each amino acid and one for gaps (*).

Example (PAM10):



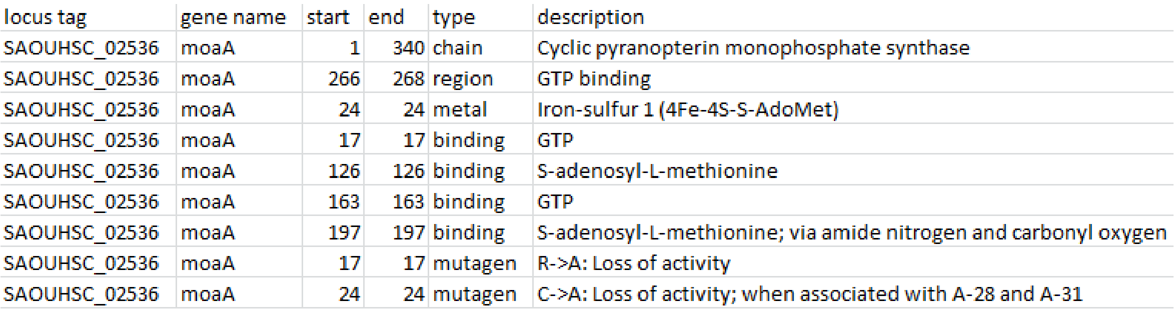
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	*
A	7	-10	-7	-6	-10	-7	-5	-4	-11	-8	-9	-10	-8	-12	-4	-3	-3	-20	-11	-5	-23
R	-10	9	-9	-17	-11	-4	-15	-13	-4	-8	-12	-2	-7	-12	-7	-6	-10	-5	-14	-11	-23
N	-7	-9	9	-1	-17	-7	-5	-6	-2	-8	-10	-4	-15	-12	-9	-2	-5	-11	-7	-12	-23
D	-6	-17	-1	8	-21	-6	0	-6	-7	-11	-19	-8	-17	-21	-12	-7	-8	-21	-17	-11	-23
C	-10	-11	-17	-21	10	-20	-20	-13	-10	-9	-21	-20	-20	-19	-11	-6	-11	-22	-7	-9	-23
Q	-7	-4	-7	-6	-20	9	-1	-10	-2	-11	-8	-6	-7	-19	-6	-8	-9	-19	-18	-10	-23
E	-5	-15	-5	0	-20	-1	8	-7	-9	-8	-13	-7	-10	-20	-9	-7	-9	-23	-11	-10	-23
G	-4	-13	-6	-6	-13	-10	-7	7	-13	-17	-14	-10	-12	-12	-10	-4	-10	-21	-20	-9	-23
H	-11	-4	-2	-7	-10	-2	-9	-13	10	-13	-9	-10	-17	-9	-7	-9	-11	-10	-6	-9	-23
I	-8	-8	-8	-11	-9	-11	-8	-17	-13	9	-4	-9	-3	-5	-12	-10	-5	-20	-9	-1	-23
L	-9	-12	-10	-19	-21	-8	-13	-14	-9	-4	7	-11	-2	-5	-10	-12	-10	-9	-10	-5	-23
K	-10	-2	-4	-8	-20	-6	-7	-10	-10	-9	-11	7	-4	-20	-10	-7	-6	-18	-12	-13	-23
M	-8	-7	-15	-17	-20	-7	-10	-12	-17	-3	-2	-4	12	-7	-11	-8	-7	-19	-17	-4	-23
F	-12	-12	-12	-21	-19	-19	-20	-12	-9	-5	-5	-20	-7	9	-13	-9	-12	-7	-1	-12	-23
P	-4	-7	-9	-12	-11	-6	-9	-10	-7	-12	-10	-10	-11	-13	8	-4	-7	-20	-20	-9	-23
S	-3	-6	-2	-7	-6	-8	-7	-4	-9	-10	-12	-7	-8	-9	-4	7	-2	-8	-10	-10	-23
T	-3	-10	-5	-8	-11	-9	-9	-10	-11	-5	-10	-6	-7	-12	-7	-2	8	-19	-9	-6	-23
W	-20	-5	-11	-21	-22	-19	-23	-21	-10	-20	-9	-18	-19	-7	-20	-8	-19	13	-8	-22	-23
Y	-11	-14	-7	-17	-7	-18	-11	-20	-6	-9	-10	-12	-17	-1	-20	-10	-9	-8	10	-10	-23
V	-5	-11	-12	-11	-9	-10	-10	-9	-9	-1	-5	-13	-4	-12	-9	-10	-6	-22	-10	8	-23
*	-23	-23	-23	-23	-23	-23	-23	-23	-23	-23	-23	-23	-23	-23	-23	-23	-23	-23	-23	-23	1

3.1.5 Protein Domains File

The protein domain file is tab–separated and is required for protein domain analysis. Each line, apart from the header, represents a protein domain.

Column Name	Description
locus tag	Unique locus tag of the gene.
gene name	General name of the gene.
start	Position of the first amino acid of the domain in the protein.
end	Position of the last amino acid of the domain in the protein.
type	Type of the domain, e.g. **chain**, **binding**,...
description	More detailed information on the domain. Multiple descriptions are separated by **’; ’**.

Example:



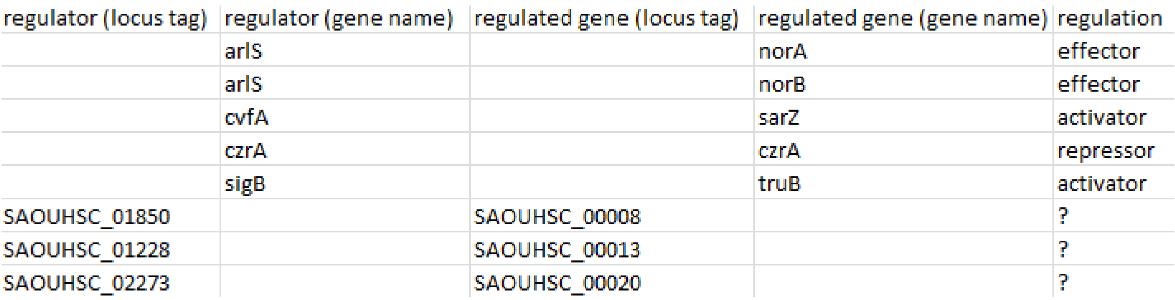
locus tag	gene name	start	end	type	description
SAOUHSC_02536	moaA	1	340	chain	Cyclic pyranopterin monophosphate synthase
SAOUHSC_02536	moaA	266	268	region	GTP binding
SAOUHSC_02536	moaA	24	24	metal	Iron-sulfur 1 (4Fe-4S-S-AdoMet)
SAOUHSC_02536	moaA	17	17	binding	GTP
SAOUHSC_02536	moaA	126	126	binding	S-adenosyl-L-methionine
SAOUHSC_02536	moaA	163	163	binding	GTP
SAOUHSC_02536	moaA	197	197	binding	S-adenosyl-L-methionine; via amide nitrogen and carbonyl oxygen
SAOUHSC_02536	moaA	17	17	mutagen	R->A: Loss of activity
SAOUHSC_02536	moaA	24	24	mutagen	C->A: Loss of activity; when associated with A-28 and A-31

3.1.6 Regulation File

The regulation file is tab–separated and is required for regulation analysis.

Column Name	Description
regulator (locus tag)	Unique locus tag of the regulator.
regulator (gene name)	General name of the regulator.
regulated gene (locus tag)	Unique locus tag of the regulated gene.
regulated gene (gene name)	General name of the regulated gene.
regulation	Type of the regulatory interaction. This can be either **activator** or **+**, **repressor** or **-**, **effector** or **+-**, or unknown.

Example:



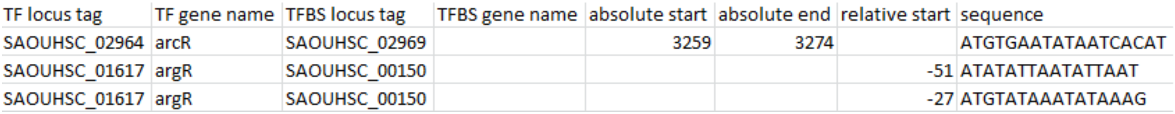
regulator (locus tag)	regulator (gene name)	regulated gene (locus tag)	regulated gene (gene name)	regulation
	arlS		norA	effector
	arlS		norB	effector
	cvfA		sarZ	activator
	czrA		czrA	repressor
	sigB		truB	activator
SAOUHSC_01850		SAOUHSC_00008		?
SAOUHSC_01228		SAOUHSC_00013		?
SAOUHSC_02273		SAOUHSC_00020		?

3.1.7 TFBS File

The transcription factor binding site (TFBS) file is tab–separated and is required for TFBS analysis. Each line, apart from the header, represents a transcription factor binding site.

Column Name	Description
TF locus tag	Unique locus tag of the transcription factor.
TF gene name	General name of the transcription factor.
TFBS locus tag	Unique locus tag of the regulated gene with the binding site.
TFBS gene name	General name of the regulated gene with the binding site.
absolute start	Start position of the TFBS in the genome on the plus strand.
absolute end	End position of the TFBS in the genome on the plus strand.
relative start	Binding site start in the regulated gene relative to the coding region start. The genome position is computed automatically and considers the strand the respective gene is on.
sequence	DNA sequence of the binding site on the strand of the regulated gene.

Example:

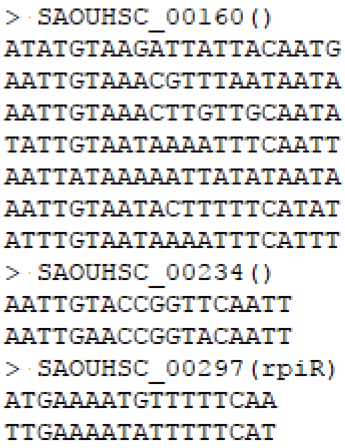


TF locus tag	TF gene name	TFBS locus tag	TFBS gene name	absolute start	absolute end	relative start	sequence
SAOUHSC_02964	arcR	SAOUHSC_02969		3259	3274		ATGTGAATATAATCACAT
SAOUHSC_01617	argR	SAOUHSC_00150				-51	ATATATTAATATTAAT
SAOUHSC_01617	argR	SAOUHSC_00150				-27	ATGTATAAATATAAAG

3.1.8 TF Motif MSA File

The transcription factor (TF) motif multiple sequence alignment (MSA) file is required for transcription factor binding site analysis. It has header lines specifying the locus tag of the transcription factor, followed by multiple rows of aligned sequences of its DNA binding domain. Each header begins with **'>'** followed by **locus tag(gene name)**. The aligned sequences need to be of equal length and are assumed to be on the respective strand.

Example:



```
> SAOUHSC_00160()
ATATGTAAGATTATTACAATG
AATTGTAAACGTTTAATAATA
AATTGTAAACTTGTTGCAATA
TATTGTAATAAAATTTCAATT
AATTATAAAAATTATATAATA
AATTGTAATACTTTTTCATAT
ATTTGTAATAAAATTTCATTT
> SAOUHSC_00234()
AATTGTACCGGTTCAATT
AATTGAACCGGTACAATT
> SAOUHSC_00297(rpiR)
ATGAAAATGTTTTTCAA
TTGAAAATATTTTTCAT
```

3.2 Results

The mutation analysis generates various files depending on the enabled analysis steps.

