RSATWSPortType

Service Documentation Web services for the Regulatory Sequence Analysis Tools (RSAT). Tools developed by Jacques van Helden (jvanheld@scmbb.ulb.ac.be), SOAP/WSDL interface developed by Olivier Sand (oly@scmbb.ulb.ac.be).

Server Address http://rsat.scmbb.ulb.ac.be/rsat/web_services/RSATWS.cgi

Method retrieve_seq

Description Returns upstream, downstream or coding DNA sequences for list of query genes.

Parameters

Input Parameters

output
Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'.
Type = String

organism (required)
Organism. Words need to be underscore separated (example: Escherichia_coli_K12).
Type = String

query
A list of query genes.
Type = String

all
Return sequences for all the genes of the organism if value = 1. Incompatible with query.
Type = Integer

noorf
Prevent overlap with upstream open reading frames (ORF) if value = 1.
Type = Integer

from
Inferior limit of the region to retrieve. Default is organism dependant (example: Saccharomyces cerevisiae = -800).
Type = Integer

to
Superior limit of the region to retrieve. Default is '-1'.
Type = Integer

feattype
Type of genome features to load. Supported: CDS, mRNA, tRNA, rRNA.
Type = String

type
Sequence type. Supported: upstream, downstream, ORF (unspliced open reading frame).
Type = String

format
Sequence format. Supported: IG (Intelligenetics), WC (wconsensus), raw, FastA
Type = String

lw
Line width (0 for whole sequence on one line).
Type = Integer

label
Field(s) to be used in the sequence label. Multiple fields can be specified, separated by commas. Supported: id, name, organism_name, sequence_type, current_from, current_to, ctg, orf_strand, reg_left, reg_right. Default: name.
Type = String

label_sep
Separator between the label fields. Default: | (pipe character).
Type = String

nocom
No comments if value = 1. Only the identifier and the sequence are returned. By default, the comment indicates the ORF and upstream sequence coordinates.
Type = Integer

repeat
Use the repeat masked version of the genome if value = 1. Attention: repeated regions are annotated for some genomes only.
Type = Integer

imp_pos
Admit imprecise positions if value = 1.
Type = Integer

Output Parameters

server
Location of the result file on the server. This can be used as input for a further request.
Type = String

command
The stand alone command executed on the server.
Type = String

client
The results.
Type = String

Method purge_seq

Description Mask repeated fragments of an input sequence.

Parameters

Input Parameters

output
Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'.
Type = String

sequence
Sequence to purge. You need to supply either this parameter or the next one (tmp_infile).
Type = String

tmp_infile
Name of the file with input sequence on the server. You need to supply either this parameter or the previous one (sequence).
Type = String

format Sequence format. Supported: IG (Intelligenetics), WC (wconsensus), raw, FastA.
Type = String

match_length Minimal match length. Default is 40.
Type = Integer

mismatch Number of mismatches allowed. Default is 3.
Type = Integer

str Discard duplications on the direct strand only (1) or on the reverse complement as well (2). Default is 2.
Type = Integer

delete Delete repeats instead of masking them if value = 1.
Type = Integer

mask_short Mask (replace by N characters) sequences shorter than the specified length.
Type = Integer

Output Parameters

server Location of the result file on the server. This can be used as input for a further request.
Type = String

command The stand alone command executed on the server.
Type = String

client The results.
Type = String

Method oligo_analysis

Description Analysis of the statistical significance of all the oligomers of a given size in a sequence. Commonly used to detect over-represented oligonucleotides in a set of promoter sequences.

Parameters

Input Parameters

output Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'.
Type = String

sequence Input sequence. You need to supply either this parameter or the next one (tmp_infile).
Type = String

tmp_infile Name of the file with input sequence on the server. You need to supply either this parameter or the previous one (sequence).
Type = String

format Input sequence format. Supported: IG (Intelligenetics), WC (wconsensus), raw, fasta. Default is fasta.
Type = String

length (required) Oligomer length.
Type = Integer

organism (required) Organism. Words need to be underscore separated (example: Escherichia_coli_K12).
Type = String

background Background model: Type of sequences used as background model for estimating expected oligonucleotide frequencies. Supported: upstream, upstreamL, upstream-noorf, intergenic, input.
Type = String

stats List of statistics to return. Supported:occ, mseq, freq, proba, ratio, zscore, like, pos, rank.
Type = String

noov No overlapping of oligos allowed if value = 1. Disable the detection of overlapping matches for self-overlapping patterns (ex TATATA, GATAGA).
Type = Integer

str Oligonucleotide occurrences found on both stands are summed (2) or not (1). Default is 2.
Type = Integer

sort Sort oligomers according to overrepresentation if value = 1.
Type = Integer

lth Lower threshold on some parameter. Format='param value'.
Type = String

Output Parameters

server Location of the result file on the server. This can be used as input for a further request.
Type = String

command The stand alone command executed on the server.
Type = String

client The results.
Type = String

Method dyad_analysis

Description Analysis of the statistical significance of all the spaced dyads of a given size in a sequence. Commonly used to detect over-represented spaced dyads in a set of promoter sequences.

