FASTAptameR 2.0 - User Interface Tutorial

Skyler Kramer, Paige Gruenke, Khalid Alam, Rebecca Burke-Agüero, Dong Xu, Donald Burke

3/22/2021

Contents

1 Introduction

FASTAptameR 2.0 is an R-based update of FASTAptamer (Alam KK 2015). Like its predecessor, FASTAptameR 2.0 is an open-source toolkit designed to analyze populations of sequences resulting from combinatorial selections. This updated version features a user interface (UI), interactive graphics, more modules, and a faster implementation of the original clustering algorithm.

This user guide will walk you through each of the modules and highlight what options you have when analyzing your data through the UI. Please see package documentation for details on running these scripts outside of the UI.

1.1 Overview

- **FASTAptameR-Count**
  - *Entry point into FASTAptameR 2.0*
  - Input: preprocessed FASTQ/A
  - Workflow:
    1. count all unique sequences (`Reads`)

        2. sort by counts (`Rank`)
        3. normalize counts as reads per million (`RPM`)
    - Plotting:
        1. line plot of reads-per-rank
        2. histogram of sequence lengths
    - Output: FASTA or CSV

- **FASTAptameR-Translate**
    - Input: counted FASTA
    - Workflow: translate D/RNA sequences to amino acid sequences
    - Plotting:
        1. line plot of reads-per-rank
        2. histogram of sequence lengths
    - Output: FASTA or CSV
- **FASTAptameR-Motif__Search**
    - Input: counted FASTA
    - Workflow: search for user-defined motifs in sequences
    - Output: FASTA or CSV
- **FASTAptameR-Motif__Enrich**
    - Input: 2-3 counted FASTAs
    - Workflow: calculate how a single user-defined motif enriches across 2-3 populations
    - Output: CSV
- **FASTAptameR-Distance**
    - Input: counted FASTA
    - Workflow: compute the Levenshtein edit distance (LED) between a single query sequence and all other provided sequences
    - Plotting: histogram of edit distances
    - Output: CSV
- **FASTAptameR-Enrich**
    - Input: 2-3 counted FASTAs
    - Workflow: calculate how each sequence enriches across 2-3 populations
    - Plotting:
        1. histogram(s) of fold changes
        2. scatter plot(s) of RPM
        3. volcano plot of fold change and a term related to frequency
    - Output: CSV
- **FASTAptameR-Cluster**
    - Input: counted FASTA
    - Workflow:
        1. filter out low-read sequences
        2. treat most abundant, non-clustered sequence as cluster seed
        3. add all sequences within a given LED of the seed to the cluster
        4. Repeat until all sequences are clustered or a maximum number of clusters are created
    - Output: FASTA or CSV
- **FASTAptameR-Cluster__Analysis**
    - Input: clustered FASTA

  - Workflow: provide metadata for each cluster
  - Output: CSV

- **FASTAptameR-Cluster_Enrich**
  - Input: clustered FASTA
  - Workflow: calculate how each cluster enriches across 2-3 populations
  - Output: CSV

A summary of inputs and outputs is given by the following table:

Table 1: Module Inputs and Outputs

Module	Input Files	Output Files
FASTAptameR-Count	Preprocessed FASTQ/A	FASTA or CSV
FASTAptameR-Translate	Counted FASTA	FASTA or CSV
FASTAptameR-Motif_Search	Counted FASTA	FASTA or CSV
FASTAptameR-Motif_Enrich	2 or 3 counted FASTAs	CSV
FASTAptameR-Distance	Counted FASTA	CSV
FASTAptameR-Enrich	2 or 3 counted FASTAs	CSV
FASTAptameR-Cluster	Counted FASTA	FASTA or CSV
FASTAptameR-Cluster_Analysis	Clustered FASTA	CSV
FASTAptameR-Cluster_Enrich	2 or 3 cluster-analysis CSVs	CSV

Note that many function inputs / outputs are simply FASTA files, so FASTAptameR 2.0 can be easily integrated into most analytical pipelines.

Importantly, FASTAptameR 2.0 does not provide any functions that are easily addressed by other software (*e.g.*, merging paired-end reads, trimming constant regions, predicting structures, *etc.*). Rather, the focus of this application is to provide flexible downstream analyses for the selections field.

1.2 Example workflow

We show all module connections in **Fig. 1**. *Orange* represents the preprocessed FASTA/Q input file. *Green* represents the entry point into the software and, thus, should be included in all pipelines. *Blue* represents modules that can feed into others. Finally, *red* represents modules that cannot feed into subsequent modules.

2 Application accessibility

Exactly like its predecessor, FASTAptameR 2.0 is designed to be **easy** to use. The web server is the easiest way to interact with this application because it only requires an internet connection and browser. However, if data restrictions (*e.g.*, size or privacy) prevent you from using a web application, you can locally run it as a Docker container. Finally, this application can be accessed through R if you are familiar with the language.

2.1 User interface

2.1.1 Web Server

This is the easiest way to use the FASTAptameR 2.0 UI. The web server can be accessed from **LINK**, which is hosted by the Digital Biology Laboratory under the direction of Dr. Dong Xu. However, this option only works if your files are less than 1 GB.