File Name	File Type	Description	Required Analysis Steps
log	.txt.	Contains warnings that occurred during the analysis without being severe enough to interrupt it.	none
results	.pdf	Tables, plots and p-values from various analysis steps, as well as descriptions. The plots are also available as individual .pdf and .png files, see below. The more analysis steps are enabled, the more information in the file.	none
genes mutations database	SQLite (.db)	Database with a table for gene and a table for mutation information that combines information from input files as well as from various analysis steps that are enabled. The more analysis steps are enabled, the more information in the database.	none
database gene table	.tsv	Gene table of the SQLite database.	none
database mutation table	.tsv	Mutation table of the SQLite database.	none
resistance table	.tsv	Table with all non–synonymous mutations in genes of interest or their direct and indirect regulators with scores of enabled analysis steps. The more analysis steps are enabled, the more information in the file.	genes of interest analysis
list of known mutations	.tsv	List with all known mutations in genes of interest from the protein domain file.	genes of interest and protein domain analysis
GRN complete	.gml	Complete gene regulatory network. (Visualisation tutorial)	regulation analysis
GRN interest	.gml	Gene regulatory network of the genes of interest sub–network that includes genes of interest and their direct and indirect regulators. (Visualisation tutorial)	regulation and genes of interest analysis
coding region score plot	.pdf, .png	Box plot comparing the scores of non–synonymous coding regions mutations in genes of interest and the remaining genes.	genes of interest and coding region analysis
mutation density plot	.pdf, .png	Box plot comparing the mutation density (number of mutations / kbp) in genes of interest and the remaining genes.	genes of interest and coding region analysis
mutation distribution plot	.pdf, .png	Bar plot comparing the distributions of non–synonymous mutations in genes of interest and the remaining genes depending on the gene regions.	genes of interest and coding region analysis
mutation type distribution plot	.pdf, .png	Bar plot comparing the distributions of non–synonymous mutations in genes of interest and the remaining genes depending on mutation type.	genes of interest analysis
TFBS score plot	.pdf, .png	Box plot comparing the scores of observed and randomly generated transcription factor binding site mutations in genes of interest and the remaining genes.	genes of interest analysis

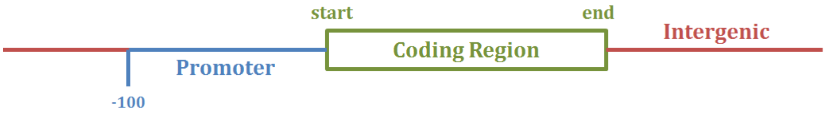
3.3 Advanced Settings

The advanced settings can be opened via **Settings → Advanced analysis settings** in the menu.

Relative default promoter start:

In the input gene file, it is possible to specify the promoter region for each gene. This can be done by either providing the start and end position of the promoter region in the genome, or by providing the promoter region start relative to the coding region start. In the latter case, the promoter region reaches from the provided relative start to the coding region start.

It happens that promoter region information is not available for all genes. In that case, this setting is used to give the gene a default promoter region ranging from the given relative start to the coding region start. This number needs to be smaller or equal 0. 0 means there is no promoter region.



Default: -100

Example:

strand	gene start	gene end	rel. promoter start	promoter start	promoter end
+	500	800	–100	400	499
–	500	800	–100	801	900
+	500	800	0	0	0

Adjust operon promoters and TFBS:

In bacteria, operons are a cluster of genes under the control of a single promoter. It is possible to specify a promoter region for each gene in the gene input file, regardless of the gene being in an operon. The same can be done for transcription factor binding sites in the transcription factor binding site input file.

If this setting is enabled, the promoter region of all genes belonging to an operon are set to the promoter region of the first gene in the operon. Furthermore, all transcription factor binding sites in the operon are associated with all genes in the operon.

Default: Enabled

Example: Let there be an operon consisting of genes A, B and C in that order with the following input promoter region positions and transcription factor binding sites

- gene A: promoter from 100 to 150, and TFBS 1
- gene B: 400 to 350
- gene C: 700 to 750, and TFBS 2

If this setting is enabled, the promoter region of A, B and C is set to 100 to 150, and all three genes are associated with TFBS 1 and 2.

Number of random mutations:

The transcription factor binding site analysis generates a number of random mutations for each observed mutation in a transcription factor binding site. This allows to examine its impact on the binding site.

This number needs to be greater or equal 0. Furthermore, it is advisable to choose a number smaller than the (average) length of the binding sites in order to avoid generating duplicate mutations.

Default: 15

Example: For *Staphylococcus aureus*, the mean length of the available transcription factor binding sites was 17 base pairs with a standard deviation of 3 base pairs. The number of random mutations chosen for the analysis was 15.

Minimum number of sequences:
The transcription factor binding site analysis uses position weight matrices of transcription factors to estimate the potential impact of transcription factor binding site mutations. The position weight matrices are constructed from the transcription factor sequences in the TF MSA input file.

If the number of sequences available for a transcription factor is below this number, the transcription factor and associated binding sites are not considered in the transcription factor binding site analysis. This number must be greater or equal 0.

Default: 5

Genes of interest / other genes: Genes of interest analysis allows to compare mutations in between two categories of genes. These four settings specify the long and short names for the two categories that are going to be used in the result tables, plots and descriptions.

Default: *genes of interest* (short *GOI*) and *other genes* (short *other*).

Example: When studying antibiotic resistance, the two categories might be called *antibiotic resistance* and *non–antibiotic resistance*, with the abbreviations *AR* and *non–AR*, respectively.

Open result directory:
If this setting is enabled, the mutation analysis result directory is opened in the operating system's file explorer after completing all analysis steps.

Save:
After making sure the new settings are valid, the advanced settings are saved to the user configuration file and the settings window is closed.

Cancel:
All changes to the advanced settings are discarded and the settings window is closed.

3.4 Runtime Examples

***E. coli* data set** 4,565 genes; 105,330 mutations; 266,482 protein domain lines; 3,864 regulation lines; 5,507 TFBS lines

***S. aureus* data set** 2,976 genes; 22,506 mutations; 8,496 protein domain lines; 761 regulation lines; 322 TFBS lines

Computer Windows 10 laptop, 64–bit, Intel i7-7700HQ 2.8Ghz CPU (only 1 core used)

Step	***E. coli***	***S. aureus***
input pre–processing	9.2s	1.3s
coding region analysis	35.0s	11.2s
TFBS analysis	0.1s	0.0s
general statistics	2.5s	0.5s
plot generation	2.8s	2.4s
GRN generation	0.1s	0.0s
writing output	12.4s	8.3s
total	**62.1s**	**23.9s**

3.5 GRN Visualisation

If **regulation analysis** is enabled, MutaNET will generate a gene regulatory network in .gml format (graph modelling language). This .gml file can be processed by Cytoscape [4], a program for modelling biological interaction networks.

Each node in the .gml file contains information on the (sub–)category of interest it belongs to. Furthermore, the node label contains the number of mutations in the gene:

(# non–syn. coding region mutations, # promoter mutations, # TFBS mutations)

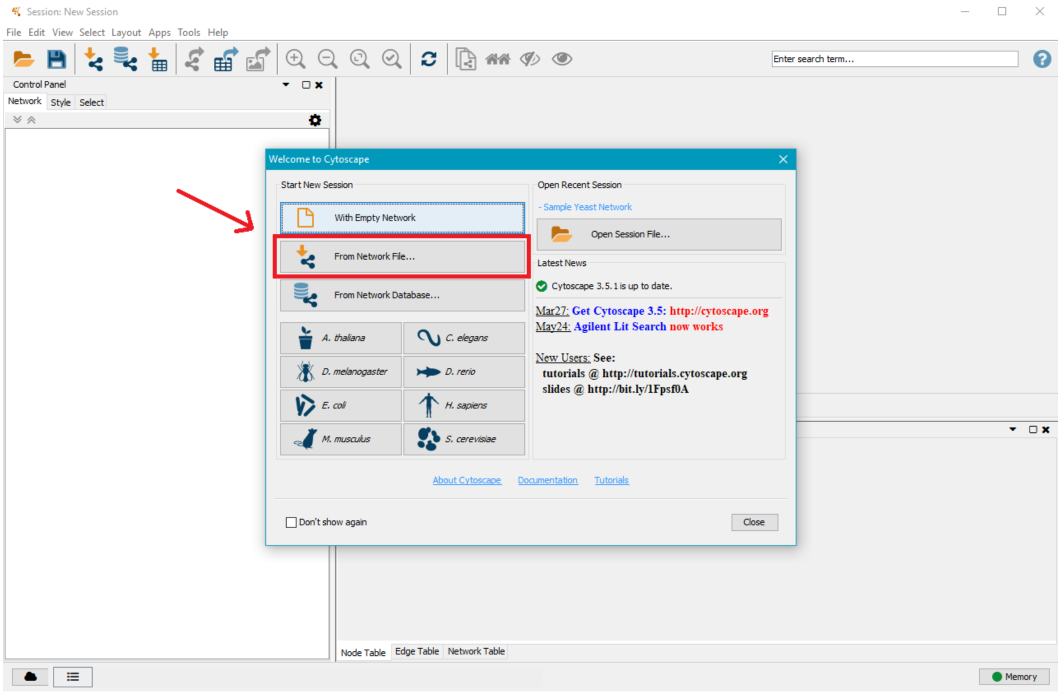
Each edge contains information on the regulation type: operon (**O**), activation (**A**), repression (**R**), effector (**E**, can act as activator or repressor) and unknown (**?**)

The following tutorial gives an introduction to how to customise .gml files in Cytoscape.

1. Go to cytoscape.org [4] and click on **download**.

2. Navigate to the download folder and execute the **installation file** you just downloaded and follow the installation instructions. It is very straightforward.

