CleaveLand4 - TUTORIAL
November 7, 2013
Michael J. Axtell
mja18@psu.edu

Introduction
This is a brief tutorial to describe the usage of CleaveLand4. CleaveLand4 is a tool designed to parse 'degradome' data. Degradome data are a variant type of RNA-seq data, where the reads derive from the 5'-ends of uncapped RNAs (Addo-Quaye et al. 2008; German et al. 2008; Gregory et al. 2008). These data can be used to identify miRNA and siRNA targets that are actively 'sliced'. CleaveLand4 handles several phases of degradome data analysis in a single command, including:

- Alignment of degradome data to the reference transcriptome, and parsing the output into a 'degradome density file'. The degradome density file reflects the counts of 5' positions across the transcriptome.
- Alignment of query miRNAs or siRNAs to the transcriptome to generate a list of potential target sites. This uses the program "GSTAr.pl" (Generic Small RNA Transcript Aligner), which ships with the CleaveLand4 program. GSTAr.pl uses RNA-RNA thermodynamic predictions instead of sequence similarity to identify potential target sites, making it much slower than generic aligners, but more sensitive in terms of finding all possible sites.
- Cross-referencing the degradome data with the alignments to identify slicing sites with evidence of slicing. This includes assessment of p-values.

More details about how the program works are in the README (the README for version 4.3 is **Appendix A** of this document for convenience). **Appendix B** contains the README for version 1.0 of GSTAr.pl.

Installation

1. Install dependencies

Math::CDF. CleaveLand4.pl will not compile unless this required perl module is installed. If you have cpanm installed on your system, type

```
sudo cpanm Math::CDF
```

If you don't have cpanm, it is easy to install by typing

```
sudo App::cpanminus
```

If this doesn't work, consult CPAN http://www.cpan.org/modules/INSTALL.html for help.

samtools. See http://samtools.sourceforge.net/. Ensure samtools is in your PATH.

bowtie. See http://bowtie-bio.sourceforge.net/index.shtml. Ensure bowtie is in your PATH. Note, do NOT use 'bowtie2', as it is a very different type of aligner.

bowtie-build. See http://bowtie-bio.sourceforge.net/index.shtml. Ensure bowtie is in your PATH. bowtie-build comes along with bowtie.

RNAplex. This is part of the Vienna RNA package. See http://www.tbi.univie.ac.at/~ronny/RNA/index.html. Ensure RNAplex is in your PATH.

R. See http://www.r-project.org/. Ensure that R is in your PATH.

2. Install CleaveLand4.pl and GSTAr.pl
Download the tarball from http://axtell-lab-psu.weebly.com/cleaveland.html and unpack it. Add both CleaveLand4.pl and GSTAr.pl to your PATH.

3. Obtain tutorial data.
Download from http://axtelldata.bio.psu.edu/data/CleaveLand4_Tutorial/, unpack, and cd into the directory. You will find the following files:

GSM278370.fasta : adapter-trimmed degradome reads in FASTA format from *Arabidopsis thaliana*
TAIR10_cdna_20110103_rgmupdated_cleand.fasta : Arabidopsis cDNA annotations (TAIR10).
ath-miR171a.fasta : *Arabidopsis* miR171a mature miRNA sequence

Example

Examine the usage statement by calling CleaveLand4.pl with no arguments:

```
$ CleaveLand4.pl

CleaveLand4.pl : Finding sliced targets of small RNAs from degradome data

Version: 4.3

Usage: CleaveLand4.pl [options] > [out.txt]

Options:
-h Print help message and quit
-v Print version and quit
-q Quiet mode .. no log/progress information to STDERR
-a Sort small RNA / transcript alignments by Allen et al. score instead of default MFEratio -- for GSTAr
-t Output in tabular format instead of the default verbose format
-r [float >0..1] Minimum Free Energy Ratio cutoff. Default: 0.65 -- for GSTAr
-o [string] : Produce T-plots in the directory indicated by the string. If the dir does not exist, it will be
created
-d [string] : Path to degradome density file.
-e [string] : Path to FASTA-formatted degradome reads.
-g [string] : Path to GSTAr-created tabular formatted query-transcript alignments.
-u [string] : Path to FASTA-formatted small RNA queries
-n [string] : Path to FASTA-formatted transcriptome
-p [float >0..1] : p-value for reporting. Default is 1 (no p-value filtering).
-c [integer 0..4] : Maximum category for reporting. Default is 4 (all categories reported).

Modes:
1. Align degradome data, align small RNA queries, and analyze.
  REQUIRED OPTIONS: -e, -u, -n
  DISALLOWED OPTIONS: -d, -g
2. Use existing degradome density file, align small RNA queries, and analyze.
  REQUIRED OPTIONS: -d, -u, -n
  DISALLOWED OPTIONS: -e, -g
3. Align degradome data, use existing small RNA query alignments, and analyze.
  REQUIRED OPTIONS: -e, -n, -g
  DISALLOWED OPTIONS: -d, -u
  IRRELEVANT OPTIONS: -a, -r
4. Use existing degradome density file and existing small RNA query alignments, and analyze.
  REQUIRED OPTIONS: -d, -g
  DISALLOWED OPTIONS: -e, -u
  IRRELEVANT OPTIONS: -a, -r

Dependencies (must be in PATH):
  R [only if making T-plots]]
  GSTAr.pl [modes 1 and 2 .. Version 1.0 or higher]
  bowtie (0.12.x OR 1.x) [modes 1, 2, and 3]
  bowtie-build [modes 1, 2, and 3]
```

```
  RNAplex (from Vienna RNA Package) [modes 1 and 2]
  samtools [modes 1 and 3]

Documentation: perldoc CleaveLand4.pl
```