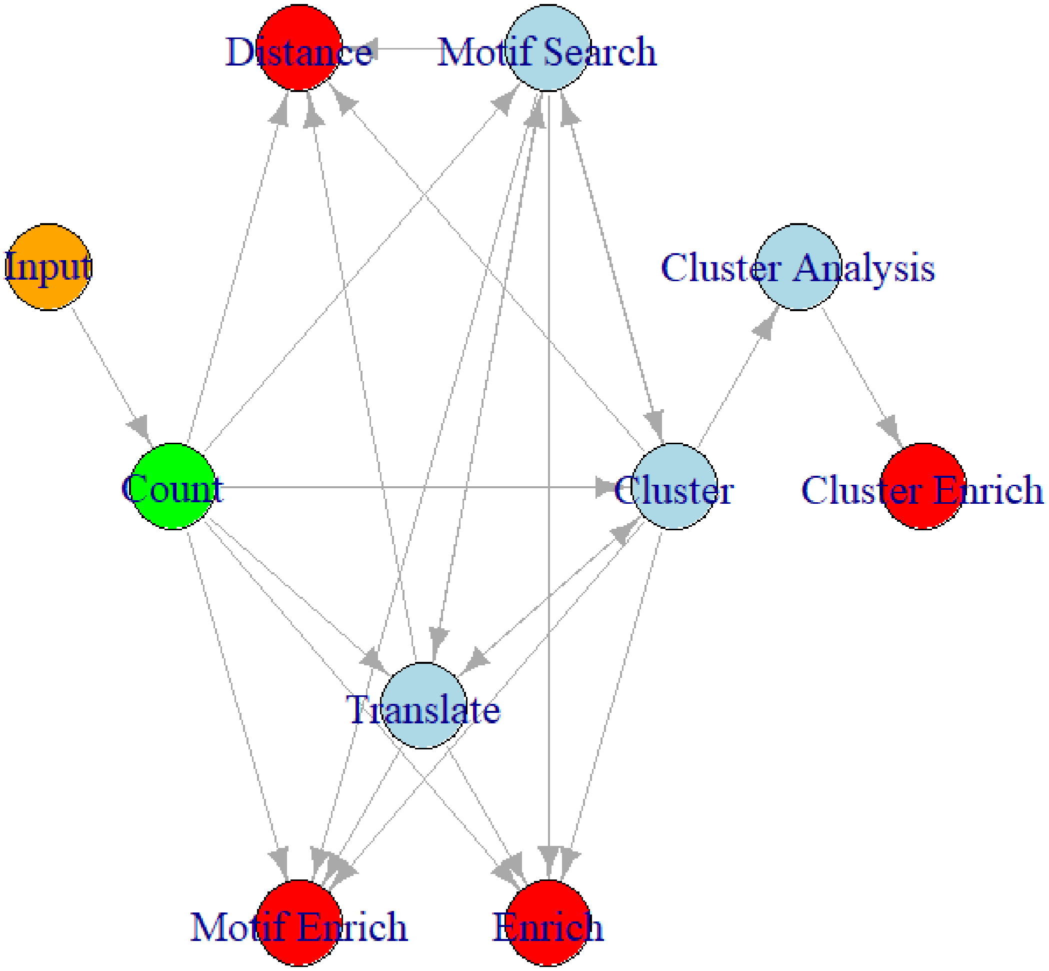

Figure 1: Module connections

2.1.2 Docker

If you have files larger than 1 GB, want to use local resources, or want to avoid uploading data, the web server can be locally run via Docker. In brief, Docker is a high-level API that may be used to construct *images* of *containers*. The *image* essentially functions as the blueprint for an application. The *image* of FASTAptameR 2.0, for example, contains all relevant software (*e.g.*, `R`), files (*e.g.*, this PDF), and packages (*e.g.*, `Shiny`). This *image* must be pulled from a repository (Docker Hub) by running **PULL COMMAND** in a Docker-active terminal.

Importantly, the FASTAptameR 2.0 *image* is built on Linux. Thus, it is necessary to run it from a Linux environment or virtual machine. For Mac or Windows users, the installation instructions for Docker Desktop (https://docs.docker.com/get-docker/) will show you how to do this.

Once you have this application's *image*, running **RUN COMMAND** from a Docker-active terminal will launch a local instance - a *container* - of FASTAptameR 2.0. You will then interact with this *container* in the same fashion as the web server.

For details on Docker or its installation, please see https://www.docker.com/ and https://docs.docker.com/get-docker/, respectively.

2.2 R

For experienced R users, all code (UI and modules) can be accessed through GitHub: **LINK**. `install_packages.R` will install all required packages and should be run first. `app.R` and `Functions.R` should be kept in the same directory. The former contains the UI and server code, whereas the latter contains all functions necessary to analyze your data.

The functions from `Functions.R` are available as a GitHub package and can be installed by **INSTALL COMMAND**. For details on usage, see the corresponding documentation: **LINK TO MANUAL**.

2.3 Software usage

If you use, adapt, or modify FASTAptameR 2.0, please cite: **CITATION**.

For any questions or concerns, please email burkelab@missouri.edu or stk7c9@umsystem.edu.

3 Tutorial

3.1 Data requirements

FASTAptameR 2.0 utilizes many string-based functions in its backend. Thus, this program can be used to analyze many types of populations, such as aptamers, peptides, and more. However, all libraries must be initially saved in a FASTA or FASTQ format and passed through **FASTAptameR-Count** prior to any subsequent analyses. Further, any data preprocessing steps must be made outside of this application.

3.2 Sample Data

All data shown in this tutorial come from the 14th and 15th rounds of an aptamer selection against HIV-1 reverse transcriptase (Burke DH 1996; Ditzler MA 2013; Whatley AS 2013). These data are preprocessed (trimmed and filtered) and available from http://burkelab.missouri.edu/fastapamer.html.

3.3 FASTAptameR-Count

3.3.1 Description

FASTAptameR-Count serves as the entry point into this suite of modules, and, thus, it should be run prior to any of the following modules. This function accepts either a FASTQ or FASTA file and returns a *counted* data table that can be downloaded as a FASTA or CSV file.

Input FASTQ files should be properly formatted (4 lines per entry with the 2nd line of each entry being the sequence). Input FASTA files are not required to have sequence identifiers. No pre-existing sequence identifiers will be conserved by this module.

3.3.2 Usage

The input FASTA/Q file must be chosen with the file browser or linked in the text box. A sample link is already provided in the text box. Note, files chosen via the file browser take precedence over those linked in the text box.

The `Start` button will begin the counting process. The results will be displayed as a data table on the right side of the screen. For file uploads, please wait for the loading bar to show *Upload complete* before using the `Start` button. A sample data table is shown here:

Table 2: Sample FASTAptameR-Count Output

id	Rank	Reads	RPM	Length	seqs
>1-417696-193358.44	1	417696	193358.44	70	ACGTTG...
>2-313312-145037.35	2	313312	145037.35	70	CATAGC...
>3-174096-80591.94	3	174096	80591.94	70	AACCGC...
>4-94978-43966.9	4	94978	43966.90	70	CATAGC...
>5-74389-34435.91	5	74389	34435.91	70	ACGTTG...
>6-57625-26675.57	6	57625	26675.57	69	CCCTCC...

Note that the *id* column has the following format: `>Rank-Reads-RPM`, where `Rank` is the order of sequences after sorting by `Reads`, which is the raw abundance of each sequence. `RPM` - Reads per Million - is the value of `Reads`, normalized by the total population size: `RPM = Reads / (populationSize / 1e6)`.

The total number of sequences and the number of unique sequences will be displayed below the `Start` and `Download` buttons after running is finished. The `Download` button opens a file browser prior to downloading the output as a FASTA or CSV file (`DEFAULT = FASTA`, which is required for subsequent modules).

3.3.3 Plotting

This module can also generate 2 types of interactive plots based on the counted data: a line plot of reads-per-rank (**Fig. 2**) and a histogram of sequence lengths (**Fig. 3**). The line plot is filterable by 1) minimum number of reads to plot and 2) maximum rank to plot. Both values are chosen with a slider bar. The histogram is not filterable.