3. Open **Cytoscape** and click on **From Network File...**

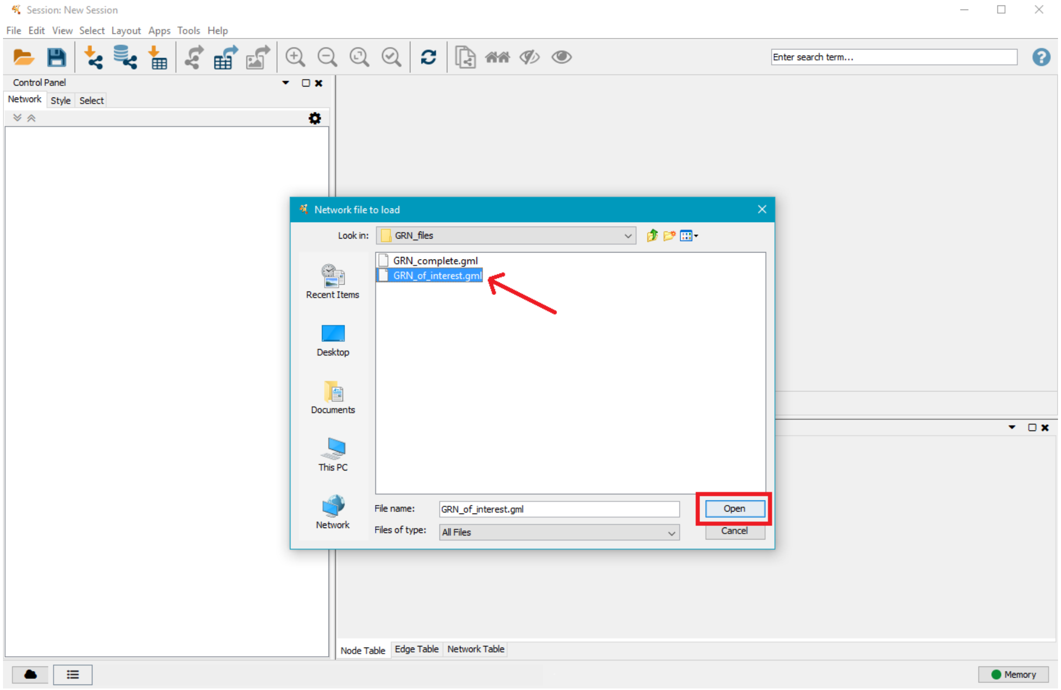


Cytoscape 3.5.1 is up to date.

Mar27: Get Cytoscape 3.5: http://cytoscape.org
May24: Agilent Lit Search now works

New Users: See:
tutorials @ http://tutorials.cytoscape.org
slides @ http://bit.ly/1Fpsf0A

4. Navigate to the **mutation analysis results** directory and then to the **GRN** directory contained within, where you will find one or two .gml files. Select the one you want to visualise and click on **Open**.

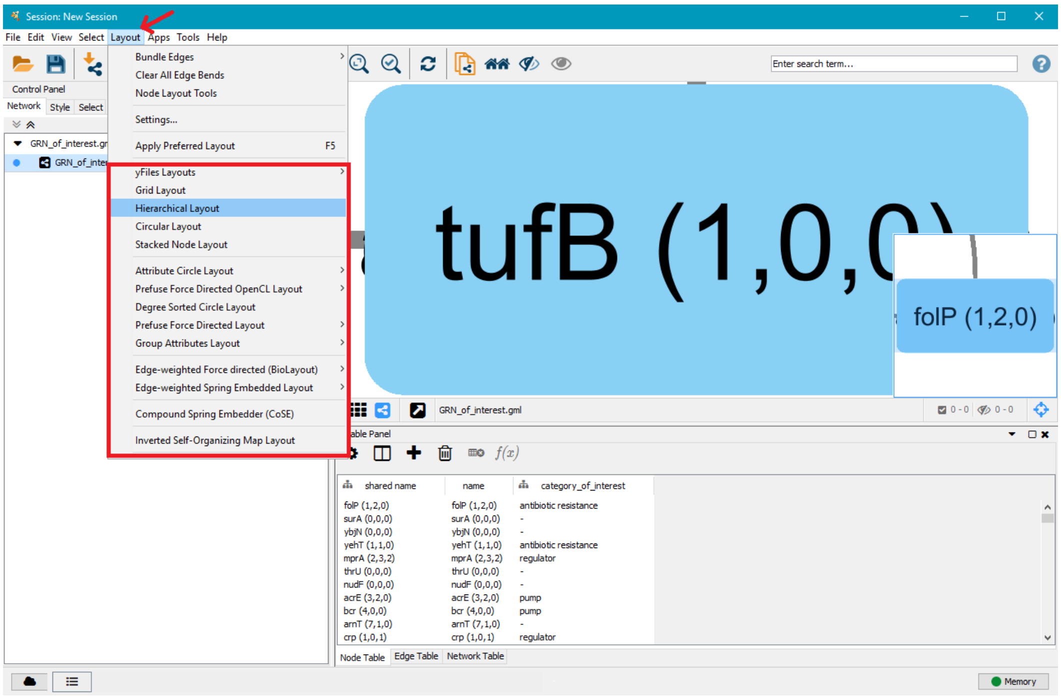


Look in: GRN_files

File name: GRN_of_interest.gml

Files of type: All Files

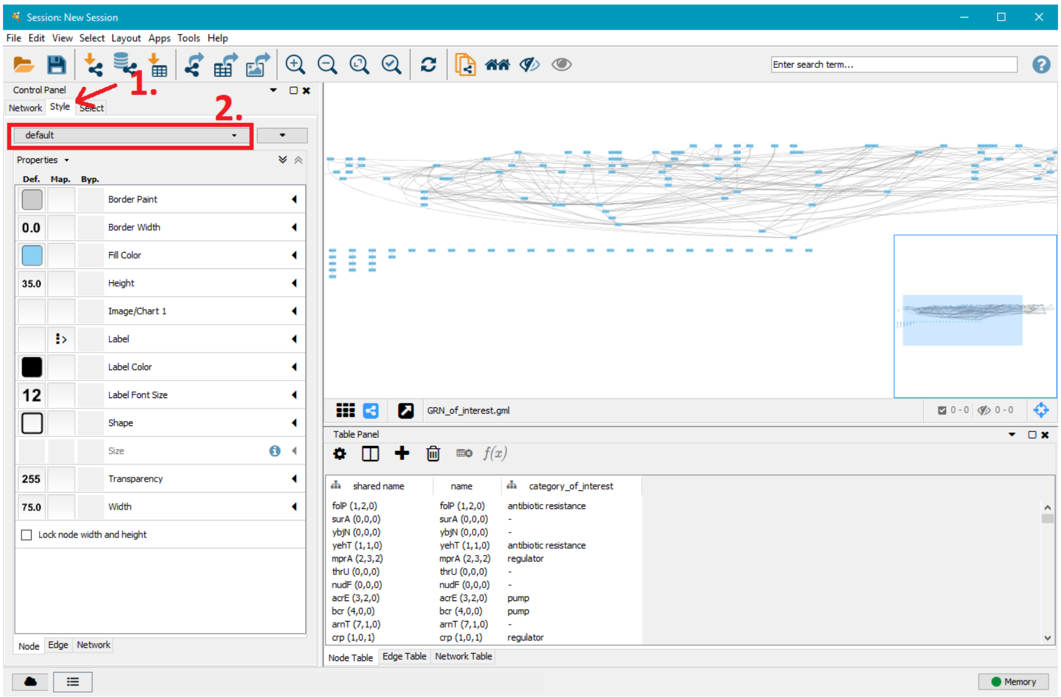
5. On the right side you will see a large node. That is the current network view. Below is a table with the nodes in the network, with an additional tab for edges. In the top menu bar, click on **Layout** and select the layout type you wish to apply to the network. In this example we selected **Hierarchical Layout**, but you can try several layouts and select the one you like the best.



shared name	name	category_of_interest
folP (1,2,0)	folP (1,2,0)	antibiotic resistance
surA (0,0,0)	surA (0,0,0)	-
ybjN (0,0,0)	ybjN (0,0,0)	-
yehT (1,1,0)	yehT (1,1,0)	antibiotic resistance
mprA (2,3,2)	mprA (2,3,2)	regulator
thrU (0,0,0)	thrU (0,0,0)	-
nudF (0,0,0)	nudF (0,0,0)	-
acrE (3,2,0)	acrE (3,2,0)	pump
bcr (4,0,0)	bcr (4,0,0)	pump
arnT (7,1,0)	arnT (7,1,0)	-
crp (1,0,1)	crp (1,0,1)	regulator

6. On the right side you can now see that the nodes and edges have been ordered according to the selected layout.You can zoom in and out of the network using the scroll wheel of your mouse, and view different parts of the network by left–clicking and dragging your mouse in the network window.

   In the top left, click on the **Style** tab in order to customise the look of your nodes and edges. First, select a network style by clicking on the dropdown menu currently saying **default**. Select **directed**.

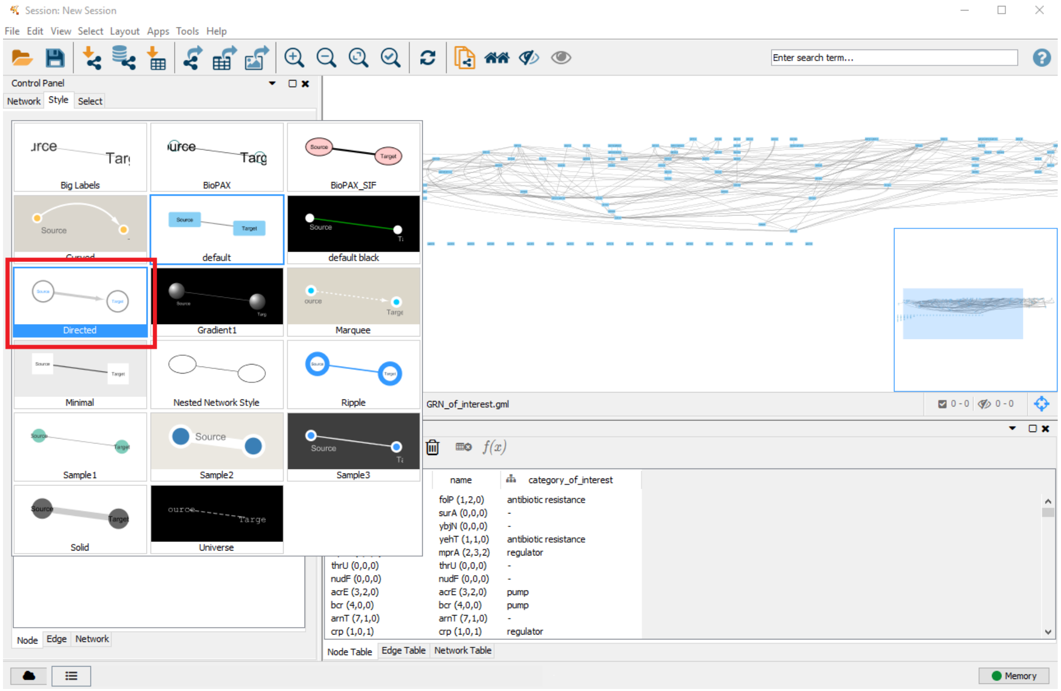


default

Def.	Map.	Byp.	
■			Border Paint
0.0			Border Width
■			Fill Color
35.0			Height
			Image/Chart 1
			Label
■			Label Color
12			Label Font Size
□			Shape
			Size
255			Transparency
75.0			Width

☐ Lock node width and height

shared name	name	category_of_interest
folP (1,2,0)	folP (1,2,0)	antibiotic resistance
surA (0,0,0)	surA (0,0,0)	-
ybjN (0,0,0)	ybjN (0,0,0)	-
yehT (1,1,0)	yehT (1,1,0)	antibiotic resistance
mprA (2,3,2)	mprA (2,3,2)	regulator
thrU (0,0,0)	thrU (0,0,0)	-
nudF (0,0,0)	nudF (0,0,0)	-
acrE (3,2,0)	acrE (3,2,0)	pump
bcr (4,0,0)	bcr (4,0,0)	pump
arnT (7,1,0)	arnT (7,1,0)	-
crp (1,0,1)	crp (1,0,1)	regulator