We will perform an example of a 'mode 1' analysis, in which degradome data is aligned to the reference, potential target sites are identified by GSTAr.pl, and the analysis occurs. To run this, type :

```
Algonquin:CleaveLand4_TUTORIAL michaelaxtell$ CleaveLand4.pl -e GSM278370.fasta -u ath-miR171a.fasta -n
TAIR10_cdna_20110103_rgmupdated_cleand.fasta -o tutorial_plots > tutorial_results.txt
```

It will take some time to align the data (including building a bowtie index for the reference, aligning, and parsing). After that, more time will be needed for GSTAr.pl to produce a list of possible target sites to examine. At the end, you will find three hits in the output, and corresponding T-plots in the directory 'tutorial_plots' (that was specified by option -o).

You can also examine the resulting degradome density file ("GSM278370.fasta_dd.txt") and GSTAr alignment file ("ath-miR171a.fasta_GSTAr.txt") to examine the format of these filetypes. Either one could be recycled in subsequent runs in modes 2, 3, or 4, depending upon the needs.

References Cited

Addo-Quaye C, Eshoo TW, Bartel DP, Axtell MJ. 2008. Endogenous siRNA and miRNA Targets Identified by Sequencing of the Arabidopsis Degradome. *Curr Biol* **18**: 758–762.

German MA, Pillay M, Jeong D-H, Hetawal A, Luo S, Janardhanan P, Kannan V, Rymarquis LA, Nobuta K, German R, et al. 2008. Global identification of microRNA–target RNA pairs by parallel analysis of RNA ends. *Nat Biotechnol* **26**: 941–946.

Gregory BD, O'Malley RC, Lister R, Urich MA, Tonti-Filippini J, Chen H, Millar AH, Ecker JR. 2008. A link between RNA metabolism and silencing affecting Arabidopsis development. *Dev Cell* **14**: 854–866.