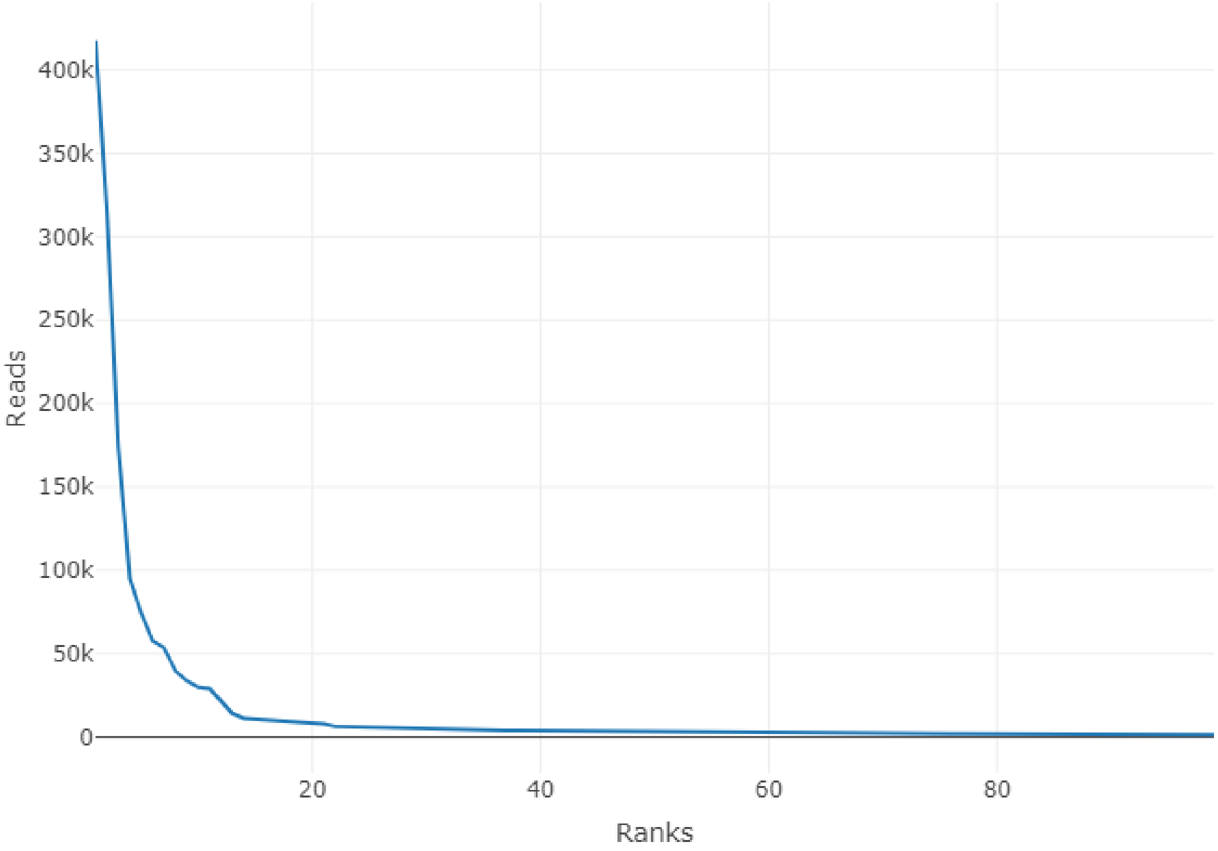

Figure 2: Reads-per-Rank Line Plot

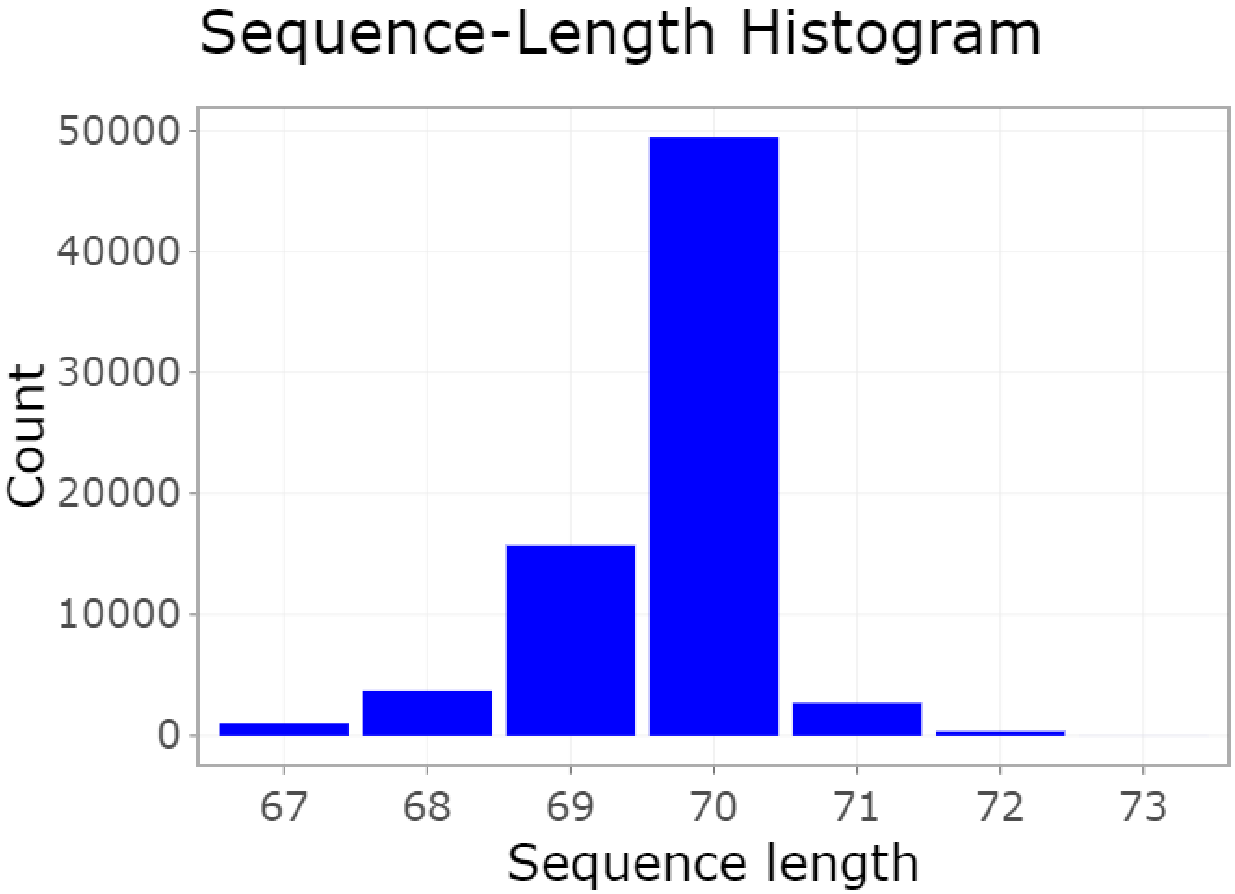

Figure 3: Sequence-Length Histogram

3.4 FASTAptameR-Translate

3.4.1 Description

FASTAptameR-Translate is intended to translate nucleotide sequences to amino acid sequences. This module accepts a *counted* FASTA file and returns a *translated* data table that can be downloaded as a FASTA or CSV file.