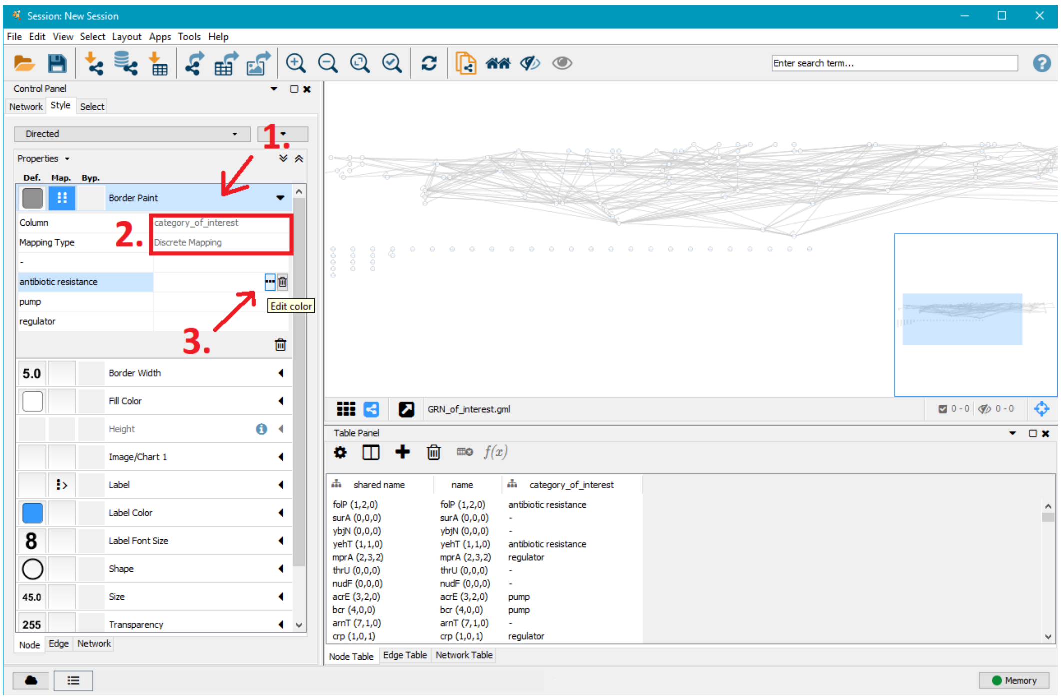
name	category_of_interest
folP (1,2,0)	antibiotic resistance
surA (0,0,0)	-
ybjN (0,0,0)	-
yehT (1,1,0)	antibiotic resistance
mprA (2,3,2)	regulator
thrU (0,0,0)	-
nudF (0,0,0)	-
acrE (3,2,0)	pump
bcr (4,0,0)	pump
arnT (7,1,0)	-
crp (1,0,1)	regulator

7. In the left menu you can now customise the colours, shapes, sizes,... of the network nodes. The .gml file gives each node information about the genes of interest (sub–)category it belongs to and Cytoscape allows to apply specific styles to these different node types.

   In this example we apply different border colours to the nodes depending on their antibiotic resistance. This approach works for all other options given in the style menu.

   1. Click on Border Paint to show more options.
   2. Click in the field next to Column and select category_of_interest.
   3. Click in the field next to Mapping Type and select Discrete Mapping. This will show additional rows listing the (sub–)categories specified in the mutation analysis.

      In this example, the sub–categories are antibiotic resistance, pump (for multidrug resistance efflux pumps), regulator (for direct multidrug resistance efflux pump regulators) and - (for non–antibiotic resistance genes).
   4. Click on the field next to one of the sub–categories, here antibiotic resistance and click on ... to select a border colour of your choice for that node type.



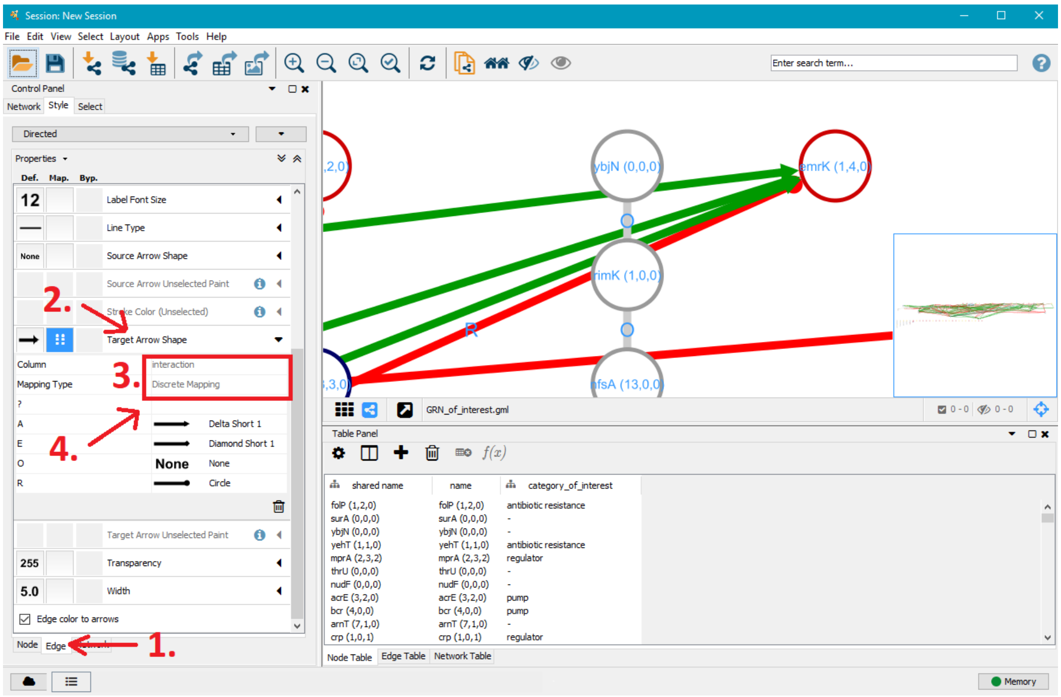
Border Paint
Column: category_of_interest
Mapping Type: Discrete Mapping

-	
antibiotic resistance	
pump	
regulator	

Def.	Property
5.0	Border Width
	Fill Color
	Height
	Image/Chart 1
	Label
	Label Color
8	Label Font Size
	Shape
45.0	Size
255	Transparency

shared name	name	category_of_interest
folP (1,2,0)	folP (1,2,0)	antibiotic resistance
surA (0,0,0)	surA (0,0,0)	-
ybjN (0,0,0)	ybjN (0,0,0)	-
yehT (1,1,0)	yehT (1,1,0)	antibiotic resistance
mprA (2,3,2)	mprA (2,3,2)	regulator
thrU (0,0,0)	thrU (0,0,0)	-
nudF (0,0,0)	nudF (0,0,0)	-
acrE (3,2,0)	acrE (3,2,0)	pump
bcr (4,0,0)	bcr (4,0,0)	pump
arnT (7,1,0)	arnT (7,1,0)	-
crp (1,0,1)	crp (1,0,1)	regulator

8. After you customised the nodes to your liking, click on the **Edge** tab in the bottom left. This will open a similar menu for edge customisation that works just like the one for nodes. Instead of **category_of_interest**, select **interaction** next to **Column**. This will give you the option to specifically target operon (**O**), activation (**A**), repression (**R**), effector (**E**, can act as activator or repressor) and unknown (**?**) interactions.



Def.	Property
12	Label Font Size
	Line Type
None	Source Arrow Shape
	Source Arrow Unselected Paint
	Stroke Color (Unselected)
	Target Arrow Shape

Column: interaction
Mapping Type: Discrete Mapping

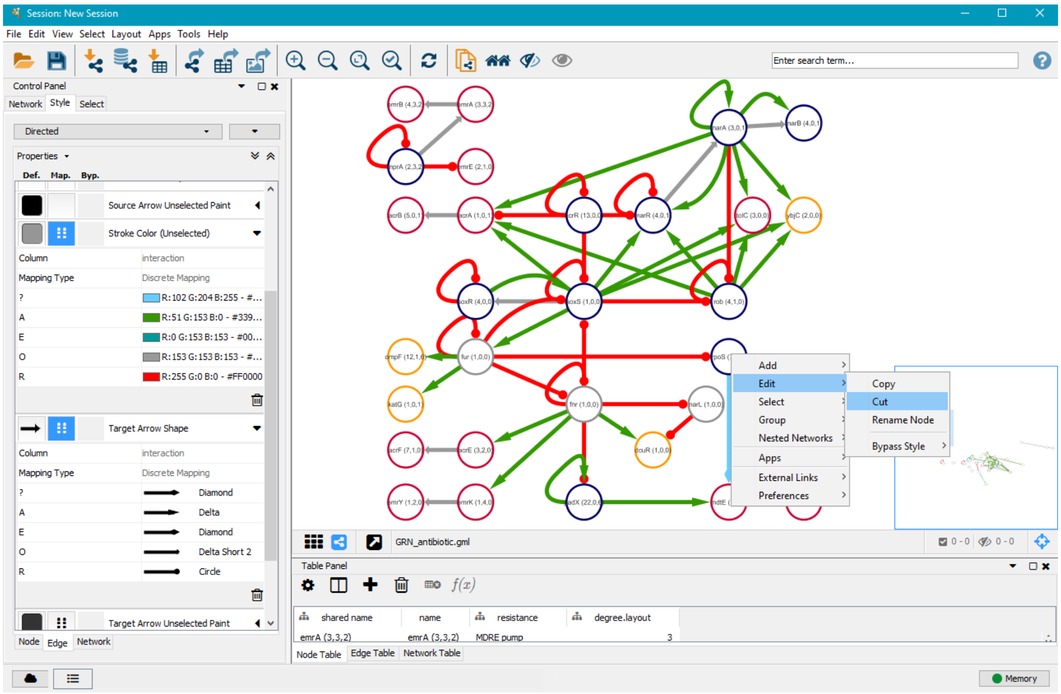
?	
A	Delta Short 1
E	Diamond Short 1
O	None
R	Circle