Appendix A. README file for CleaveLand4.pl, version 4.3

```
LICENSE
    CleaveLand4.pl

    Copyright (c) 2013 Michael J. Axtell

    This program is free software: you can redistribute it and/or modify it
    under the terms of the GNU General Public License as published by the
    Free Software Foundation, either version 3 of the License, or (at your
    option) any later version.

    This program is distributed in the hope that it will be useful, but
    WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General
    Public License for more details.

    You should have received a copy of the GNU General Public License along
    with this program. If not, see <http://www.gnu.org/licenses/>.

SYNOPSIS
    CleaveLand4 : Finding evidence of sliced targets of small RNAs from
    degradome data

AUTHOR
    Michael J. Axtell, Penn State University, mja18@psu.edu

VERSION
    4.3 : November 7, 2013

INSTALL
```

```
  Dependencies - Required Perl Modules
            Getopt::Std
            Math::CDF

    CleaveLand4.pl is a perl program, so it needs perl installed on your
    system. It was developed on perl version 5.10.0, and hasn't been tested
    on other versions (but there is no reason to suspect problems with other
    perl 5.x versions). CleaveLand4.pl will not compile unless the above two
    Perl modules are also installed in your perl's @INC. Getopt::Std is
    pre-loaded into most (all?) Perl distros. But you may need to install
    Math::CDF from CPAN. Only one Math::CDF function is used by CleaveLand4
    ('pbinom').

  Dependencies - PATH executables
            bowtie (version 0.12.x or 1.x)
            bowtie-build
            RNAplex (from Vienna RNA package)
            GSTAr.pl (Version 1.0 or higher -- distrubuted with CleaveLand4)
            R
            samtools

    All of the above must be executable from your PATH. Depending on the
    mode of the CleaveLand4 run (see below), only a subset of these programs
    may be required for a given run.

  Installation
    Except for the above dependencies, there is no "real" installation. If
    the script is in your working directory, you can call it with

            ./CleaveLand4.pl

    For convenience, you can add it to your PATH. e.g.

            sudo mv CleaveLand4.pl /usr/bin/

    GSTAr.pl expects to find perl in /usr/bin/perl .. if not, edit line 1
    (the hashbang) accordingly.

USAGE
            CleaveLand4.pl [options] > [out.txt]

    Log and progress information goes to STDERR, and can be suppressed with
    option -q (quiet mode).

  Options
    -h Print help message and quit

    -v Print version and quit

    -q Quiet mode .. no log/progress information to STDERR

    -a Sort small RNA / transcript alignments by Allen et al. score* instead
    of default MFEratio -- for GSTAr

    -t Output in tabular format instead of the default verbose format

    -r [float >0..1] Minimum Free Energy Ratio cutoff. Default: 0.65 -- for
    GSTAr

    -o [string] : Produce T-plots in the directory indicated by the string.
    If the dir does not exist, it will be created

    -d [string] : Path to degradome density file.

    -e [string] : Path to FASTA-formatted degradome reads.

    -g [string] : Path to GSTAr-created tabular formatted query-transcript
    alignments.

    -u [string] : Path to FASTA-formatted small RNA queries

    -n [string] : Path to FASTA-formatted transcriptome

    -p [float >0..1] : p-value for reporting. Default is 1 (no p-value
    filtering).

    -c [integer 0..4] : Maximum category for reporting. Default is 4 (all
    categories reported).

    *Allen et al. score: This is a score based on the position-specific
    penalties described by Allen et al. (2005) Cell, 121:207-221 [PMID:
    15851028]. Specifically, mismatched query bases or target-bulged bases,
    are penalized 1. G-U wobbles are penalized 0.5. These penalties are
```

```
    double within positions 2-13 of the query.

  Modes
    CleaveLand4 runs in one of four different modes. Each mode has a
    required set of options and a disallowed set of options, as described
    below:

    Mode 1: Align degradome data and create degradome density file, perform
    new small RNA query/transcriptome alignment with GSTAr, and analyze.
    Required options: -e, -u, -n. Disallowed options: -d, -g.

    Mode 2: Use existing degradome density file, perform new small RNA
    query/transcriptome alignment with GSTAr, and analyze. Required options:
    -d, -u, -n. Disallowed options: -e, -g.

    Mode 3: Align degradome data and create degradome density file, use
    existing GSTAr alignments, and analyze. Required options: -e, -n, -g.
    Disallowed options: -d, -u.

    Mode 4: Use existing degradome density file and existing GSTAr
    alignments, and analyze. Required options: -d, -g. Disallowed options:
    -e, -u.

METHODS
  Degradome data --> transcriptome alignments --> degradome density file creation (modes 1 and 3)
    Degradome data is aligned to the reference transcriptome using bowtie.
    If needed, the bowtie indices for the transcript are built with
    bowtie-build using default parameters. This results in the creation of
    six files, each including "ebwt" in their suffix. Alignment parameters
    allow zero or one mismatch, and are only allowed to the forward strand
    of the transcriptome. In the case of multiple valid alignments only one
    is randomly selected and reported. (The specific bowtie command used is
    "bowtie -f -v 1 --best -k 1 --norc -S"). The alignment process uses
    samtools to generate a sorted BAM alignment file from the bowtie output
    stream.

    After creation of the sorted BAM alignment file, the alignments are
    parsed to quantify the density of observed 5' ends at each nt of the
    transcriptome. The results are written to a 'degradome density' file in