3.4.2 Usage

The input FASTA file must be chosen with the file browser. The open reading frame may be selected by the 1st set of radio buttons (`DEFAULT = 1`). The 2nd set of radio buttons indicates whether non-unique sequences should be merged (`DEFAULT = Yes`). If `Yes`, then redundant amino acid sequences are converged, and a new column (`Unique.Nt.Count`) will specify how many non-unique nucleotide sequences from the *counted* input were merged into each amino acid sequence.

The `Start` button begins the translation process. The *translated* data table will be shown on the right side of the screen. The `Download` button opens a file browser prior to downloading the output as a FASTA or CSV file (`DEFAULT = FASTA`, which is required for subsequent modules).

3.4.3 Plotting

This module generates the same two plots as FASTAptameR-Count. See that section for more details.

3.5 FASTAptameR-Motif_Search

3.5.1 Description

FASTAptameR-Motif_Search accepts a *counted* FASTA file and returns a *searched* data table that can be downloaded as a FASTA or CSV file. Sequences in the output must have at least 1 occurrence of each pattern or at least 1 occurrence of at least 1 pattern (see details below for the `partial match` radio button).

3.5.2 Usage

The input FASTA file must be chosen with the file browser. The following text box must contain at least 1 pattern, and multiple patterns must be separated by commas (*e.g.*, `AAA` or `AAA,GTG`). The 1st set of radio buttons determines whether the output has parentheses set around identified patterns (`DEFAULT = No`). Note, this is unable to set parentheses around overlapping patterns (*e.g.*, when `pattern = AGGC,GGCT` and `sequence = AAAGGCT`, the output is `AA(AGGC)T`).

Note that parentheses will be treated as individual characters by subsequent modules and may alter downstream analyses.

The 2nd set of radio buttons determines whether partial filtering - a Boolean `OR` function - is used for multiple pattern matching (`DEFAULT = No`). If `Yes`, filtered sequences must have at least 1 occurrence of **any** listed pattern. If `No`, filtered sequences must have at least 1 occurrence of **each** listed pattern.

The 3rd set of radio buttons determines the type of pattern (`DEFAULT = Nucleotide`). If `Nucleotide`, then degenerate nucleotide codes are allowed, and T/U are interchangeable. Note, all patterns are converted to uppercase and have white spaces removed regardless of the pattern type.

1. **A/T/G/C/U** - single bases
2. **R** - puRine (A/G)
3. **Y** - pYrimidine (C/T)
4. **W** - Weak (A/T)
5. **S** - Strong (G/C)
6. **M** - aMino (A/C)
7. **K** - Keto (G/T)
8. **B** - not A
9. **D** - not C
10. **H** - not G
11. **V** - not T/U
12. **N** - aNy base (not *gap*)

The `Start` button begins the search process. The *searched* data table will be shown on the right side of the screen. The `Download` button opens a file browser prior to downloading the output as a FASTA or CSV file (`DEFAULT = FASTA`, which is required for subsequent modules).

A sample data table is shown here with the following parameters: `comma-separated patterns = CGT`, `parentheses = Yes`, `partial filtering = No`, and `pattern type = Nucleotide`.

Table 3: Sample FASTAptameR-Motif_Search Output

id	Rank	Reads	RPM	seqs
>1-417696-193358.44	1	417696	193358.44	A(CGT)...
>5-74389-34435.91	5	74389	34435.91	A(CGT)...
>6-57625-26675.57	6	57625	26675.57	CCCTCC...
>7-53608-24816.04	7	53608	24816.04	A(CGT)...
>10-29794-13792.14	10	29794	13792.14	GCGAAC...
>11-29062-13453.28	11	29062	13453.28	A(CGT)...

3.6 FASTAptameR-Motif_Enrich

3.6.1 Description

FASTAptameR-Motif_Enrichment accepts 2-3 *counted* FASTA files and returns a data table of metadata related to a **single** pattern's enrichment across multiple populations. Columns of the data table include the following:

1. Input file names
2. Number of unique reads
3. Total number of reads
4. Total number of reads with motif
5. Total number of motif occurrences
6. Total motif RPM

This output can be downloaded as a CSV file and will include appended enrichment scores.

3.6.2 Usage

The input FASTA files must be chosen with the file browser. The following text box must contain a single pattern (**not** comma-separated). The set of radio buttons determines the type of pattern (`DEFAULT = Nucleotide`). If `Nucleotide`, then degenerate nucleotide codes are allowed. Note, the pattern is converted to uppercase and has white space removed regardless of the pattern type.

The `Start` button begins the motif enrichment process. The resulting data table will be shown on the right side of the screen. The `Download` button opens a file browser prior to downloading the output as a CSV file. The enrichment scores (`RPM of 2nd file divided by RPM of 1st file, etc.`) are shown below these buttons. These scores will be included in the downloaded CSV file.

A sample data table (*without* enrichment scores) is shown here with the following pattern: `AAAA`.

Table 4: Sample FASTAptameR-Motif_Enrichment Output

Files	Unique Reads	Total Reads	Seqs with Motif	Motif Occurrences	Motif RPM
70HRT14-count.fasta	72921	2160216	798574	811257	375544.4
70HRT15-count.fasta	62444	1987867	1023154	1038932	522636.6

3.7 FASTAptameR-Distance

3.7.1 Description

FASTAptameR-Distance accepts a *counted* FASTA file as input and returns a data table that contains a column for the Levenshtein edit distance between each input sequence and a query sequence. The output can be downloaded as a CSV.

3.7.2 Usage

The input FASTA file must be chosen with the file browser, and the following text box must contain a single query sequence. Note, this query sequence may not have any degenerate nucleotide codes. The `Start` button begins the distance calculations. The resulting data table will be shown on the right side of the screen. The `Download` button opens a file browser prior to downloading the output as a CSV file.

A sample data table is shown here with the following query sequence:

`CTCTTACCCAAGACTGATCCGAAGGCAACGGGACAAAAGGCAAGAGCGCGATACCAATGCTGGACTG`.

Table 5: Sample FASTAptameR-Distance Output

seqs	Rank	Reads	RPM	Distance
CTCTTA...	1274	63	29.16	0
CTCTTA...	14083	3	1.39	1
CCCTTA...	21230	2	0.93	1
CTCTTA...	21539	2	0.93	1
CTCTTA...	21540	2	0.93	1
CTCTTA...	21543	2	0.93	1

3.7.3 Plotting

This module can also generate an interactive histogram of distances (**Fig. 4**).