Def.	Property
	Target Arrow Unselected Paint
255	Transparency
5.0	Width

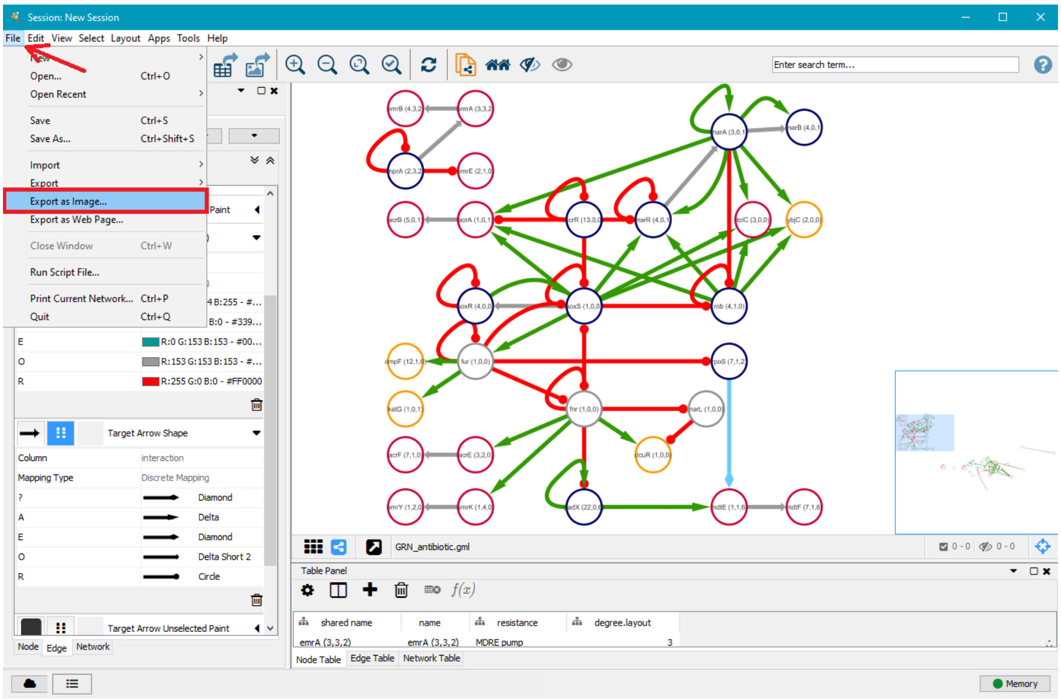
☑ Edge color to arrows

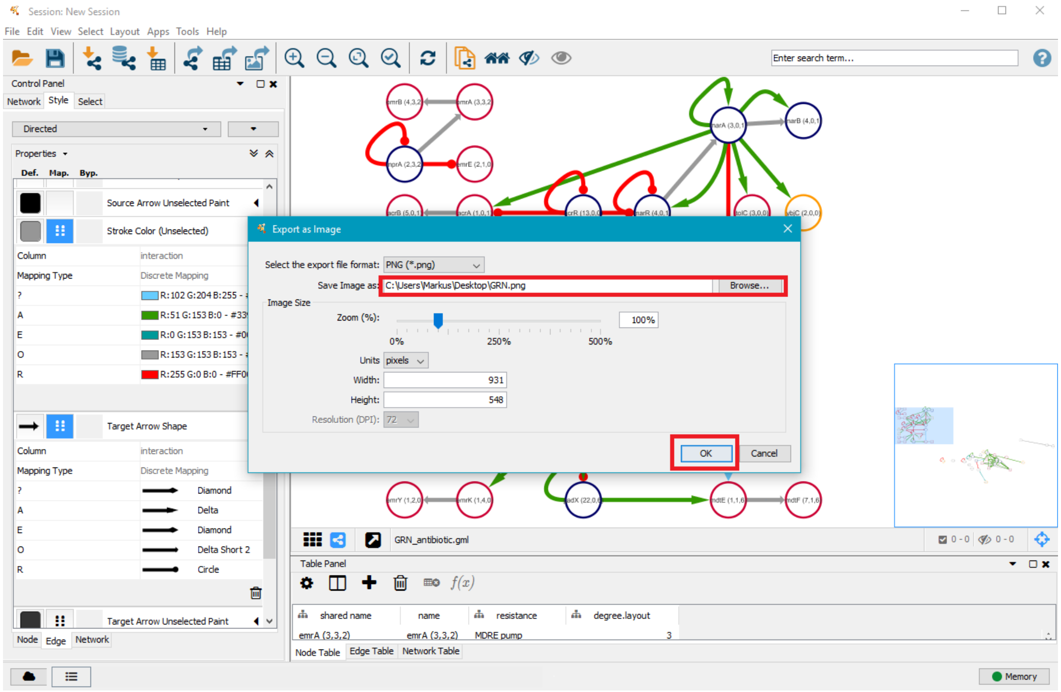
shared name	name	category_of_interest
folP (1,2,0)	folP (1,2,0)	antibiotic resistance
surA (0,0,0)	surA (0,0,0)	-
ybjN (0,0,0)	ybjN (0,0,0)	-
yehT (1,1,0)	yehT (1,1,0)	antibiotic resistance
mprA (2,3,2)	mprA (2,3,2)	regulator
thrU (0,0,0)	thrU (0,0,0)	-
nudF (0,0,0)	nudF (0,0,0)	-
acrE (3,2,0)	acrE (3,2,0)	pump
bcr (4,0,0)	bcr (4,0,0)	pump
arnT (7,1,0)	arnT (7,1,0)	-
crp (1,0,1)	crp (1,0,1)	regulator

9. After you customised the edges to your liking, it might be necessary to manually order the network and maybe even remove some unimportant or less important nodes and edges. You can left–click on nodes and drag them to other positions. You can delete nodes and edges by right–clicking on them, selecting **Edit** and then **Cut**.



10. Once you are satisfied with your network, you can export it as a **.png**, **.pdf**, **.jpeg**, **.svg** or **.ps** file by clicking on **File** in the top menu and then selecting **Export as Image...**. Choose the file type and where you want to save it, and then click on **OK**.





Source Arrow Unselected Paint

Stroke Color (Unselected)

Column	interaction
Mapping Type	Discrete Mapping
?	R:102 G:204 B:255 -
A	R:51 G:153 B:0 - #33
E	R:0 G:153 B:153 - #0
O	R:153 G:153 B:153 -
R	R:255 G:0 B:0 - #FF0

Target Arrow Shape

Column	interaction
Mapping Type	Discrete Mapping
?	Diamond
A	Delta
E	Diamond
O	Delta Short 2
R	Circle

Target Arrow Unselected Paint

Export as Image

Select the export file format: PNG (*.png)

Save Image as: C:\Users\Markus\Desktop\GRN.png Browse...

Image Size

Zoom (%): 100%

0% 250% 500%

Units pixels

Width: 931

Height: 548

Resolution (DPI): 72

OK Cancel

shared name	name	resistance	degree.layout
emrA (3,3,2)	emrA (3,3,2)	MDRE pump	3

4 File Converter and File Merger

Different file converters and file mergers can be opened via **Tools** in the menu.

4.1 Mutation VCF Merger

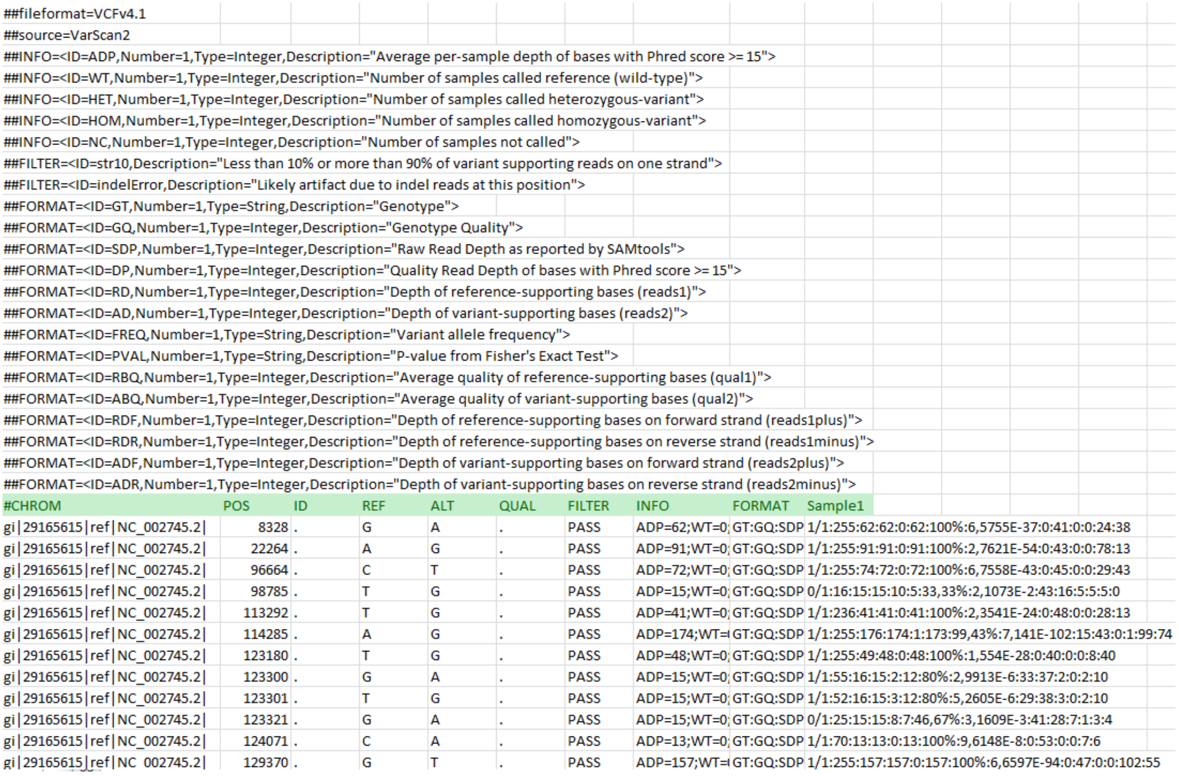
This tool merges variant call format (.vcf) files with mutation information into a single tab–separated (.tsv) file. That file can be used in the mutation analysis of MutaNET and contains information on the genome position, reference and alternative DNA base(s) of the mutations.

4.1.1 VCF Input Directory

A directory that contains variant call format (.vcf) files with mutation information. These files can also be in sub–directories.

Lines beginning with ## are comment lines and are ignored. The file needs to contain tab–separated columns with a header line directly under the first comments. This header needs to contain at least the columns **POS**, **REF** and **ALT**.

