WOLAND Documentation

Release 0.1.1

Tiago de Souza, Alexandre Defelicibus, Carlos Menck

Oct 06, 2017

Contents

WOLAND is a tool to analyze point mutation patterns using resequencing data from any organism or cell. There are many suitable applications for WOLAND such as the study of the genome-wide impact of both endogenous and exogenous mutagens on organisms and cells and analysis of mutagenic profiles of DNA repair-associated diseases. Also, mechanisms of molecular mutational processes involving specific target proteins and identification of potential hazardous mutagens in environmental samples can be profiled using WOLAND.

WOLAND retrieves a comprehensive visual report with R-based graphics to enable fast and reliable interpretation by the user.

Given one or more list of SNVs in ANNOVAR variant_function format, WOLAND can perform:

- **Nucleotide type changes identification and context-sequence extration**: identify frequency of nucleotide type changes and extract context-sequences around each point mutation.
- **Search for mutagen-associated motifs**: Retrieve the number, frequency and localization of mutagen-associated motif sequences such as UV-light, 8-oxoguanine, 6-4 photoproducts, ENU, among others.
- **Search for mutational hotspots**: Identify mutational hotspots using a user-defined sliding window which considers each SNV as the window center.
- **Transcriptional strand bias**: Retrieve scores associated with the strand of each mutational motif found using RefSeq annotation data.

You can use Galaxy-WOLAND directly through your web navigator or install a local version. Galaxy-WOLAND is a friendly and easy interface while installing a local version will allow you to customize all analyses steps. Please see docs for further information!

The documentation of WOLAND is organized into:

CHAPTER 1

README

WOLAND is a multiplatform tool to analyze point mutation patterns using resequencing data from any organism or cell. It is implemented as a Perl and R tool using as inputs filtered unannotated or annotated SNV lists, combined with its correspondent genome sequences.

What do you need to use woland:

> annotated SNV list(s) genome sequence and annotation length of each chromosome or a bed file from a targeted-resequencing experiment (exome, for example)

Features

WOLAND can provide:

- the number and frequency of nucleotide type changes
- detection of regions enriched in mutations alongside the genome (hotspots)
- extraction of sequence-context sequences of each SNV.
- count established mutational motifs associated with environmental mutagens and DNA-repair mechanisms
- calculation of transcriptional strand bias of mutations linked to the mutational motifs found.

Contribute

- Issue Tracker: github.com/woland
- Source Code: github.com/woland

Support

If you are having issues, please let us know. We have a mailing list located at: woland@google-groups.com

License

CHAPTER 2

Install

WOLAND is a multiplatform tool based on Perl and R. Please observe prerequisites and modules and libraries need. They must be installed before WOLAND installation.

Prerequisites

The following software must be present before installing Woland:

Perl (Minimal recommended Perl version: 5.17)

To check Perl version type:

```
$ perl -v
```

The following Perl modules should be present:

- Bio::DB::Fasta
- Cwd
- List::Util
- IPC::System::Simple
- IPC::Run
- Parallel::ForkManager
- Regexp::Common
- Text::Balanced(>=1.97)
- Text::Wrap
- Statistics::R

R (Minimal recommended R version: 3.1)

To check R version type (in R):

```
$ R.Version()
```

The following R packages must be installed:

- plyr
- reshape2
- ggplot2
- qqman
- RColorBrewer

Installation

Installing Perl modules

You can use CPAN to get any missing module. First install cpanm:

```
$ sudo cpan App::cpanminus
```

Them you can install each module:

```
$ sudo cpanm Module::Name
```

Installing R packages

You can use this following command in R to install each missing library:

$R install.packages(`packagename`)

Installing source WOLAND files

Download last Woland source release in <link>. Woland is provided as a tar.gz file which could be extracted using, for example:

```
$ tar -xvzf woland-<version>-install.tar.gz
```

Woland will be installed inside woland installation folder `install_dir`.