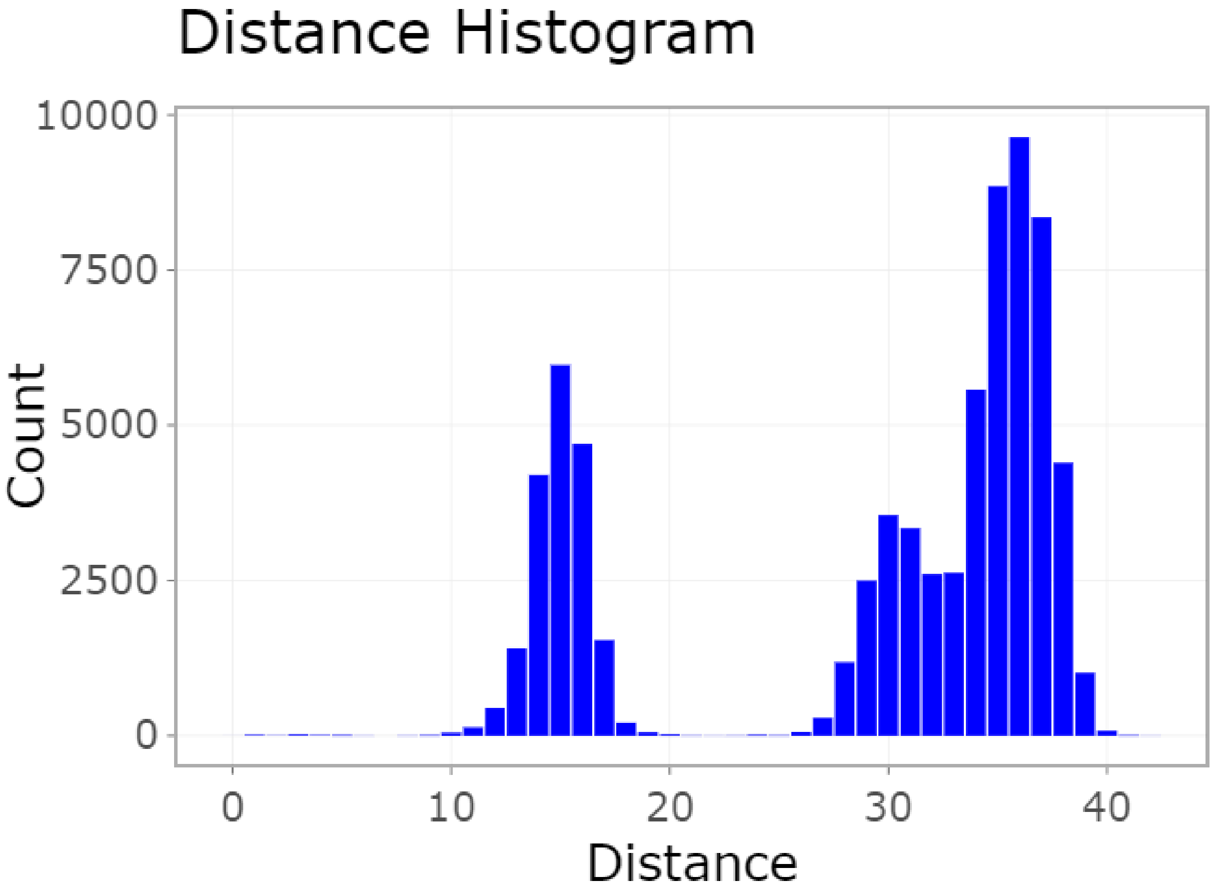

Figure 4: Distance Histogram

3.8 FASTAptameR-Enrich

3.8.1 Description

FASTAptameR-Enrich accepts 2-3 *counted* FASTA files as input and returns a single data table after merging by sequences. Columns from the 1st, 2nd, and optional 3rd file are appended with *.x*, *.y*, and *.z*, respectively. Additional columns include enrichment scores - `RPM2 / RPM1` - and fold changes - `log2(enrichment)`. The order of comparisons are `y:x`, `z:y`, and `z:x`. This output can be downloaded as a CSV.

3.8.2 Usage

The input FASTA files must be chosen with the file browser. The following set of radio buttons determines whether missing values are allowed in the output. Missing values result from sequences that are not present in all input files.

The `Start` button begins the enrichment calculations, and the resulting data table will be shown on the right side of the screen. All numeric columns in this data table are filterable by typing into the corresponding text box (*e.g.*, `1 ... 10` to keep values in the range `[1:10]`) or by using the slider bar that is displayed after clicking in the corresponding text box.

The `Download` button opens a file browser prior to downloading the output as a CSV file.

A sample data table is shown here:

Table 6: Sample FASTAptameR-Enrich Output

seqs	Rank.x	Reads.x	RPM.x	Rank.y	Reads.y	RPM.y	foldChange_yx	enrichment_yx
ACGTTG...	1	417696	193358.44	3	161830	81408.87	-1.25	0.42
CATAGC...	2	313312	145037.35	1	382391	192362.47	0.41	1.33
AACCGC...	3	174096	80591.94	5	104932	52786.23	-0.61	0.66
CATAGC...	4	94978	43966.90	6	42954	21608.09	-1.02	0.49
ACGTTG...	5	74389	34435.91	9	32821	16510.66	-1.06	0.48
CCCTCC...	6	57625	26675.57	7	37701	18965.55	-0.49	0.71

3.8.3 Plotting

This module can also generate 3 types of interactive plots: fold-change histograms (1 per comparison - **Fig. 5**), RPM scatter plots (2D plot for 2 populations - **Fig. 6** - or 3D plot for 3 populations), and volcano plots (1 per comparison - **Fig. 7**).

The spread of the fold-change histogram can indicate the magnitudes of enrichment and possible directionality.

Similarly, the spread of the RPM scatter plot can also indicate the magnitudes of enrichment and possible directionality.

The volcano plot is used to show the relationship between the respective fold change and number of reads for each sequence. Note, the y-axis is given by $y(seq) = \sqrt{\log_{10}(seq.Reads)/(log_{10}Total.Reads})$.