Example VCF File: Marked in green is the header with column names.



```
##fileformat=VCFv4.1
##source=VarScan2
##INFO=<ID=ADP,Number=1,Type=Integer,Description="Average per-sample depth of bases with Phred score >= 15">
##INFO=<ID=WT,Number=1,Type=Integer,Description="Number of samples called reference (wild-type)">
##INFO=<ID=HET,Number=1,Type=Integer,Description="Number of samples called heterozygous-variant">
##INFO=<ID=HOM,Number=1,Type=Integer,Description="Number of samples called homozygous-variant">
##INFO=<ID=NC,Number=1,Type=Integer,Description="Number of samples not called">
##FILTER=<ID=str10,Description="Less than 10% or more than 90% of variant supporting reads on one strand">
##FILTER=<ID=indelError,Description="Likely artifact due to indel reads at this position">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=SDP,Number=1,Type=Integer,Description="Raw Read Depth as reported by SAMtools">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Quality Read Depth of bases with Phred score >= 15">
##FORMAT=<ID=RD,Number=1,Type=Integer,Description="Depth of reference-supporting bases (reads1)">
##FORMAT=<ID=AD,Number=1,Type=Integer,Description="Depth of variant-supporting bases (reads2)">
##FORMAT=<ID=FREQ,Number=1,Type=String,Description="Variant allele frequency">
##FORMAT=<ID=PVAL,Number=1,Type=String,Description="P-value from Fisher's Exact Test">
##FORMAT=<ID=RBQ,Number=1,Type=Integer,Description="Average quality of reference-supporting bases (qual1)">
##FORMAT=<ID=ABQ,Number=1,Type=Integer,Description="Average quality of variant-supporting bases (qual2)">
##FORMAT=<ID=RDF,Number=1,Type=Integer,Description="Depth of reference-supporting bases on forward strand (reads1plus)">
##FORMAT=<ID=RDR,Number=1,Type=Integer,Description="Depth of reference-supporting bases on reverse strand (reads1minus)">
##FORMAT=<ID=ADF,Number=1,Type=Integer,Description="Depth of variant-supporting bases on forward strand (reads2plus)">
##FORMAT=<ID=ADR,Number=1,Type=Integer,Description="Depth of variant-supporting bases on reverse strand (reads2minus)">
#CHROM                        POS     ID  REF  ALT  QUAL  FILTER  INFO            FORMAT      Sample1
gi|29165615|ref|NC_002745.2|    8328  .   G    A    .     PASS    ADP=62;WT=0;    GT:GQ:SDP   1/1:255:62:62:0:62:100%:6,5755E-37:0:41:0:0:24:38
gi|29165615|ref|NC_002745.2|   22264  .   A    G    .     PASS    ADP=91;WT=0;    GT:GQ:SDP   1/1:255:91:91:0:91:100%:2,7621E-54:0:43:0:0:78:13
gi|29165615|ref|NC_002745.2|   96664  .   C    T    .     PASS    ADP=72;WT=0;    GT:GQ:SDP   1/1:255:74:72:0:72:100%:6,7558E-43:0:45:0:0:29:43
gi|29165615|ref|NC_002745.2|   98785  .   T    G    .     PASS    ADP=15;WT=0;    GT:GQ:SDP   0/1:16:15:15:10:5:33,33%:2,1073E-2:43:16:5:5:5:0
gi|29165615|ref|NC_002745.2|  113292  .   T    G    .     PASS    ADP=41;WT=0;    GT:GQ:SDP   1/1:236:41:41:0:41:100%:2,3541E-24:0:48:0:0:28:13
gi|29165615|ref|NC_002745.2|  114285  .   A    G    .     PASS    ADP=174;WT=     GT:GQ:SDP   1/1:255:176:174:1:173:99,43%:7,141E-102:15:43:0:1:99:74
gi|29165615|ref|NC_002745.2|  123180  .   T    G    .     PASS    ADP=48;WT=0;    GT:GQ:SDP   1/1:255:49:48:0:48:100%:1,554E-28:0:40:0:0:8:40
gi|29165615|ref|NC_002745.2|  123300  .   G    A    .     PASS    ADP=15;WT=0;    GT:GQ:SDP   1/1:55:16:15:2:12:80%:2,9913E-6:33:37:2:0:2:10
gi|29165615|ref|NC_002745.2|  123301  .   T    G    .     PASS    ADP=15;WT=0;    GT:GQ:SDP   1/1:52:16:15:3:12:80%:5,2605E-6:29:38:3:0:2:10
gi|29165615|ref|NC_002745.2|  123321  .   G    A    .     PASS    ADP=15;WT=0;    GT:GQ:SDP   0/1:25:15:15:8:7:46,67%:3,1609E-3:41:28:7:1:3:4
gi|29165615|ref|NC_002745.2|  124071  .   C    A    .     PASS    ADP=13;WT=0;    GT:GQ:SDP   1/1:70:13:13:0:13:100%:9,6148E-8:0:53:0:0:7:6
gi|29165615|ref|NC_002745.2|  129370  .   G    T    .     PASS    ADP=157;WT=     GT:GQ:SDP   1/1:255:157:157:0:157:100%:6,6597E-94:0:47:0:0:102:55
```

4.1.2 Result Mutation File

The result file is tab–separated (.tsv) and contains a header with column names in the first line. The mutations are ordered by their position in the genome and duplicates are eliminated. See Section 3.1.2 for a description of the columns as well as an example.

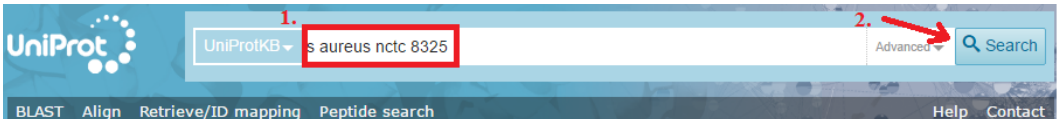
4.2 UniProt Protein Domain Converter

This tool parses a UniProt [5] database text (.txt) file with protein entries, extracts protein domain information and writes them into a protein domain tab–separated (.tsv) file that can be used for protein domain analysis.

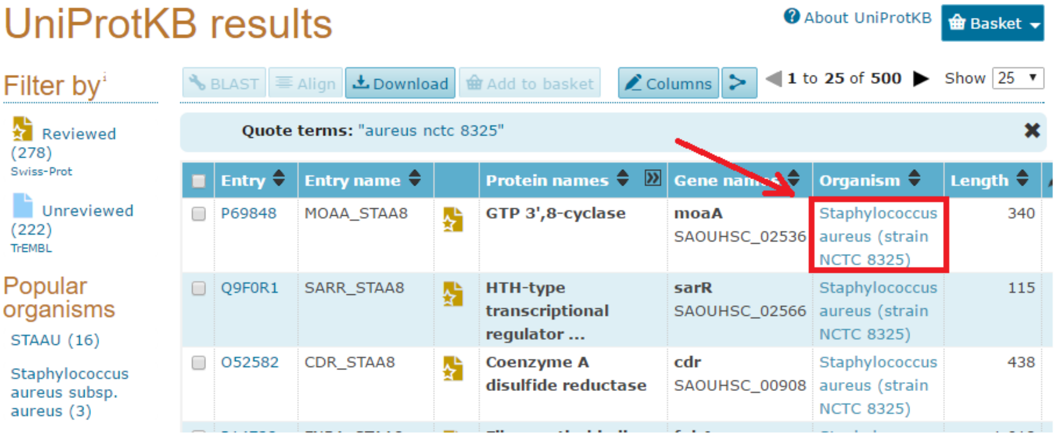
4.2.1 Download an UniProt Database Text File

1. Go to UniProt.

2. Enter the name of the organism into the search bar and either press **Enter** or click on **Search**. For example, for *Staphylococcus aures* strain NCTC 8325, enter **s aureus NCTC 8325**. If the database contains information on that organism, a table with protein entries is shown.



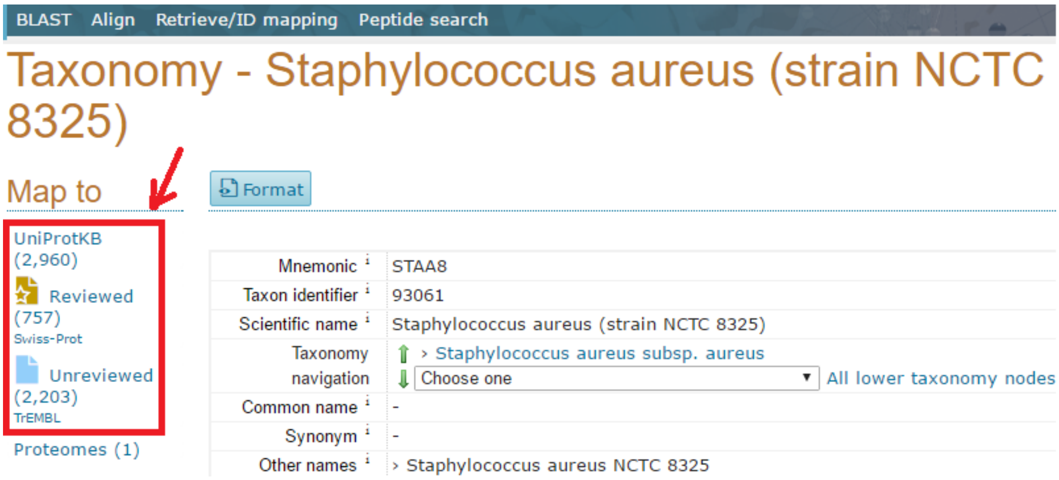
3. In the table, click on the name of your organism in the **Organism** column. This opens the taxonomy page for that organism.



Quote terms: "aureus nctc 8325"

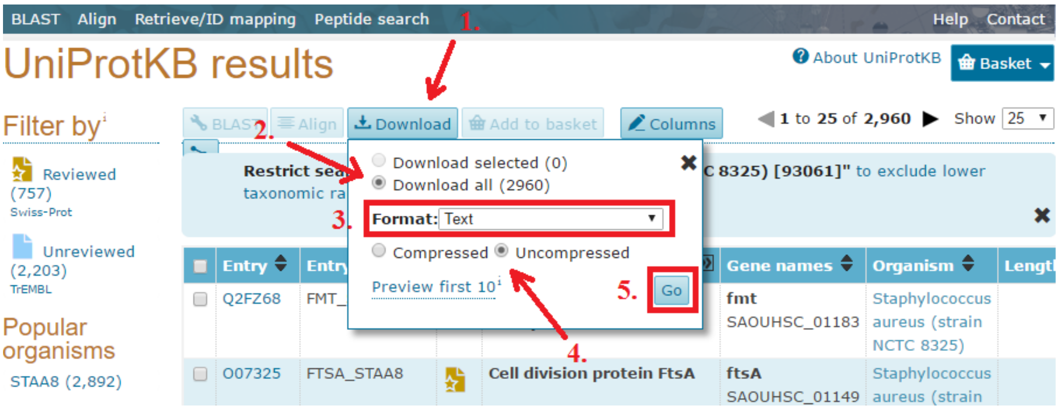
	Entry	Entry name		Protein names	Gene names	Organism	Length
□	P69848	MOAA_STAA8		GTP 3',8-cyclase	moaA SAOUHSC_02536	Staphylococcus aureus (strain NCTC 8325)	340
□	Q9F0R1	SARR_STAA8		HTH-type transcriptional regulator ...	sarR SAOUHSC_02566	Staphylococcus aureus (strain NCTC 8325)	115
□	O52582	CDR_STAA8		Coenzyme A disulfide reductase	cdr SAOUHSC_00908	Staphylococcus aureus (strain NCTC 8325)	438

4. On the left under **Map to**, click on **UniProtKB**, **Reviewed** or **Unreviewed** depending on the kind of data you want to convert. In this example, **UniProtKB** was selected. This opens a table with all protein entries for that organism in the database.


BLAST
Align
Retrieve/ID mapping
Peptide search
Taxonomy - Staphylococcus aureus (strain NCTC 8325)
Map to
UniProtKB (2,960)
Reviewed (757) Swiss-Prot
Unreviewed (2,203) TrEMBL
Proteomes (1)
Format

Mnemonic	STAA8
Taxon identifier	93061
Scientific name	Staphylococcus aureus (strain NCTC 8325)
Taxonomy navigation	› Staphylococcus aureus subsp. aureus Choose one ▾ All lower taxonomy nodes
Common name	-
Synonym	-
Other names	› Staphylococcus aureus NCTC 8325

5. Click on **Download**, choose **Text** in the **Format** drop down menu, select **Uncompressed** and click on **Go** to start the download.