    the working directory. The BAM alignment file is deleted at the
    completion of this process. The degradome density files contain the
    position of the transcript, the number of 5' ends at that position, and
    the degradome peak 'category'. Categories are determined as follows:

    Category 4: Just one read at that position

    Category 3: >1 read, but below or equal to the average* depth of
    coverage on the transcript

    Category 2: >1 read, above the average* depth, but not the maximum on
    the transcript

    Cateogry 1: >1 read, equal to the maximum on the transcript, when there
    is >1 position at maximum value

    Cateogry 0: >1 read, equal to the maximum on the transcript, when there
    is just 1 position at the the maximum value

    * Note that the average does not include all of the 'zeroes' for
    non-occupied positions within a transcript. Instead, it is the average
    of all positions that have at least one read.

  Small RNA --> transcriptome alignments with GSTAr (modes 1 and 2)
    Potential target sites are generated with GSTAr.pl, which ships with
    CleaveLand4. Options -r and -a are passed to GSTAr. By default,
    potential target sites are sorted in descending order by MFE ratio. If
    the -a switch is present, this is changed to ascending order based on
    Allen et al. score. Note that GSTAr can be expected to take 90-120
    seconds per query when analyzing a typically sized transcritpome.
    GSTAr.pl is always called to output in tabular format by CleaveLand4.pl.
    Only alignments to the reverse-comp strand of the transcriptome are
    considered. Upon completion, a GSTAr alignment file is written to the
    working directory. See the GSTAr documentation for more details on this
    program.

  Analysis (all modes)
    After loading valid degradome density and GSTAr alignment files,
    CleaveLand4 first checks to ensure that the transcriptome (as noted in
    the headers) is the same. If so, analysis progresses. For each alignment
    in the GSTAr alignment file, CleaveLand4 searches the degradome density
    file to see if there are any degradome reads at the predicted slicing
    site. If there are, a p-value is calculated. The p-value takes into
    account both the noise in the degradome density file and the quality of
```

```
    the small RNA-transcriptome alignment. First, the chances of observing a
    deagrdome 'peak' of the given category by random chance is calculated.
    The chance is the total number of peaks of the given category divided by
    the effective transcriptome size*. Then, the quality of the alignment is
    simply the rank of the alignment in the GSTAr alignment file (which is
    either sorted by MFE ratio [default] or by Allen et al. score). The
    p-value is calculated as the binomial probability of observing one or
    more 'hits' in 'x' trials given probability 'c', where 'x' is the rank
    of the alignment, and 'c' is the chance described above.

    * The effective transcriptome size is the total number of bases in the
    transcriptome - (n * mean_read_size), where 'n' is the number of
    transcripts. This adjustment accounts for the fact that the very ends of
    the transcripts could not possibly have any mapped 5' ends.

    Any hits with a p-value <= the cutoff specified by option -p AND a
    category <= the cutoff specified by option -c are output to STDOUT. By
    default, all hits are reported (option p default is 1, option c default
    is 4).

INPUT FILE FORMAT REQUIREMENTS
  Newlines
    All files are assumed to have "\n" as newline characters. Files with
    MS-DOS text encoding, or others, that do not conform to this assumption
    will cause unexpected behavior and likely meaningless results.

  Transcriptome (option -n)
    This must be a multiline FASTA file. The headers should be short and
    simple and devoid of whitespace (e.g. ">AT1G12345" is good, ">AT1G12345
    | this is my favorite gene | it is awesome" is not. The filename of the
    transcriptome file should also be devoid of whitespace.

  Degradome reads (option -e)
    This must be a multiline FASTA file. The reads are assumed to have
    already clipped to remove adapters. Furthermore, the reads must not have
    been collapsed in any way. In other words, each read off the sequencer
    should have an entry. Sequences that appear 50 times in the raw data
    from 50 different reads should each have a line.

    Finally, CleaveLand4 assumes that each degradome read represents the
    5'-3' sequence of a transcript, and that the first nt of each read
    represents the 5' end of an RNA.

  Small RNA Queries (option -u)
    This must be a multiline FASTA file with the full sequence of a given
    small RNA on one line (e.g. each line is either a header beginning with
    ">" or the full-length sequence of the small RNA). The headers should be
    short and simple and devoid of whitespace (e.g. ">ath-miR169a" is good,
    ">ath-miR169a MIMAT0000200 Arabidopsis thaliana miR169a" is not.
    Sequences can have either T's or U's.

    Note: miRBase often has several mature miRNAs with exactly the same
    sequence, relfecting paralogous miRNA genes within a species. There is
    no use querying the same sequence multiple times, so it is a good idea
    to collapse the redundancy by query sequence when making a file of small
    RNA queries.

  Degradome density files (option -d)
    Most of the time, these will be files created by previous runs of
    CleaveLand4 that will have the suffix "_dd.txt". If you don't like the
    alignment parameters that CleaveLand4 uses, you could create your own
    degradome denstiy files. The format specification is:

    Header region: Lines begin with "#". Here is an example:

    [line1]# CleaveLand4 degradome density

    [line2]# Fri Sep 13 09:21:38 EDT 2013