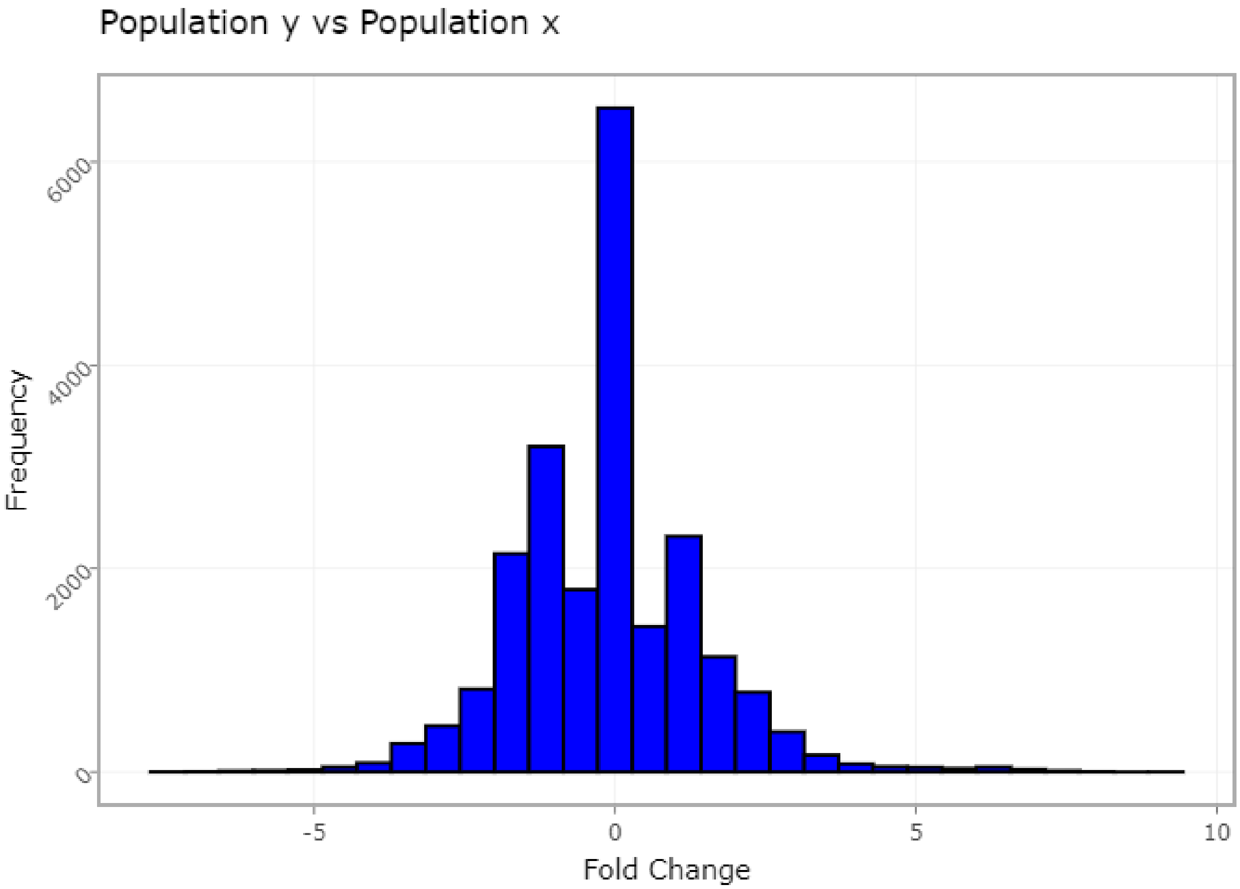

Figure 5: Fold-Change Histogram

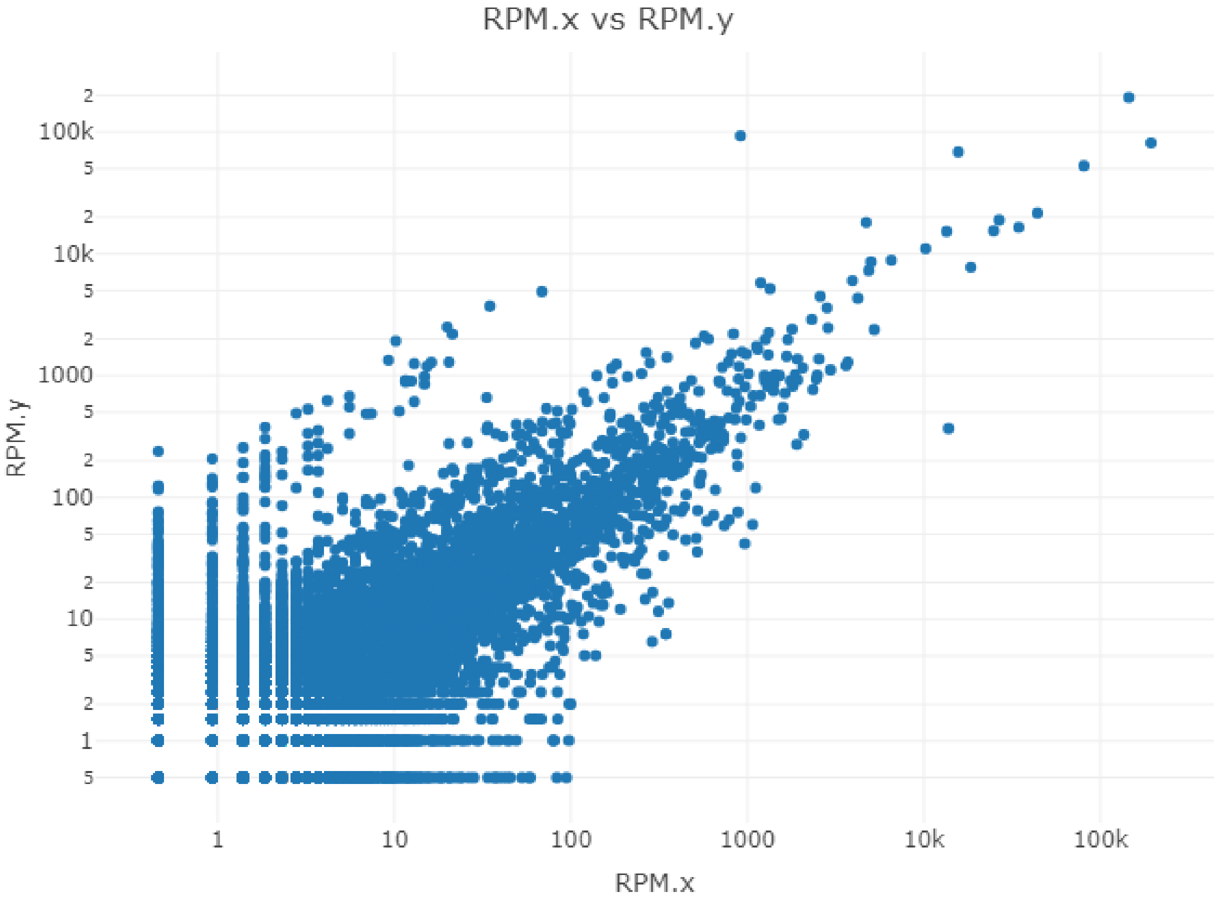

Figure 6: RPM Scatter Plot

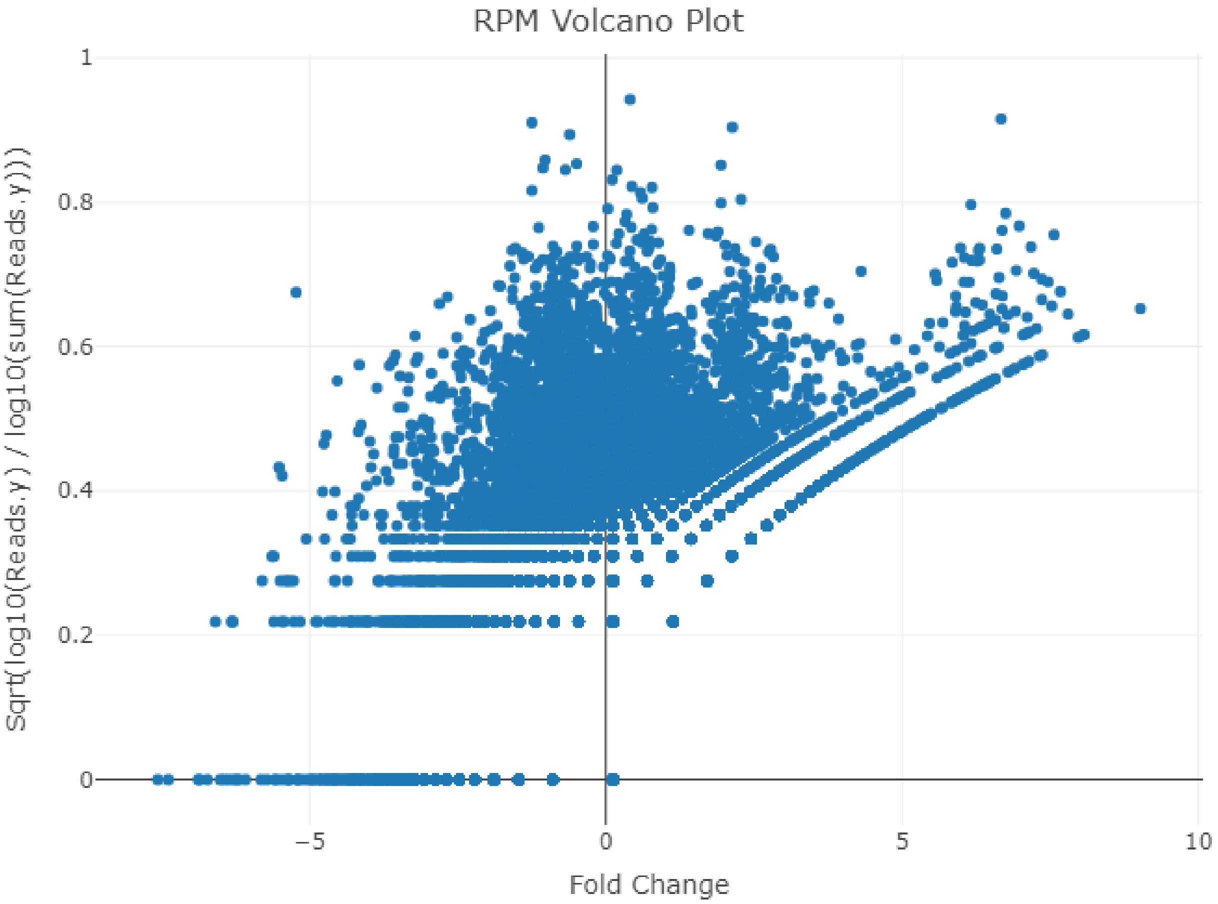

Figure 7: Volcano Plot

3.9 FASTAptameR-Cluster

3.9.1 Description

FASTAptameR-Cluster accepts a *counted* FASTA file as input and returns a *clustered* data table if no output directory is specified. This data table contains all sequences / clusters and can be downloaded as a single FASTA or CSV file. If an output directory is specified, then no data table will be created, and 1 FASTA file per cluster will be written to the output directory.

Briefly, the most abundant, non-clustered sequence becomes a cluster seed. Any other sequence within a predefined edit distance is added to this cluster. This process repeats until all sequences are clustered or a predefined number of clusters is created.

3.9.2 Usage

The input FASTA file must be chosen with the file browser. The 1st slider bar sets the minimum number of reads a sequence must have to be clustered (`DEFAULT = 10`). Sequences with fewer reads are removed prior to clustering. The 2nd slider bar sets the maximum Levenshtein edit distance to consider between a seed sequence and all other sequences (`DEFAULT = 7`). The 3rd slider bar sets the total number of desired clusters (`DEFAULT = 20`). Note, any remaining sequences will be grouped as `NC` ("not clustered").

The subsequent radio buttons indicate if each cluster should be written to a different FASTA file (`DEFAULT = No`). If `No`, then all clusters are grouped together and downloaded in a single file. If `Yes`, then each cluster will be written to its own FASTA file, and no data table will be displayed. Note, a directory path must be specified if this option is `Yes`.