Copying genome sequences and annotation

Woland needs genome reference sequence and its gene annotation for each organism in the `$install_dir/genomes` folder. User should download two files in order to perform its analysis:

Note: These files can be downloaded using UCSC database (http://hgdownload.cse.ucsc.edu/downloads.html). See below how to rename them:

```
$install_dir/genomes/genome_<genome_version>.fa
$install_dir/genomes/refseq_<genome_version>.txt
```

Warning: You must rename `<genome_version>.fa` and `refGene.txt` according to `<genome_version>`.

For example:

- hg19:

```
$install_dir/genomes/genome_hg19.fa

$install_dir/genomes/refseq_hg19.fa
```

- mm10:

```
$install_dir/genomes/genome_mm10.fa

$install_dir/genomes/refseq_mm10.fa
```

CHAPTER 3

Using

WOLAND has three main scripts: `woland-anno.pl`, `woland-report.pl` and `woland-batch.pl`) plus a acessory script `woland-bed.pl`. Please observe descriptions of each script, input file file requirements, usage and outputs.

Scripts

woland-batch.pl

Script which automatically runs multiple instances of woland-anno.pl and build a single report using woland-report.pl:

```
$ perl woland-batch.pl <input_table> <chr_profile> <hs_window> <genome_version>
```

woland-anno.pl

Script used to calculate mutational patterns of a single annotated variant file. Uses a single `<variant_function>` to calculate all patterns in a single result folder for `<results-variant_function>` provided:

```
$ perl woland-anno.pl <variant_function> <chr_profile> <hs_window> <genome_version>
```

woland-report.pl

Script used to build a report of multiple annotated variant files assigned as groups with results already done by woland-anno.pl:

```
$ perl woland-report.pl <input_table>
```

Inputs

<input.table>

Regular tabular file without header. First column is group name. Second column is file sample name of annovar annotated.variant file. Samples files MUST be located in the Woland install folder.

<annovar.variant_function>

Annotated .variant_function file which can be obtained using annotate-variation.pl script from ANNOVAR.

<chromosome_profile>

Regular tabular file without header . First column is chromosome name in chr format (e.g. chr13). Second column is chromosome length sequenced. User can build this file with target .BED file using `woland-bed-to-profile.pl`.

<hotspot_window>

A natural number N (N>1), for hotspot window length. Hotspot window corresponds to N nucleotides flanking each SNV.

<genome_version>

Genome version of `genomes/genome_<genome_version>` and `genomes/refseq_<genome_version>` files.

<coordinates.bed>

Coordinates of target regions used in sequencing experiment in BED format.

Usage

woland-batch.pl

woland.batch enables batch submission of multiple samples as provided by <input.table> file. This script runs Woland-anno.pl for each sample followed by Woland-report.pl generating one result folder for each sample file provided and a grouped report folder for whole analysis as provided by `<input.table>`

```
$ perl woland-batch.pl <input.table> <chromosome_profile> <hotspot_window> <genome_
↪version>
```

woland-anno.pl

Uses a single <annovar.variant_function> to calculate all patterns in a single result folder for <annovar.variant_function> provided:

```
$ perl woland-anno.pl <annovar.variant_function> <chromosome_profile> <hotspot_window>
↪ <genome_version>
```

woland-report.pl

This script uses a group of samples to perform Woland-anno.pl script for each sample provided to build an unique grouped report output folder for the analysis. Sample names and groups are provided by <input.table> and one result folder is also generated for each sample provided:

```
$ perl woland-report.pl <input.table> <chromosome_profile> <hotspot_window> <genome_
↪version>
```

woland-bed.pl

This script generates a <chromosome_profile> file using a .bed file from a targeted-sequencing experiment, for example. The <chromosome_profile> file could be used in other Woland scripts:

```
$ perl woland-bed.pl <coordinates.bed>
```