    [line3]# Degradome Reads:GSM278335.fasta

    [line4]# Transcriptome:TAIR10_cdna_20110103_rgmupdated_cleand.fasta

    [line5]# TranscriptomeCharacters:51074197

    [line6]# Mean Degradome Read Size:20

    [line7]# Estimated effective Transcriptome Size:50402157

    [line8]# Category 0:16430

    [line9]# Category 1:3456
```

```
    [line10]# Category 2:95747

    [line11]# Category 3:207062

    [line12]# Category 4:78279

    CleaveLand4 demands that the first line of a valid degradome density
    file is "# CleaveLand4 degradome density". It also requires all other
    header lines, except the date on line 2, to be present. All of this
    information is required for analysis.

    Data Region: Each transcript begins with two lines as follows:

    [line1]@ID:AT1G50920.1

    [line2]@LN:2394

    The @ID: gives the name of the transcript, while the @LN: gives the
    length of the transcript. After this, each line gives a one-based
    position, the number of 5' ends at that positions, and the degradome
    category. The data lines are tab-delimited. Note that positons with zero
    reads are NOT shown.

  GSTAr query-transcriptome alignments (option -g)
    These are files created by GSTAr. If they were created as part of a
    CleaveLand4 run, they will have the suffix "_GSTAr.txt". They must be in
    the 'tabular' format, and have a proper header as shown below:

    [line1]# GSTAr version 1.0

    [line2]# Thu Sep 12 13:56:58 EDT 2013

    [line3]# Queries: test_mir.fasta

    [line4]# Transcripts: TAIR10_cdna_20110103_rgmupdated_cleand.fasta

    [line5]# Hit seed length required to initiate RNAplex analysis (option
    -s): 7

    [line6]# Minimum Free Energy Ratio cutoff (option -r): 0.65

    [line7]# Sorted by: MFEratio

    [line8]# Output Format: Tabular

    Is is strongly recommended NOT to try to produce these files by means
    not involving CleaveLand4/GSTAr.

OUTPUT
  Pretty format
    By default, CleaveLand prints hits that pass the p-value and category
    filters to STDOUT in a human-readble, verbose format that is
    self-explanatory. A header (lines beginning with "#") is printed giving
    basic information on the analysis.

  Tabular format
    If option -t is specified, any hits passing the p-value filter are
    printed in a tab-delimited format. First, a header (lines beginning with
    "#") is printed giving basic information on the analysis. After that, a
    line giving the names of the columns is printed. Each subsequent line
    gives information on a single hit. The format is similar to that of the
    GSTAr alignments. Column information is:

    1: SiteID: A unique name (within the scope of the output of a particular
    run) for the putative slicing site. In the form
    "[transcript]:[slice_site]". The output is sorted by SiteID.

    2: Query: Name of query

    3: Transcript: Name of transcript

    4: TStart: One-based start position of the alignment within the
    transcript

    5: TStop: One-based stop position of the alignment within the transcript

    6: TSLice: One-based position of the alignment opposite position 10 of
    the query

    7: MFEperfect: Minimum free energy of a perfectly matched site
    (approximate)

    8: MFEsite: Minimum free energy of the alignment in question
```

```
    9: MFEratio: MFEsite / MFEperfect

    10: AllenScore: Penalty score calculated per Allen et al. (2005) Cell,
    121:207-221 [PMID: 15851028].

    11: Paired: String representing paired positions in the query and
    transcript. The format is Query5'-Query3',Transcript3'-Transcript5'.
    Positions are one-based. Discrete blocks of pairing are separated by ;

    12: Unpaired: String representing unpaired positions in the query and
    transcript. The format is
    Query5'-Query3',Transcript3'-Transcript5'[code]. Possible codes are
    "UP5" (Unpaired region at 5' end of query), "UP3" (Unpaired region at 3'
    end of query), "SIL" (symmetric internal loop), "AILt" (asymmetric
    internal loop with more unpaired nts on the transcript side), "AILq"
    (asymmetric internal loop with more unpaired nts on the query side),
    "BULt" (Bulged on the transcript side), or "BULq" (bulged on the query
    side). Positions are one-based. Discrete blocks of pairing are separated
    by ;

    13: Structure: Aligned secondary structure. The region before the "&"
    represents the transcript, 5'-3', while the region after the "&"
    represents the query, 5'-3'. "(" represents a transcript base that is
    paired, ")" represents a query based that is paired, "." represents an
    unpaired base, and "-" represents a gap inserted to facilitate
    alignment.

    14: Sequence: Aligned sequence. The region before the "&" represents the
    transcript, 5'-3', while the region after the "&" represents the query,
    5'-3'.

    15: DegradomeCategory:

            Category 4: Just one read at that position

            Category 3: >1 read, but below or equal to the average* depth of coverage on the transcript

            Category 2: >1 read, above the average* depth, but not the maximum on the transcript

            Cateogry 1: >1 read, equal to the maximum on the transcript, when there is >1 position at maximum
value

            Cateogry 0: >1 read, equal to the maximum on the transcript, when there is just 1 position at the
the maximum value

    16: DegradomePval: p-value for this degradome hit.

    17: Tplot_file_path: File path for the T-plot of this hit.

    * Note that the average does not include all of the 'zeroes' for
    non-occupied positions within a transcript. Instead, it is the average
    of all positions that have at least one read. =head2 T-plots

    If the user requests them by including the -o option, T-plots for each
    hit that passes the p-value cutoff are created and written to the