The `Start` button will begin the clustering process. The results will be displayed as a data table on the right side of the screen. The `Download` button opens a file browser prior to downloading the output as a FASTA or CSV file (`DEFAULT = FASTA`, which is required for subsequent modules). Algorithm progress will be shown below these buttons and will update after each cluster finishes.

A sample data table is shown here:

Table 7: Sample FASTAptameR-Cluster Output

id	Rank	Reads	RPM	cluster	rankInCluster	LED	seqs
>1-417696-193358.44-1-1-0	1	417696	193358.44	1	1	0	ACGTTG...
>2-313312-145037.35-2-1-0	2	313312	145037.35	2	1	0	CATAGC...
>3-174096-80591.94-3-1-0	3	174096	80591.94	3	1	0	AACCGC...
>4-94978-43966.9-2-2-1	4	94978	43966.90	2	2	1	CATAGC...
>5-74389-34435.91-1-2-1	5	74389	34435.91	1	2	1	ACGTTG...
>6-57625-26675.57-4-1-0	6	57625	26675.57	4	1	0	CCCTCC...

Note that the new *id* column is the old *id* with `Cluster Number`, `Rank in Cluster`, and `Distance to Cluster Seed` appended to the end.

3.10 FASTAptameR-Cluster_Analysis

3.10.1 Description

FASTAptameR-Cluster_Analysis accepts a *clustered* FASTA file as input and returns a data table with metadata for each cluster. This data table can be downloaded as a CSV file.

3.10.2 Usage

The input FASTA file must be chosen with the file browser. The `Start` button begins the analysis process. The results will be displayed as a data table on the right side of the screen and include the following columns: `Cluster Number`, `Seed Sequence`, `Total Sequences`, `Total Reads`, and `Total RPM`. The `Download` button opens a file browser prior to downloading the output as a CSV file, which can be used by FASTAptameR-Cluster_Enrich.