CHAPTER 4

Tutorial

The easiest way to perform a WOLAND analysis is through a single batch submission using `woland-batch.pl`. This will envolve initial 4-step preparation but in the next-time that you will use WOLAND with other samples (and we believe that you do!) you will use only the first step. It is easy no? Each step will prepare the inputs for this script:

```
$ perl woland-batch.pl -i <input.table file> -c <chromosome.profile file> -g <genomes.
↪folder> -n <genome.version> -r <refseq.file> -w <hotspot.window length> -t <number.
↪of.threads> -o <target.output folder>
```

First Step - Preparing input-table

Filtering

In most cases, a raw .vcf file containing SNVs from a resequencing pipeline is not suitable for a point mutation analysis. First, you have to filter polymorphisms and false-positives from each sample using, for example, `vcftools` (<http://vcftools.sourceforge.net/>) and/or `ANNOVAR` (<http://annovar.openbioinformatics.org/en/>).

Annotating

Several tools are available to annotate .vcf files. However, WOLAND accepts only `ANNOVAR` (<http://annovar.openbioinformatics.org/en/>) gene annotation. It is easy to use ANNOVAR and you can find information about downloading, installing and using it at its website. Here is an example how to use annovar to annotate a .vcf file using `annotate_variation.pl` from ANNOVAR:

```
$ perl annotate_variation.pl -geneanno -buildver hg19 example/ex1.avinput humandb/
```

Warning: WOLAND accepts ONLY `.variant_function` files from ANNOVAR. It is not possible to use `exonic_variant_function` output.

At this time you have a `.variant_function` for each sample to be analyzed. You can manually annotate a file (think twice before) or force annotation when gene information are not avaialble (or not necessary). Let's take a look at a `variant-function` file from annovar:

exonic	Lrp1b	chr2	3432131	3432131	A	G
intergenic	Rbpj	chr5	25465	25465	T	A
intronic	Cmklr1	chr5	4234231	4234231	C	T
intronic	Setd8	chr5	8423415	8423415	G	C
...	...	...	...	...	.	.

Now you have to build a tabular `input-table` file to assign samples into a group name - a "Control" or a "Treated" group, for example.

Grouping samples

At this step you must create a simple tabular file (`input-table`). Each line must corresponds to each file sample name in the first column and its group in the second column. Let's see an example:

Control	Sample1.txt.variant_function
Control	Sample2.txt.variant_function
Treated	Sample3.txt.variant_function
Treated	Sample4.txt.variant_function
Treated	Sample5.txt.variant_function

This file `input-table` must be saved as a tabular text file and it will be used as the first argument in `woland-batch.pl` script.

Note: You can provide a path for each file in `input-table` if it does not rely on WOLAND `$install_dir`.

Second Step - Chromosome profile

At this step you must check you chromosome profile file. This file contains the length of each chromosome of your genome and it is used to calculate frequency of mutational changes. You can manually create your own chromosome profile or use the `woland-bed.pl` script if you have a .BED file from you targeted resequencing experiment (exome, for example). Let's see an example of a `chr_profile` file:

chr1	195471971
chr2	182113224
chr3	160039680
chr4	156508116
...	...

Note: If you have a .BED file from you experiment you can use `woland-bed.pl`. For example:

```
$ perl woland-bed.pl hg19-exome-enrichment.bed
```

This will create a `WOLAND-BED-PROFILE-hg19-exome-enrichment.bed` file which can be used as `chr_profile` argument.

Third Step - Genome information

WOLAND uses genome sequences in FASTA format to extract context sequences and RefSeq annotation to obtain gene and transcriptional information. So you must provide two files for each genome. It is easy to obtain them and you MUST rename the according to <genome_version> parameter and move them to `$install_dir/genomes/` folder.

A lot of genome sequences are avaialble nowadays. We advise you to use UCSC genome database to obtain your genome sequence and your RefSeq annotation file. Please check http://hgdownload.cse.ucsc.edu/downloads.html.