Parameters

Input Parameters

output Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'.
Type = String

sequence Input sequence. You need to supply either this parameter or the next one (tmp_infile).
Type = String

tmp_infile Name of the file with input sequence on the server. You need to supply either this parameter or the previous one (sequence).
Type = String

format Input sequence format. Supported: IG (Intelligenetics), WC (wconsensus), raw, fasta. Default is fasta.
Type = String

length (required) Dyad length.
Type = Integer

spacing (required) Spacing between elements of the dyads.
Type = String

organism (required) Organism. Words need to be underscore separated (example: Escherichia_coli_K12).
Type = String

background Background model: Type of sequences used as background model for estimating expected oligonucleotide frequencies. Supported: upstream, upstreamL, upstream-noorf, intergenic, input.
Type = String

stats List of statistics to return. Supported:occ, mseq, freq, proba, ratio, zscore, like, pos, rank.
Type = String

type dyad_type (dr | ir | rep | any) In order to fasten execution, the program can be asked to restrict its analysis to symmetric dyads. Four types are accepted: dr - direct repeats: the second element is the same as the first one; ir - inverted repeats: the second element is the reverse complement of the first one; rep - repeats: direct and inverted repeats are evaluated. any - (default) When selecting the option any, the analysis is performed on all non-symmetric dyads as well.
Type = String

noov No overlapping of dyads allowed if value = 1. Disable the detection of overlapping matches for self-overlapping patterns (ex TATATA, GATAGA).
Type = Integer

str Dyad occurrences found on both stands are summed (2) or not (1). Default is 2.
Type = Integer

sort Sort dyads according to overrepresentation if value = 1.
Type = Integer

under Detect under-represented instead of over-represented dyads (left tail significance test) if value = 1.
Type = Integer

two_tails Detect under-represented and over-represented dyads (two-fail significance test) if value = 1.
Type = Integer

zeroocc Report also dyads with zero occurrences (provided they fit the other thresholds) if value = 1. By default, the program reports only patterns present in the sequence. If the left tail or two-tail test is applied, patterns with zero occurrences are automatically taken into account. In some other cases, one would also like to detect patterns absent from the sequence. This is the function of the option -zeroocc.
Type = Integer

lth Lower threshold on some parameter. Format='param value'.
Type = String

uth Upper threshold on some parameter. Format='param value'.
Type = String

Output Parameters

server Location of the result file on the server. This can be used as input for a further request.
Type = String

command The stand alone command executed on the server.
Type = String

client The results.
Type = String

Method dna_pattern

Description Searches all occurrences of a pattern within DNA sequences.

Parameters

Input Parameters

output Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'.
Type = String

sequence Input sequence. You need to supply either this parameter or the next one (tmp_infile).
Type = String

tmp_infile Name of the file with input sequence on the server. You need to supply either this parameter or the previous one (sequence).
Type = String

format
Input sequence format. Supported: IG (Intelligenetics), WC (wconsensus), raw, fasta. Default is fasta.
Type = String

subst
Number of substitutions allowed.
Type = Integer

pattern (required)
Pattern to match.
Type = String

id
Pattern identifier.
Type = String

origin
Origin for the calculation of positions (0 for end of sequence).
Type = String

noov
No overlapping of oligos allowed if value = 1. Disable the detection of overlapping matches for self-overlapping patterns (ex TATATA, GATAGA).
Type = Integer

str
Oligonucleotide occurrences found on both stands are summed (2) or not (1). Default is 2.
Type = Integer

sort
Sort oligomers according to overrepresentation if value = 1.
Type = Integer

th
Threshold on match count.
Type = Integer

Output Parameters

server
Location of the result file on the server. This can be used as input for a further request.
Type = String

command
The stand alone command executed on the server.
Type = String

client
The results.
Type = String

Method gene_info

Description Get information about genes.