A sample data table is shown here:

Table 8: Sample FASTAptameR-Cluster_Analysis Output

Cluster	Seeds	TotalSequences	TotalReads	TotalRPM
1	ACGTTG...	1259	770383	356623.54
2	CATAGC...	1295	652468	302038.75
3	AACCGC...	528	257675	119282.23
4	CCCTCC...	324	101448	46962.14
5	AGCGCG...	262	73809	34167.47
6	TTGACA...	275	67908	31435.87

3.10.3 Plotting

This module is also able to analyze clusters by converting all sequences into k-mer vectors (`k=4`) and rendering an interactive 2D PCA plot, colored by cluster (**Fig. 8**).

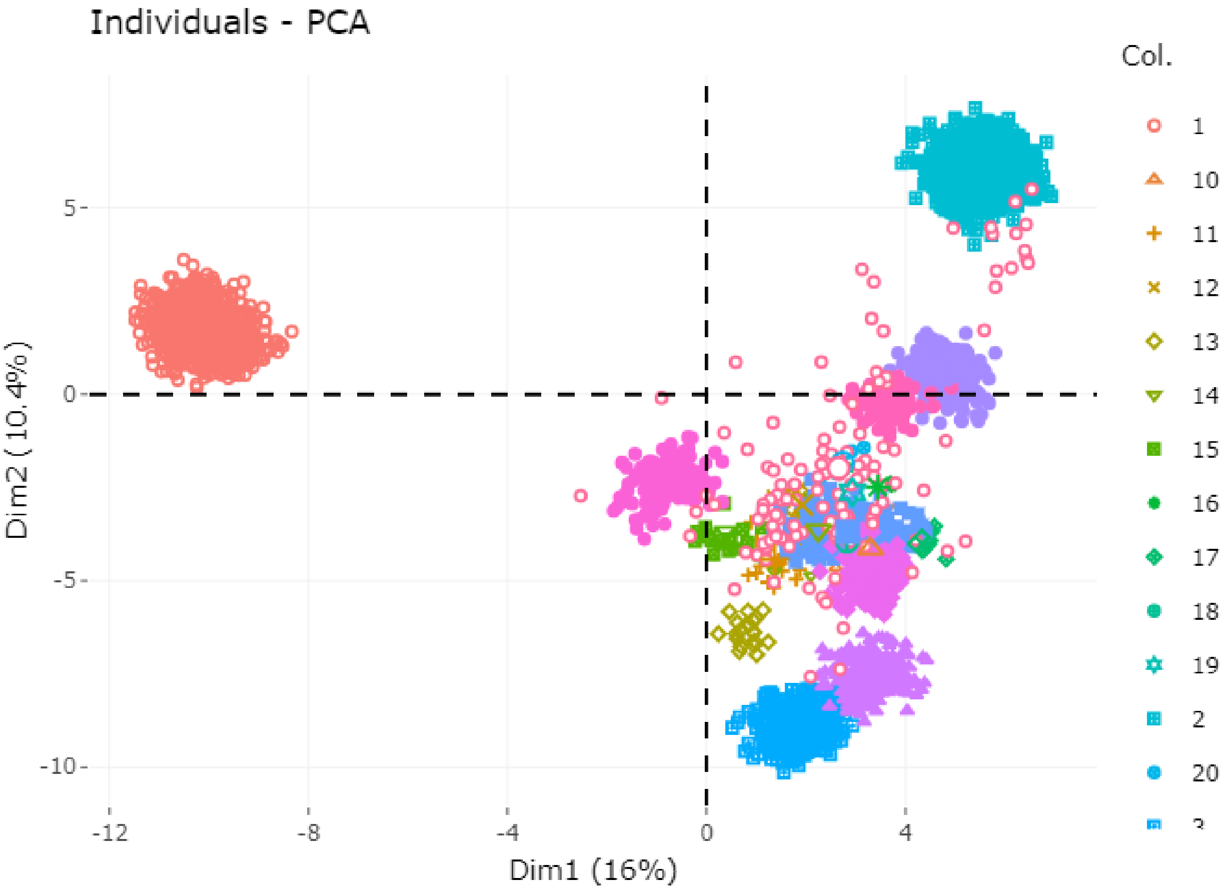

Figure 8: kmer PCA

3.11 FASTAptameR-Cluster_Enrich

3.11.1 Description

FASTAptameR-Cluster_Enrich accepts 2-3 *cluster-analysis* CSV files as input and returns a data table after merging by `Seed Sequence`. This data table includes a column for `Enrichment` can be downloaded as a CSV file.

3.11.2 Usage

The input FASTA files must be chosen with the file browser. The `Start` button begins the enrichment calculation. The results will be displayed as a data table on the right side of the screen and include a column for `Enrichment`. The `Download` button opens a file browser prior to downloading the output as a CSV file.

A sample data table is shown here:

Table 9: Sample FASTAptameR-Cluster_Enrichment Output

Seed	Cluster.x	Seqs.x	Reads.x	RPM.x	Cluster.y	Seqs.y	Reads.y	RPM.y	Enrichment_yx
ACGTTG...	1	1259	770383	356623.54	3	668	315492	158708.77	0.45
CATAGC...	2	1295	652468	302038.75	1	1261	685328	344755.34	1.14
AACCGC...	3	528	257675	119282.23	5	326	152680	76805.85	0.64
CCCTCC...	4	324	101448	46962.14	6	167	60085	30225.89	0.64
AGCGCG...	5	262	73809	34167.47	8	139	27584	13876.15	0.41
TTGACA...	6	275	67908	31435.87	4	646	276605	139146.49	4.43

Note that columns 3-5 and 7-9 refer to *total* values in the given cluster.

4 Version history

References

Alam KK, Burke DH, Chang JL. 2015. “FASTAptamer: A Bioinformatic Toolkit for High-throughput Sequence Analysis of Combinatorial Selections.” *Mol Ther Nucleic Acids* 4. https://doi.org/10.1038/mtna.2015.4.

Burke DH, Andrews K, Scates L. 1996. “Bent pseudoknots and novel RNA inhibitors of type 1 human immunodeficiency virus (HIV-1) reverse transcriptase.” *J Mol Biol* 264. https://doi.org/10.1006/jmbi.1996.0667.

Ditzler MA, Bose D, Lange MJ. 2013. “High-throughput sequence analysis reveals structural diversity and improved potency among RNA inhibitors of HIV reverse transcriptase.” *Nucleic Acids Res* 41. https://doi.org/10.1093/nar/gks1190.

Whatley AS, Lange MJ, Ditzler MA. 2013. “Potent Inhibition of HIV-1 Reverse Transcriptase and Replication by Nonpseudoknot, ‘UCAA-motif’ RNA Aptamers.” *Mol Ther Nucleic Acids* 2. https://doi.org/10.1038/mtna.2012.62.