Genome sequence in FASTA format

The genome sequence must contain all chromosomes in `chr` format. For example:

```
>chr1
AGCATCGATCGGCATGCATGCTAGCTAGCTACGATGCTAGCAT (...)
>chr2
GCATGCATCGTACGTACGATCGATCGATCGATCGATCGATCGA (...)
(...)
```

Please rename the FASTA file to `genome_<genome_version>.fa` and move it to `$install_dir/genomes/`. For example:

```
$ mv hg19.fa $install_dir/genomes/hg19.fa
```

RefSeq annotation

The RefSeq annotation can be obtained through http://hgdownload.cse.ucsc.edu/downloads.html .

Note: You MUST download the RefGene file - usually provided as `refGene.txt`.

Please rename the RefGene file to `refseq_<genome_version>.txt` and move it to `$install_dir/genomes/`. For example:

```
$ mv RefGene $install_dir/genomes/refseq_hg19.txt
```

Fourth Step - Choosing hotspot window length and running!

Now you can choose a natural number >1 for the hotspot window length `<hotspot_window>`, for example: 1000. Now, voilà, you can run `woland-batch.pl`!:

```
$ perl woland-batch.pl -i input.table.tgca.csv -c profiles/chromosome.profile.hg19.
↪bed.exons.txt -w 1000 -g genomes/ -n hg19 -r genomes/refseq_hg19.txt -o .
```

CHAPTER 5

Outputs

WOLAND provide different set of outputs depending on the script used - and how far you want to go in your mutational pattern analysis. `woland-batch.pl` runs the most complete analysis but you must have at least two groups of samples (a control and a treated group, for example).

Note: `woland-batch.pl` provides the faster and straighforward way to analyze samples. It runs `woland-anno.pl` for each sample then runs `woland-report.pl`.

woland-anno.pl basic outputs

This script will analyze each ANNOVAR `variant_function` files and will provide a total of 13 tabular text files + 1 log file. We consider these output files as the most raw type of WOLAND analysis. Let's take a look at each class and its output files explanations:

Note: All `woland-anno.pl` outputs were saved in a `results-$sample_name/` folder.

Nucleotide type-changes and frequency

`WOLAND-basechange-$sample_name`:

`WOLAND-mutfreq-$sample_name`:

Extracted context sequences

`WOLAND-contextsequences-$sample_name`:

`WOLAND-contextsequencesanno-$sample_name`:

Hotspots

`WOLAND-hotspots-$sample_name`:

Mutational motifs

`WOLAND-motifs-$sample_name`:

`WOLAND-norm_motifs-$sample_name`:

Transcriptional strand bias

`WOLAND-bias_motif-$sample_name`:

CHAPTER 6

Report

WOLAND provide different set of outputs depending on the script used - and how far you want to go in your mutational pattern analysis. `woland-batch.pl` runs the most complete analysis but you must have at least two groups of samples (a control and a treated group, for example).

Note: `woland-batch.pl` provides the faster and straighforward way to analyze samples. It runs `woland-anno.pl` for each sample then runs `woland-report.pl` to build the report.

Graphical report

`woland-report.pl` uses outputs from `woland-anno.pl` to build a comprehensive grouped analysis with some R graphical data. Those outputs are saved as graphical data as .SVG and .PDF files and a tabular text file (`.tmp`) for each graphic;

Note: All `woland-report.pl` outputs were saved in the folder `report-$input_table/`. You can use `.tmp` files to build your own graphic or use a statistical approach to test your hypothesis.

Nucleotide type-changes

- Point mutation frequency across chromosomes:

This barplot shows mutation frequency of each group in all chromosomes with standard deviation as error bars. It also shows an average line point mutation frequency for each group.

Warning: Woland uses the <chromosome_profile> file to calculate mutational frequency. If <chromosome_profile> was build using a targeted-enriched .bed file only those regions were used in the frequency calculation, for example.