Parameters

Input Parameters

output
Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'.
Type = String

organism (required)
Organism. Words need to be underscore separated (example: Escherichia_coli_K12).
Type = String

query (required)
List of gene(s) for which you want info on or list of keywords to search for (can be regular expressions).
Type = String

full
Full match only (no substring matching) if value = 1.
Type = Integer

noquery
Do not print the query at the begining of each line if value = 1.
Type = Integer

descr
Match query against the description, too, not just against gene ID and name if value = 1.
Type = Integer

feattype
Feature type (CDS, mRNA, tRNA, rRNA, scRNA).
Type = String

Output Parameters

server
Location of the result file on the server. This can be used as input for a further request.
Type = String

command
The stand alone command executed on the server.
Type = String

client
The results.
Type = String

Method supported_organisms

Description Get a list of supported organisms.

Parameters

Input Parameters

output
Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'.
Type = String

format
Output format (supported: html_list,html_table,array,text,keys,names,sizes,full,tree,html_tree).
Type = String

taxon
Root taxon.
Type = String

Output Parameters

server
Location of the result file on the server. This can be used as input for a further request.
Type = String

command
The stand alone command executed on the server.
Type = String

client
The results.
Type = String

Method convert_seq

Description Converts a sequence between two formats (e.g. fasta -> raw).

Parameters Input Parameters

output
Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'.
Type = String

sequence
Sequence to convert.
Type = String

tmp_infile
Name of the file with input sequence on the server. You need to supply either this parameter or the previous one (sequence).
Type = String

from (required)
Format of input sequence (embl, fasta, filelist, gcg, genbank, ig, maf, multi, ncbi, raw, tab, wc, wconsensus).
Type = String

to (required)
Format of output sequence (fasta, filelist, ft, ig, multi, raw, tab, wc, wconsensus).
Type = String

Output Parameters

server
Location of the result file on the server. This can be used as input for a further request.
Type = String

command
The stand alone command executed on the server.
Type = String

client
The results.
Type = String

Method compare_classes

Description Compare two class files(the query file and the reference file).Each class of the query file is compared to each class of the reference file.The number of common elements is reperted, as well as the probability to observe at least this number of common elements by chance alone.

Parameters Input Parameters

output
Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'.
Type = String

ref_classes
A tab-delimited text file containing the description of reference classes
Type = String

query_classes
A tab-delimited text file containing the description of query classes
Type = String

return_fields
List of fields to return. Supported field: dotprod,entropy,freq,jac_sim,members,occ,proba,rank
Type = Integer

score_column
Specify a column of the input file containing a score associated to each member
Type = Integer

input_classes
This file will be used as both reference and query.This is equivalent to -q input_file -r input_file.
Type = String

upper_threshold_field
Supported fields : E(QR),E_val,F(!Q!R),F(Q!R),F(Q),F(QR),F(R!Q),F(R),H(Q) ,H(Q,R),H(Q|R),H(R),H(R|Q),I(Q,R),IC,P(QR),P(Q|R),P(R|Q),P_val,Q,QR,QvR,R,U(Q|R) ,U(R|Q),dH(Q,R),dotprod,jac_sim,rank,sig
Type = String

upper_threshold_value
upper threshold value for a supported field
Type = Float

lower_threshold_field
Supported fields: same fields as upper_threshold_field.
Type = String

lower_threshold_value
lower threshold value for a given fields.
Type = Float

sort
sort on the basis of the specified key.
Type = String

distinct
Prevent to compare each class with itself (when the reference and query files contain the same classes)
Type = Integer

triangle
(only valid if query file and reference file are the same) Do not perform the reciprocal comparisons.
Type = Integer

matrix
Return a pairwise matrix, where each row corresponds to a reference class, each column to a query class, and each cell contains a comparison between the two classes. The next argument indicates which statistics has to be return in the matrix (default=sig)Supported: E(QR),E_val,F(!Q!R),F(Q!R),F(Q),F(QR),F(R!Q),F(R),H(Q),H(Q,R),H(Q|R),H(R),H(R|Q),I(Q,R),IC,P(QR),P(Q|R),P(R|Q),P_val,Q,QR,QvR,R,U(Q|R),U(R|Q),dH(Q,R),dotprod,jac_sim,rank,sig
Type = String

Output Parameters

server
Location of the result file on the server. This can be used as input for a further request.
Type = String

command
The stand alone command executed on the server.
Type = String

client
The results.
Type = String

Method matrix_scan

Description
Scan sequences with one or several position-specific scoring matrices (PSSM) to identify instances of the corresponding motifs(putative sites). This program supports a variety of background models (Bernoulli, Markov chains of any order).