    directory specificied by option -o. The black line on the plot shows all
    of the degradome data, and the red dot shows the putative slicing site.
    The title of each T-plot indicates the transcript ("T="), the query
    ("Q="), and the putative slicing site ("S="), as well as the category
    and p-value.

    Existing T-plot files in the -o directory with the same name will be
    over-written without warning.

WARNINGS
  Don't believe the hype - part 1
    Under default settings, CleaveLand4 reports ALL putative slicing sites
    with ANY degradome reads at all, regardless of the liklihood of a given
    hit of being due to random chance. Without any filtering, most of your
    hits probably ARE due to random chance. This means that there will be
    many many hits of Category 4 (just one read) and/or at very high
    p-values. Exercise skepticism when interpreting these results. More
    confidence in the reality of a given slicing event can come from
    restricting analysis to hits with low p-values and/or high categories,
    and, even better, observing the slicing event in multiple degradome
    libraries.

  Don't believe the hype - part 2
    The p-value calculation is built around the ASSUMPTION that the rank
    order of alignments for a given query reflects their liklihood of being
    functional. Under default settings, GSTAr will sort the alignments for
    each query based on descending MFE ratio. Alternatively, when option a
```

```
  is specified for the GSTAr run, they will be sorted in ascending order
  according the Allen et al. score. The extent to which the p-values are
  trustworthy is dictated directly by the extent to which you believe that
  MFEratio or Allen et al. scores are predictive of function. If you don't
  believe that, you should treat the p-values with due skepticism, and
  focus instead on high category hits and reproducibility between
  libraries.

Not for whole genomes
  Degradome alignment by CleaveLand4 only searches the top strand of the
  transcriptome. Also, GSTAr holds the entire contents of the
  transcripts.fasta file in memory to speed the isolation of
  sub-sequences, and CleaveLand4 holds the entire contents of the
  degradome density file in memory. This will be impractical in terms of
  memory usage if a user attempts to load a whole genome. Similarly, GSTAr
  will only search for pairing between the top strand of the
  transcripts.fasta file, making it also impractical for a genome
  analysis, where sites might be on either strand.

Temp files
  CleaveLand4 writes several temp files during the course of a run. So,
  don't mess with them during a run. In addition, it is a very bad idea to
  have two CleaveLand4 runs operating concurrently from the same working
  directory. CleaveLand4 will clean up all temp files at the conclusion of
  a run.

Not too fast in modes 1 and 2 (and maybe 3).
  GSTAr is a very fast intermolecular RNA-RNA hybridization calculator.
  But when applied to whole transcriptomes, it is still very
  time-consuming. When running in modes 1 or 2, plan on about 90-120
  seconds per query during the GSTAr phase. Additionally, bowtie
  alignments and index bulding (modes 1 or 3) can also be time-consuming.
  Finally, requesting T-plots can also slow things down, especially if a
  great number of hits are being returned.

No ambiguity codes
  Query sequences with characters other than A, T, U, C, or G
  (case-insensitive) will not be analyzed, and a warning will be sent to
  the user. Transcript sub-sequences for potential query alignments will
  be *silently* ignored if they contain any characters other than A, T, U,
  C,or G (case-insensitive).

Small queries
  GSTAR demands that query sequences must be small (15-26 nts). Queries
  that don't meet these size requirements will not be analyzed and a
  warning sent to the user.

No redundancy
  For a GIVEN QUERY, GSTAr alignments are non-redundant in terms of the
  slicing site of the alignment. However, a single query can have multiple
  overlapping alignment patterns that have differing predicted slicing
  sites. In addition, if multiple queries are similar in sequence, the
  same alignment (in terms of putative slicing site) can be present
  multiple times for different queries. If CleaveLand4 identifies more
  than one alignment at the same putative slicing site, it will only
  report the one with the best (lowest) p-value (subject to the maximum
  allowed p-value, option -p). Therefore there should be no redundancy in
  the putative slice sites returned by a given CleaveLand4 run.

No reverse-compatibility
  Degradome density files created by versions of CleaveLand prior to 4.0
  are NOT compatible with CleaveLand 4. Sorry.

Change in category definitions
  The categories used by CleaveLand4 differ slightly from those used in
  CleaveLand3 and earlier. Specifically, categories 3 and 2 now rely upon
  calculating the mean, not the median, level of coverage in the
  transcript. In addition, transcript positions with zero coverage are no
  longer included in the calculation. The effect of this is to make
  category 2 hits much more rare, and category 3 hits much more common.

Slicing at position 10
  CleaveLand4 only looks for evidence of slicing at position 10 relative
  to the aligned small RNA. There is no ambiguity -- data at position 11
  or 9 is not relevant to CleaveLand4. This is because, as far as I know,
  there is no direct evidence showing Argonaute proteins cut anywhere
  besides position 10. However, there IS clear evidence for isomirs:
  lower-abundance variants of miRNAs with alternative 5' or 3' ends.
  Isomirs with alternative 5' ends could certainly cause offset slicing.
  If you wish to search for slicing at slightly 'off' locations with
  CleaveLand4, you will need to explictly query with the isomirs of
  interest.
```