Tabular text file:mutfreq-$input_table.tmp

- Number of nucleotide type changes:

This box-plot graph show the absolute number of each nucleotide type change of grouped samples.

Tabular text file:nucleotide_type_change-$input_table.tmp

- Frequency of nucleotide type changes:

This box-plot graphic shows the frequency of each nucleotide type change of the grouped samples. Y-label "value" means the number of each nucleotide type change in each sample divided by the total number of point mutations detected in each sample.

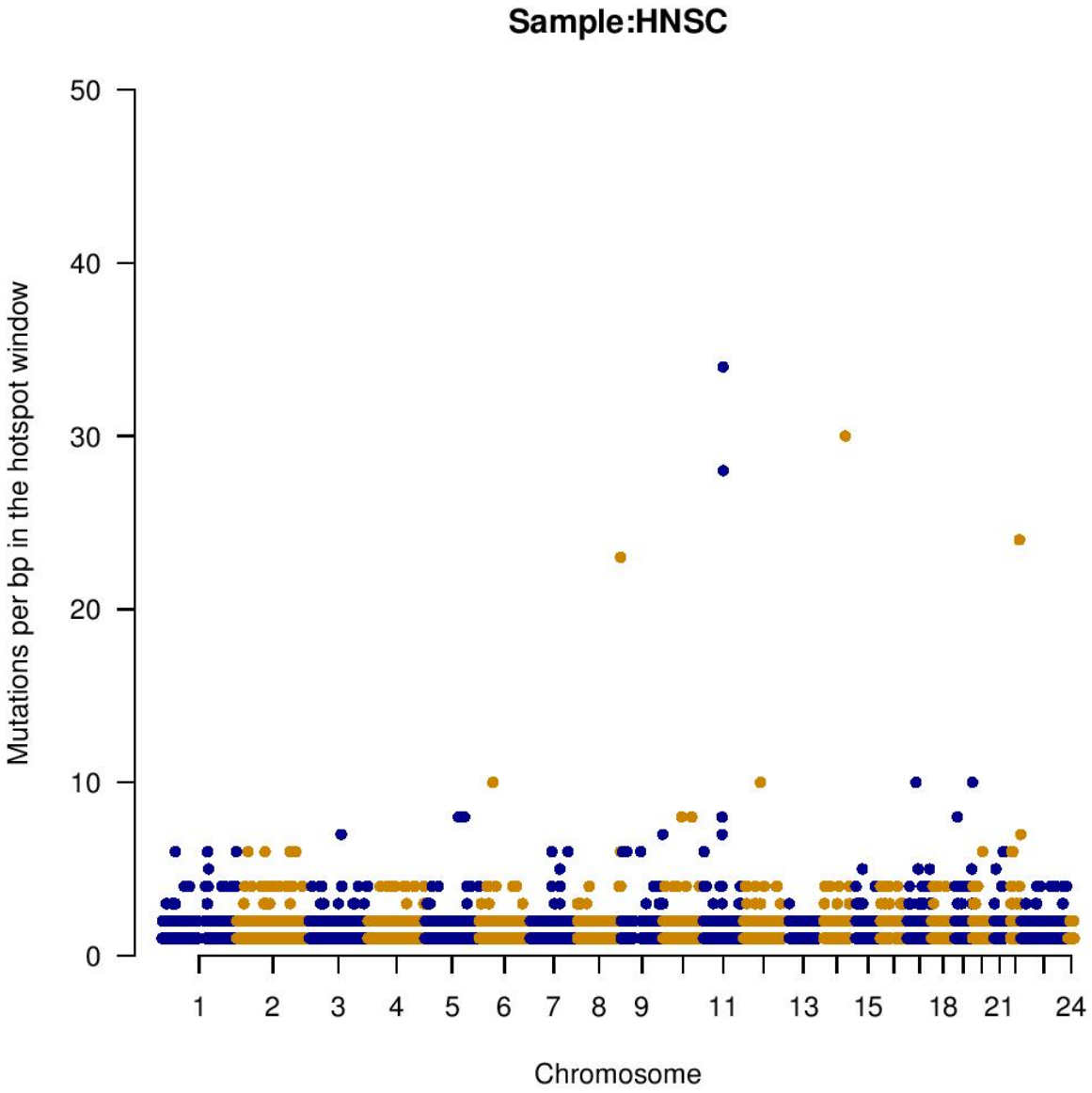
Tabular text file:nucleotide_type_changeF-$input_table.tmp