Parameters

Input Parameters

output
Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'.
Type = String

sequence_file
Sequence file - all the formats supported in RSAT can be used as input (default: fasta)
Type = String

matrix_file
The matrix file is specified with the option "matrix_format" (see below) Default format: tab .
Type = String

matrix_format
Supported field: tab, cb, consensus, gibbs, meme, assembly.
Type = String

matrix_list
Indicate a file containing a list of matrices to be used for scanning the region. This facilitates the scanning of a sequence with a library of matrices (e.g. all the matrices from RegulonDB or TRANSFAC) Format: the matrix list file is a text file. The first word of each row is suppose to indicate a file name. Any further information on the same row is ignored.
Type = String

top_matrices
Only scan with the top # matrices per matrix file. This option is valid for some file formats containing multiple matrices where top matrices are generally more informative.
Type = Integer

background
Background model file is a tab-delimited file containing the specification of oligonucleootide frequencies.
Type = String

background_input
Calculate background model from the input sequence set. This option requires to specify the order of the background model with the option markov .
Type = Integer

background_window
Size of the sliding window for the background model calculation.This option requires to specify the order of the background model with the option markov (suitable for short order model only markov 0 or 1)
Type = Integer

markov
Order of the markov chain for the background model. This option is incompatible with the option background .
Type = Integer

background_pseudo
Pseudo frequency for the background models. Value must be a real between 0 and 1. If this option is not specified, the pseudo-frequency value depends on the background calculation. For -bginput and -window, the pseudo frequency is automatically calculated with the length (L) of the sequence following this formula : square-root of L divided by L+squareroot of L. For -bgfile, default value is 0.01. If the training sequence length (L) is known, the value can be set by -bg_pseudo option to square-root of L divided by L+squareroot of L.
Type = Float

return_fields
List of fields to return. Supported fields: sites, rank, limits, normw, bg_model, matrix, freq_matrix, weight_matrix, distrib .
Type = String

upper_threshold_field
Type = String

upper_threshold_value
Type = Float

lower_threshold_field
Type = String

lower_threshold_value
Type = Float

both_strand
Type = Integer

single_strand
Type = Integer

Output Parameters

server
Location of the result file on the server. This can be used as input for a further request.
Type = String

command
The stand alone command executed on the server.
Type = String

client
The results.
Type = String

Method **random_seq**

Description
Generates random sequences.

Parameters

Input Parameters

output
Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'.
Type = String

sequence_length
Length of sequence to generate.
Type = Integer

repetition
Number of sequences to generate.
Type = Integer

format
Format of sequence(s) to generate.
Type = String

line_width
A newline character will be inserted in the sequence every # bases, where # is the number provided. Default is 70. A value of 0 will prevent newline insertion.
Type = Integer

type
Type of sequence(s) to generate (protein | DNA | other).
Type = String

seed
Seed for the random generator.
Type = Integer

alphabet
alphabet must be followed by residue frequencies expressed precisely this way: -a a:t # c:g #
Type = String

expfreq
Expected frequencies of oligomers in sequence(s) to generate. Indicate the file that contains expected oligomer frequencies. When this option is used, the sequences are generated according to a Markov chain.
Type = String

tmp_expfreq_file
Name of the file with expected frequencies on the server.
Type = String

bg_model
Background model. Automatically load a pre-calibrated exected frequency file from the RSAT genome distribution. When this option is used, the options organism and oligo_length are also required, to indicate the organism and the oligonucleotide length, respectively. This option is incompatible with the option expfreq. Type of sequences used as background model for estimating expected oligonucleotide frequencies (supported models): equi (equiprobable residue frequencies [default]), upstream (all upstream sequences, allowing overlap with upstream ORFs. Requires to speciy a model organism), upstream-noorf (all upstream sequences, preventing overlap with upstream ORFs. Requires to specify a model organism), and intergenic (intergenic frequencies. Whole set of intergenic regions, including upstream and downstream sequences. Requires to specify a model organism).
Type = String

organism
Name of the organism when using a background model.
Type = String

oligo_length
Length of oligomer when using a background model.
Type = Integer

length_file
length file Allows to generate random sequences with the same lengths as a set of reference sequences. The length file contains two columns : sequence ID (ignored) and sequence length.
Type = String

Output Parameters

server
Location of the result file on the server. This can be used as input for a further request.
Type = String

command
The stand alone command executed on the server.
Type = String

client
The results.
Type = String