Appendix B. README file for GSTAr.pl, version 1.0

```
LICENSE
    GSTAr.pl

    Copyright (c) 2013 Michael J. Axtell

    This program is free software: you can redistribute it and/or modify it
    under the terms of the GNU General Public License as published by the
    Free Software Foundation, either version 3 of the License, or (at your
    option) any later version.

    This program is distributed in the hope that it will be useful, but
    WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General
    Public License for more details.

    You should have received a copy of the GNU General Public License along
    with this program. If not, see <http://www.gnu.org/licenses/>.

SYNOPSIS
    GSTAr : Generic Small RNA-Transcriptome Aligner.

    Flexible RNAplex-based alignment of miRNAs and siRNAs (15-26 nts) to a
    transcriptome.

AUTHOR
    Michael J. Axtell, Penn State University, mja18@psu.edu

VERSION
    1.0 : September 17, 2013

INSTALL
  Dependencies

            perl
            RNAplex (from Vienna RNA package)

    RNAplex must be executable from your PATH. GSTAr.pl was developed using
    RNAplex version RNAplex 2.1.3. It has not been tested on other versoins
    of RNAplex.

  Installation

    No "real" installation. If the script is in your working directory, you
    can call it with

            ./GSTAr.pl

    For convenience, you can add it to your PATH. e.g.

            sudo mv GSTAr.pl /usr/bin/

    GSTAr.pl expects to find perl in /usr/bin/perl .. if not, edit line 1
    (the hashbang) accordingly.

USAGE
            GSTAr.pl [options] queries.fasta transcriptome.fasta

    Output alignments go to STDOUT, and can be redirected to a file with >
    or piped to another process with |

    Log and progress information goes to STDERR, and can be suppressed with
    option -q (quiet mode).

  Options

    Options:

    -h Print help message and quit

    -v Print version and quit

    -q Quiet mode .. no log/progress information to STDERR

    -t Tabular output format ... More suitable for parsing

    -a Sort by Allen et al. score instead of the default MFEratio

    -r [float >0..1] Minimum Free Energy Ratio cutoff. Default: 0.65

METHODS
```

```
    GSTAr.pl is essentially a wrapper and parser for RNAplex desgined for
    aligning short queries (15-26nts) against the rev-comp. strand of a
    eukaryotic transcriptome. For each query sequence, the minimum free
    energy of a perfectly complelementary sequence is calculated using
    RNAplex under all default parameters. Following this, the same query is
    then analyzed against the entire transcriptome. Hits where the MFEratio
    (i.e. MFE / MFE-perfect) is >= the cutoff established by option r are
    retained and parsed.

    The detailed RNAplex parameters (see RNAplex man page for details) for
    each query analysis are

            -f 2 : Fast mode .. structure based on approximated plex model.
            -e [minMFEratio * perfectMFE] : Minimum acceptable MFE value to keep a hit
            -z : [10 + query_length] : Acceptable alignments can span no more than length of query + 10nts.

    Slice Site is the transcript nt opposite nt 10 of the query. This is
    where one should look to find evidence of AGO-catalyzed slicing in the
    event that a) the transcript was really a target of the query at that
    site, and b) it really was sliced. GSTAr makes no judgements on the
    likelihood of either of those events, and the recording of the Slice
    Site position should NOT be taken as evidence that slicing exists or is
    even possible at that site.

    For a given query, all returned sites are non-redundant. Redunancy is
    based upon the putative slice site only. By default, the output is
    sorted in descending order (best to worst) according to the MFE ratio.
    In the alternative option -a mode, the results are instead sorted in
    ascending order (best to worst) according to the Allen et al. score.

    Allen et al. score: This is a score for plant miRNA/siRNA-target
    interactions based on the position-specific penalties described by Allen
    et al. (2005) Cell, 121:207-221 [PMID: 15851028]. Specifically,
    mismatched query bases or target-bulged bases, are penalized 1. G-U
    wobbles are penalized 0.5. These penalties are double within positions
    2-13 of the query.

WARNINGS
  NOT a target predictor

    GSTAr is very explicitly NOT a target predictor for miRNAs or siRNAs. It
    is only an aligner based on RNA-RNA hybridization thermodynamic
    predictions. Users should make no claims as to whether the identified
    alignments are actually targets of the query without independent data of
    some sort.

  Slice Sites are NOT predictions of slicing

    Although GSTAr reports a "Slicing Site" position for each alignment,
    this is merely for conveneince when using GSTAr alignments to guide
    subsequent experiments searching for AGO-catalyzed slicing evidence. No
    claim is made that any alignment is actually AGO-cleaved or even
    theoretically AGO-cleavable.

  Not for whole genomes