- Transversion & transition rate:

This stacked barplot graphic shows the frequency of transversion & transition rate of each sample analyzed grouped by name.

Tabular text file: transitiontransversionF-$input_table.tmp

Hotspots



A manhattan plot-like graphic is provided for each group analyzed. Y-axis means the number of mutation in the `<hotspot_window>` parameter provided in `<woland-batch.pl>` or `<woland-anno.pl>`. Gene names must be checked in the tabular text file `hotspots-$group.tmp`.

Tabular text file: `hotspots-$group.tmp`:

Mutational motifs

- Number of motifs associated with mutagens:

Box-plot graph showing the absolute number of each mutational motif sequence analyzed.

Tabular text file: `motif_number-$input_table.tmp`

- Number of normalized motifs associated with mutagens:

Box-plot graphic of each mutational motif sequence found normalized by the total number of point mutations of each sample file grouped by name. A value of 1.00 means that all point mutations contains the mutational motif.

Tabular text file: `motif_numberNorm-$input_table.tmp`

Transcriptional strand bias

Box-plot graphic of strand score (SC) concordance/discordance ratio for each mutational motif. Values = 1 means that there is no transcription associated strand bias. Values > 1 means more motifs in the same transcribed strand and values <1 means more motifs not in transcribed strand. Only SC scores equal to -1, 0, 1 are considered.

Tabular text file: `SC_concordance_ratio-$input_table.tmp`:

CHAPTER 7

FAQ

Is WOLAND free?

WOLAND is freely-avaialble to non-profit use, including research and academic users, under a GNU license. For commercial use please contact author at <tiagoantonio@gmail.com>.

Where can I get help?

If you are having issues, please let us know. We have a mailing list located at: <woland@google-groups.com>.

CHAPTER 8

Galaxy

WOLAND is implemented in a web-friendly Galaxy environment at <link>. This implementation covers only batch submissions (more than two sample groups) and a limited number of genomes (hg19, hg18, mm10, mm9). If you are interested to use WOLAND in a custom-user way please consider to install it locally.

Using WOLAND @ Galaxy

You must access our Galaxy server at <link>. You can enter WOLAND page clicking in the left-side panel. Let's take a look at WOLAND main page:

<image>

Adding ANNOVAR .variant_function files

If you already have ANNOVAR .variant_function files you must submit them to Galaxy. Thus, you must assign each file submitted to a group name.

Adding chromosome profile

You must submit a chromosome profile file containing the total length of each chromosome to be considered in frequency calculations.

Selecting genome

Please select the genome among organisms available. If you did not find a genome for your organism of interest please consider to install a local version of WOLAND.

Choosing hotspot window length

Now you must enter a natural number > 1 which will correspond to the radius of the hotspot window length.

Starting WOLAND

Click EXECUTE and wait analysis to finish

WOLAND web-report

After finishing you may able to access a tar.gz result file containing all woland outputs. In addition you can view and save a small web-report containing some report graphs.

CHAPTER 9

Contact

Support

If you are having issues, please let us know. We have a mailing list located at: woland@google-groups.com

FAQ

Please access our FAQ at.

EMAIL

To obtain a commercial license, contact author at the address below:

Tiago A. de Souza tiagoantonio@gmail.com skype::tiagoantonio github::tiagoantonio

CHAPTER 10

References