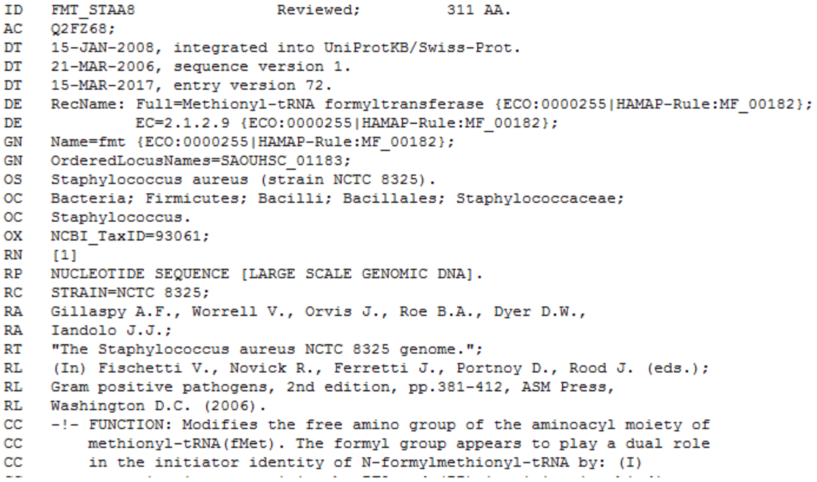


○ Download selected (0)
● Download all (2960)

Format: Text

○ Compressed ● Uncompressed

Example File:



```
ID   FMT_STAA8               Reviewed;         311 AA.
AC   Q2FZ68;
DT   15-JAN-2008, integrated into UniProtKB/Swiss-Prot.
DT   21-MAR-2006, sequence version 1.
DT   15-MAR-2017, entry version 72.
DE   RecName: Full=Methionyl-tRNA formyltransferase {ECO:0000255|HAMAP-Rule:MF_00182};
DE            EC=2.1.2.9 {ECO:0000255|HAMAP-Rule:MF_00182};
GN   Name=fmt {ECO:0000255|HAMAP-Rule:MF_00182};
GN   OrderedLocusNames=SAOUHSC_01183;
OS   Staphylococcus aureus (strain NCTC 8325).
OC   Bacteria; Firmicutes; Bacilli; Bacillales; Staphylococcaceae;
OC   Staphylococcus.
OX   NCBI_TaxID=93061;
RN   [1]
RP   NUCLEOTIDE SEQUENCE [LARGE SCALE GENOMIC DNA].
RC   STRAIN=NCTC 8325;
RA   Gillaspy A.F., Worrell V., Orvis J., Roe B.A., Dyer D.W.,
RA   Iandolo J.J.;
RT   "The Staphylococcus aureus NCTC 8325 genome.";
RL   (In) Fischetti V., Novick R., Ferretti J., Portnoy D., Rood J. (eds.);
RL   Gram positive pathogens, 2nd edition, pp.381-412, ASM Press,
RL   Washington D.C. (2006).
CC   -!- FUNCTION: Modifies the free amino group of the aminoacyl moiety of
CC       methionyl-tRNA(fMet). The formyl group appears to play a dual role
CC       in the initiator identity of N-formylmethionyl-tRNA by: (I)
```

4.2.2 Result Protein Domain File

The result file is tab–separated and contains all protein domains from the input UniProt database text file. These protein domains are sorted by the locus tag or gene name of the respective protein. This file can be used for protein domain analysis. See Section 3.1.5 for a description of the file columns as well as an example.

A description of different protein domain tags can be found in the UniProt user manual.

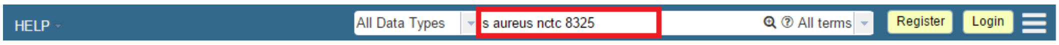
4.3 PATRIC Antibiotic Resistance Converter

This tool converts a PATRIC [6] antibiotic resistance comma-separated (.csv) file to a tab–separated (.tsv) file that can be used for antibiotic resistance analysis.

4.3.1 Download a PATRIC Antibiotic Resistance File

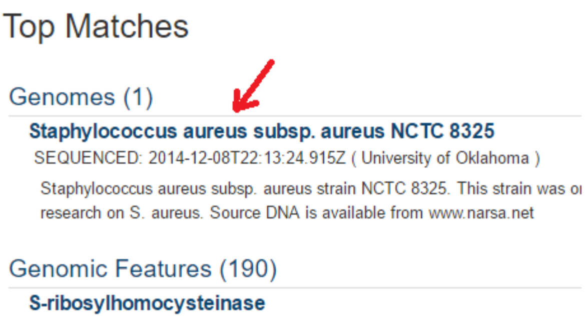
1. Go to PATRIC.

2. Enter the name of the organism into the search bar at the top and confirm by pressing **Enter**. For example, for *Staphylococcus aures* strain NCTC 8325, enter **s aureus NCTC 8325**.



s aureus nctc 8325

3. Click on your organism under **Genomes**. If it is not among the search results, PATRIC does not have the required information.



Top Matches

Genomes (1)

Staphylococcus aureus subsp. aureus NCTC 8325

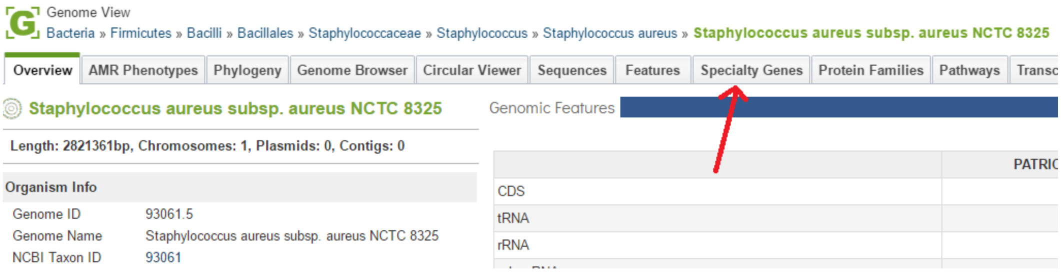
SEQUENCED: 2014-12-08T22:13:24.915Z (University of Oklahoma)

Staphylococcus aureus subsp. aureus strain NCTC 8325. This strain was o
research on S. aureus. Source DNA is available from www.narsa.net

Genomic Features (190)

S-ribosylhomocysteinase

4. Click on the **Specialty Genes** tab.



Staphylococcus aureus subsp. aureus NCTC 8325

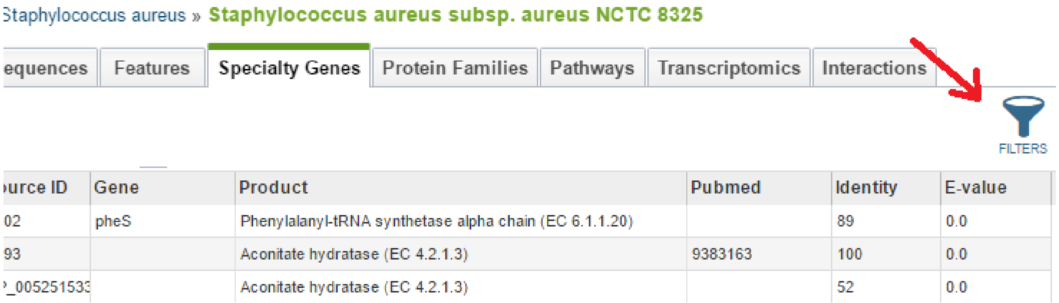
Length: 2821361bp, Chromosomes: 1, Plasmids: 0, Contigs: 0

Organism Info

Genome ID	93061.5
Genome Name	Staphylococcus aureus subsp. aureus NCTC 8325
NCBI Taxon ID	93061

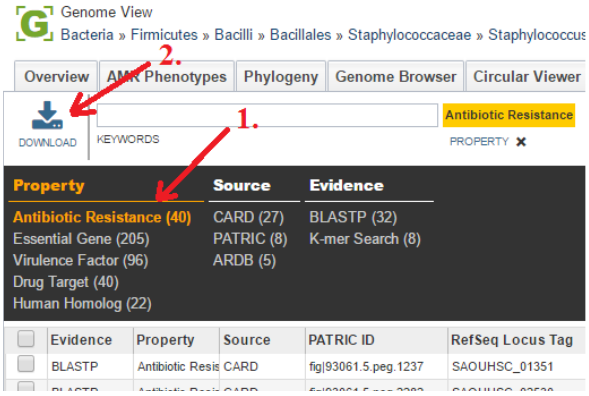
CDS
tRNA
rRNA

5. Click on **Filters**.



urce ID	Gene	Product	Pubmed	Identity	E-value
02	pheS	Phenylalanyl-tRNA synthetase alpha chain (EC 6.1.1.20)		89	0.0
93		Aconitate hydratase (EC 4.2.1.3)	9383163	100	0.0
_005251533		Aconitate hydratase (EC 4.2.1.3)		52	0.0

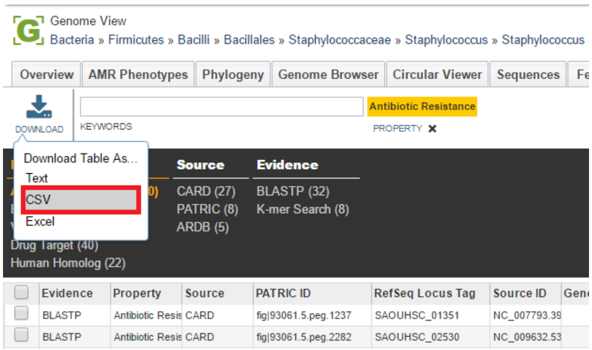
6. Select **Antibiotic Resistance** under **Property** to obtain a list of all antibiotic resistance genes in the database. Then click on **Download**.