    GSTAr holds the entire contents of the transcripts.fasta file in memory
    to speed the isolation of sub-sequences. This will be impractical in
    terms of memory usage if a user attempts to load a whole genome.
    Similarly, GSTAr will only search for pairing between the top strand of
    the transcripts.fasta file, making it also impractical for a genome
    analysis, where sites might be on either strand.

  Temp files

    GSTAr writes temp files to the working directory. Their contents change
    dynamically during a run, and they will be deleted at the end of a run.
    So, don't mess with them during a run. In addition, it is a very bad
    idea to have two GSTAr runs operating concurrently from the same working
    directory because there will be clashes and overwrites for these temp
    files.

  Not too fast

    GSTAr uses RNAplex (Tafer and Hofacker, 2008. Bioinformatics 24:2657-63,
    PMID: 18434344, doi:10.1093/bioinformatics/btn193), which is
    exceptionally fast for an inter-molecular RNA-RNA hybridization
    calculator. However, when applied to entire eukaryotic transcriptomes
    the CPU time per query is still significant. Run time is only slightly
    affected (much less than 2-fold) by the setting of -r. Setting tabular
    mode (option -t) also increases speed just a tiny bit for runs with a
    low option -r. In tests with the Arabidopsis transcriptome (33,602
    mRNAs, total nts=51,074,197), a single 21nt miRNA query typically takes
```

```
    about 90-110 seconds to complete.

  No ambiguity codes

    Query sequences with characters other than A, T, U, C, or G
    (case-insensitive) will not be analyzed, and a warning will be sent to
    the user. Transcript sub-sequences for potential alignments will be
    *silently* ignored if they contain any characters other than A, T, U,
    C,or G (case-insensitive).

  Small queries

    Query sequences must be small (between 15 and 26nts). Queries that don't
    meet these size requirements will not be analyzed and a warning sent to
    the user.

  Redundant output

    GSTAr guarantees that, FOR A GIVEN QUERY, the returned alignments will
    be unique in terms of their PUTATIVE SLICING SITE POSITION. However, the
    same query could generate multiple overlapping aligments that each have
    different putative slice sites. Furthermore, if different queries in a
    multi-query analysis have similar (or identical !) sequences, the same
    alignment position (based on putative slicing site position) could be
    returned multiple times, once for each of the similar/identical queries.
    Therefore, for multi-query result files, there is no guarantee of
    non-redundancy among the returned sites.

OUTPUT
    Both output formats begin with a series of commented lines that provide
    details about the run.

  Pretty output

    The default output "pretty" mode is easily parsed by humans and should
    be self-explanatory

  Tabular output

    Tabular output (which occurs when option -t is specified) is a
    tab-delimited text file. The first non-commented line is the column
    headings, with meanings as follows:

    1: Query: Name of query

    2: Transcript: Name of transcript

    3: TStart: One-based start position of the alignment within the
    transcript

    4: TStop: One-based stop position of the alignment within the transcript

    5: TSLice: One-based position of the alignment opposite position 10 of
    the query

    6: MFEperfect: Minimum free energy of a perfectly matched site
    (approximate)

    7: MFEsite: Minimum free energy of the alignment in question

    8: MFEratio: MFEsite / MFEperfect

    9: AllenScore: Penalty score calculated per Allen et al. (2005) Cell,
    121:207-221 [PMID: 15851028].

    10: Paired: String representing paired positions in the query and
    transcript. The format is Query5'-Query3',Transcript3'-Transcript5'.
    Positions are one-based. Discrete blocks of pairing are separated by ;

    11: Unpaired: String representing unpaired positions in the query and
    transcript. The format is
    Query5'-Query3',Transcript3'-Transcript5'[code]. Possible codes are
    "UP5" (Unpaired region at 5' end of query), "UP3" (Unpaired region at 3'
    end of query), "SIL" (symmetric internal loop), "AILt" (asymmetric
    internal loop with more unpaired nts on the transcript side), "AILq"
    (asymmetric internal loop with more unpaired nts on the query side),
    "BULt" (Bulged on the transcript side), or "BULq" (bulged on the query
    side). Positions are one-based. Discrete blocks of pairing are separated
    by ;

    12: Structure: Aligned secondary structure. The region before the "&"
    represents the transcript, 5'-3', while the region after the "&"
    represents the query, 5'-3'. "(" represents a transcript base that is
```

```
paired, ")" represents a query based that is paired, "." represents an
unpaired base, and "-" represents a gap inserted to facilitate
alignment.

13: Sequence: Aligned sequence. The region before the "&" represents the
transcript, 5'-3', while the region after the "&" represents the query,
5'-3'.
```