Property	Source	Evidence
Antibiotic Resistance (40)	CARD (27)	BLASTP (32)
Essential Gene (205)	PATRIC (8)	K-mer Search (8)
Virulence Factor (96)	ARDB (5)	
Drug Target (40)		
Human Homolog (22)		

□	Evidence	Property	Source	PATRIC ID	RefSeq Locus Tag
□	BLASTP	Antibiotic Resis	CARD	fig|93061.5.peg.1237	SAOUHSC_01351

7. Select **CSV** to download the entire antibiotic resistance list.



□	Evidence	Property	Source	PATRIC ID	RefSeq Locus Tag	Source ID	Gen
□	BLASTP	Antibiotic Resis	CARD	fig|93061.5.peg.1237	SAOUHSC_01351	NC_007793.39	
□	BLASTP	Antibiotic Resis	CARD	fig|93061.5.peg.2282	SAOUHSC_02530	NC_009632.53	

Example File:

```
Evidence,Property,Source,Genome Name,PATRIC ID,RefSeq Locus Tag,Alt Locus Tag,Source ID,Source Organism
"K-mer Search","Antibiotic Resistance","PATRIC","Staphylococcus aureus subsp. aureus NCTC 8325","fig|93
"BLASTP","Antibiotic Resistance","CARD","Staphylococcus aureus subsp. aureus NCTC 8325","fig|93061.5.pe
"K-mer Search","Antibiotic Resistance","PATRIC","Staphylococcus aureus subsp. aureus NCTC 8325","fig|93
"BLASTP","Antibiotic Resistance","CARD","Staphylococcus aureus subsp. aureus NCTC 8325","fig|93061.5.pe
"BLASTP","Antibiotic Resistance","ARDB","Staphylococcus aureus subsp. aureus NCTC 8325","fig|93061.5.pe
"BLASTP","Antibiotic Resistance","CARD","Staphylococcus aureus subsp. aureus NCTC 8325","fig|93061.5.pe
"K-mer Search","Antibiotic Resistance","PATRIC","Staphylococcus aureus subsp. aureus NCTC 8325","fig|93
"BLASTP","Antibiotic Resistance","CARD","Staphylococcus aureus subsp. aureus NCTC 8325","fig|93061.5.pe
"BLASTP","Antibiotic Resistance","ARDB","Staphylococcus aureus subsp. aureus NCTC 8325","fig|93061.5.pe
```

4.3.2 Result Antibiotic Resistance File

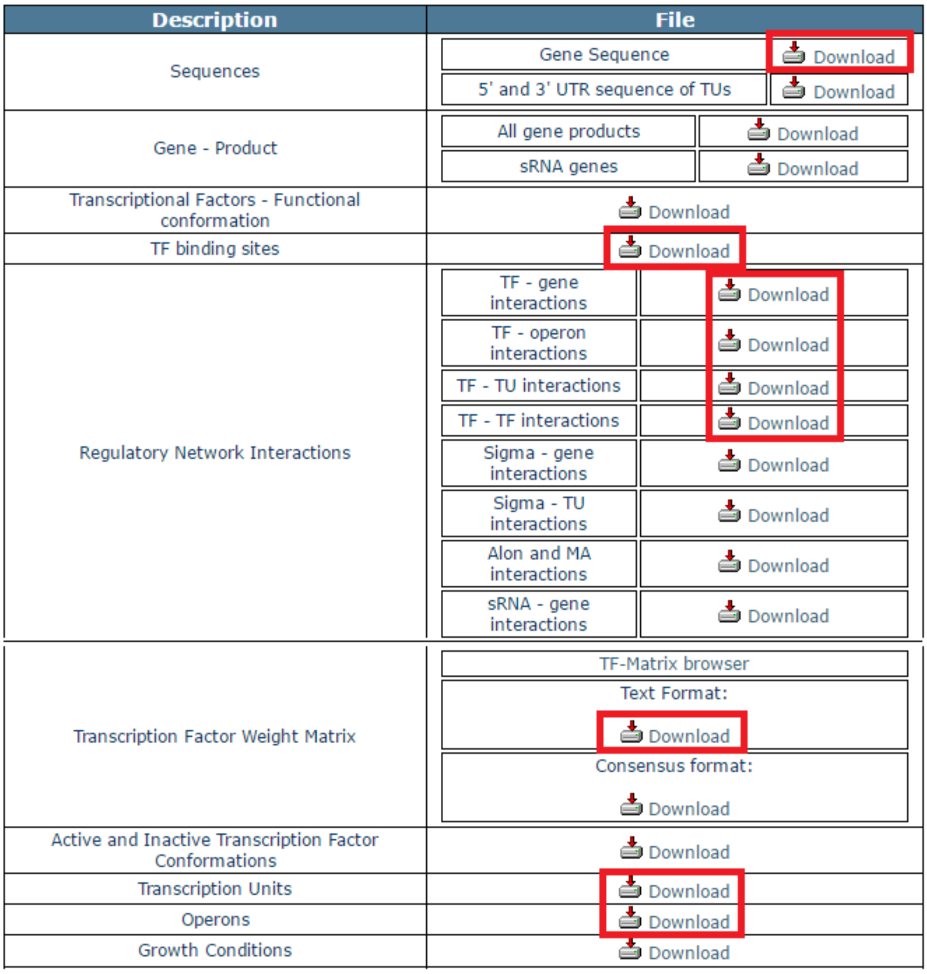
The result file is tab–separated and contains all locus tags and gene names of genes that were labelled as antibiotic resistant in the input file. This file can be used for antibiotic resistance analysis. See Section 3.1.3 for a description of the file columns as well as an example.

4.4 RegulonDB Converter

This tool takes several RegulonDB [7] files, merges their content and converts them to files suitable for mutation analysis.

4.4.1 Downloading RegulonDB Files

1. Go to RegulonDB.
2. Download the following files:
   - Gene sequence
   - TF binding sites
   - TF – gene interactions
   - TF – operon interactions
   - TF – TU interactions
   - TF – TF interactions
   - Transcription factor weight matrix – text format
   - Transcription units
   - Operons



Description	File	
Sequences	Gene Sequence	Download
	5' and 3' UTR sequence of TUs	Download
Gene - Product	All gene products	Download
	sRNA genes	Download
Transcriptional Factors - Functional conformation		Download
TF binding sites		Download
Regulatory Network Interactions	TF - gene interactions	Download
	TF - operon interactions	Download
	TF - TU interactions	Download
	TF - TF interactions	Download
	Sigma - gene interactions	Download
	Sigma - TU interactions	Download
	Alon and MA interactions	Download
	sRNA - gene interactions	Download
Transcription Factor Weight Matrix	TF-Matrix browser	
	Text Format:	Download
	Consensus format:	Download
Active and Inactive Transcription Factor Conformations		Download
Transcription Units		Download
Operons		Download
Growth Conditions		Download

4.4.2 Result Files

The converter yields the following result files that can be used for mutation analysis:

File Name	Type	Description	Example
genes	.tsv	General gene information. The gene names from RegulonDB are used as gene names and locus tags.	Section 3.1.1
regulation	.tsv	Transcription factors, regulated genes and type of regulation.	Section 3.1.6
TFBS	.tsv	Transcription factor binding sites mapped to the regulated genes.	Section 3.1.7
TF MSA	.fasta	Multiple sequence alignments of binding sequences of transcription factors.	Section 3.1.8
log	.txt	Lists of entries that could not be mapped across files or were incomplete.	

5 Example Workflows

5.1 S. aureus Analysis with NGS Pipeline and File Conversion

Goal: Analysis of the potential impact of mutations on the antibiotic resistance of *Staphylococcus aureus* with reference strain NCTC 8325.

Procedure:

1. Obtain gene information of the reference strain (e.g. from AureoWiki) as a tab–separated (.tsv) file. Rename relevant columns as specified in Section 3.1.1.

2. Generate the mutations file.
   1. Obtain a genome .fasta file of the reference strain. Set the NGS pipeline **reference genome file** path accordingly.
   2. Obtain NGS reads as paired .fastq files of another strain and place them in a directory. Set the NGS pipeline **reads directory** accordingly.
   3. Set the NGS pipeline **result directory**.
   4. Click on NGS pipeline **Run**.

3. Generate the protein domain file using UniProt.
   1. Open the converter under **Tools → UniProt protein domain converter**.
   2. Obtain a UniProt protein domain text file of the reference strain as described in Section 4.2.1. Set the converter **UniProt input file** accordingly.
   3. Set the converter **result file** to an existing .tsv file to override it or create a new one in the file dialog.
   4. Click on **Run**.

4. Generate the antibiotic resistance file using PATRIC.
   1. Open the converter under **Tools → PATRIC antibiotic resistance converter**.
   2. Obtain a PATRIC antibiotic resistance file of the reference strain as described in Section 4.3.1. Set the converter **PATRIC input file** accordingly.
   3. Set the converter **result file** to an existing .tsv file to override it or create a new one in the file dialog.
   4. Click on **Run**.

5. Search the literature for regulation information and create a regulation .tsv file as specfied in Section 3.1.6.

6. Obtain transcription factor (TF) sequences and transcription factor binding site (TFBS) information for the reference strain (e.g. from RegPrecise). Use that information to create a TFBS .tsv file as specified in Section 3.1.7 and a TF multiple sequence alignment file as specified in Section 3.1.8.

7. Set the mutation analysis paths.
   1. Set the **gene file** to the file from step 1.
   2. Set the **mutation file** to the file in the NGS pipeline **result directory** from step 2.
   3. Enable **Genes of Interest Analysis** and set **genes of interest file** to the file from step 4.
   4. Enable **Protein Domain Analysis** and set the **protein domain file** to the file obtained in step 3.
   5. Enabled **Coding Region Analysis** and set the **substitution matrix file** to the **PAM10** file in the **substitution_matrices** directory.
   6. Enable **Regulation Analysis** and set the **regulation file** to the file created in step 5.

7. Enable **Transcription Factor Binding Site Analysis** and set the **TFBS file** and **TF MSA file** that were created in step 6.

8. Click on mutation analysis **Run**.

Results: All files listed in Section 3.2.

5.2 NGS Pipeline Only

Goal: Generate a set of mutations in .vcf and .tsv format based on paired–end NGS reads.

Procedure:

1. Obtain a genome .fasta file of the reference strain. Set the NGS pipeline **reference genome file** path accordingly.

2. Obtain NGS reads as paired .fastq files of another strain and place them in a directory. Set the NGS pipeline **reads directory** accordingly.

3. Set the NGS pipeline **result directory**.

4. Click on NGS pipeline **Run**.

Results: A .tsv file with all mutations ordered by genome position, as well as SNP and indel .vcf files for each paired read file used in the pipeline.

References

[1] H. Li and R. Durbin. "Fast and accurate long-read alignment with Burrows-Wheeler transform". In: *Bioinformatics* 26.5 (Mar. 2010), pp. 589–595. URL: http://bio-bwa.sourceforge.net/.

[2] H. Li et al. "The Sequence Alignment/Map format and SAMtools". In: *Bioinformatics* 25.16 (Aug. 2009), pp. 2078–2079. URL: http://samtools.sourceforge.net/.

[3] D. C. Koboldt et al. "VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing". In: *Genome Res.* 22.3 (Mar. 2012), pp. 568–576. URL: http://varscan.sourceforge.net.

[4] P. Shannon et al. "Cytoscape: a software environment for integrated models of biomolecular interaction networks". In: *Genome Res.* 13.11 (Nov. 2003), pp. 2498–2504.

[5] No authors listed. "UniProt: the universal protein knowledgebase". In: *Nucleic Acids Res.* 45.D1 (Jan. 2017), pp. D158–D169. URL: http://www.uniprot.org/.

[6] A. R. Wattam et al. "PATRIC, the bacterial bioinformatics database and analysis resource". In: *Nucleic Acids Res.* 42.Database issue (Jan. 2014), pp. D581–591. URL: https://www.patricbrc.org/.

[7] S. Gama-Castro et al. "RegulonDB version 9.0: high-level integration of gene regulation, coexpression, motif clustering and beyond". In: *Nucleic Acids Res.* 44.D1 (Jan. 2016), pp. D133–143. URL: http://regulondb.ccg.unam.mx/.