Identification of targets for the microRNA miR-4728-3p in ribosome and polysome sequencing data

Author: Euisuk Han
Date: 27/05/23
GitHub Repository: https://github.com/robinyn/miR-4728

The links to scripts in the text will only be functional when viewing the README in the above GitHub repository.

Data

The data for this project consists of ribosome and polysome profiling data from SK-BR-3 (HER2-positive breast cancer cell line) transfected with antisense oligonucleotide (ASO) to block the miRNA with matched total RNA-seq. Control samples were transfected with scrambled ASO. There are three replicates per condition.

Analysis environment

A conda environment was created for this analysis with the following tools installed.

- samtools (v1.6)
- htseq (v2.0.2)
- fastqc (v0.12.1)
- trimmomatic (v0.39)

The following R(v4.3.2) packages were also installed:

- tidyverse(v2.0.0)
    - stringr(v1.5.0)
    - ggplot2(v3.4.2)
- anota2seq (v1.20.0)
- clusterProfiler (v4.6.2)
- DESeq2 (v1.38.3)
- AnnotationDbi (v1.60.2)
- org.Hs.eg.db (v3.16.0)
- biomaRt (v2.54.1)
- msigdbr (v7.5.1)
- edgeR (v3.40.2)
- limma (v3.54.2)

```
# Create conda environment
conda create -n mir-4728 python=3.7.12

# Activate environment
conda activate mir-4728
```

```
# Install required tools
conda install -c bioconda -c conda-forge samtools=1.6 htseq=2.0.2
fastqc=0.12.1 trimmomatic=0.39

# Install R and required packages
conda install -c bioconda -c conda-forge \
r-base=4.2.3                          \
r-tidyverse=2.0.0                     \
bioconductor-anota2seq=1.20.0         \
bioconductor-clusterProfiler=4.6.2    \
bioconductor-DESeq2=1.38.3            \
bioconductor-AnnotationDbi=1.60.2     \
bioconductor-org.Hs.eg.db=3.16.0      \
bioconductor-biomaRt=2.54.1           \
bioconductor-msigdbr=7.5.1            \
bioconductor-limma=3.54.2             \
bioconductor-edgeR=3.40.2             \

# Start R console
R

# The R scripts found in the GitHub repository can now be run from the R
console
source("directory/of/script/and/name.R")
```

If any of the R packages fail to install through conda, they can be installed manually in the R console (command-line or RStudio) with the following command:

```
# For non-bioconductor packages
install.packages("Name_of_the_package")

# For bioconductor packages install BiocManager
if (!require("BiocManager", quietly = TRUE))
    install.packages("BiocManager")

# Install packages through BiocManager
BiocManager::install("Name_of_the_package")
```

The following tools were not available through Conda and were installed manually following the installation steps provided in their respective documentations:

- HISAT2 (v2.2.1)(available at http://daehwankimlab.github.io/hisat2/)
- Novoalign (v4.03.07)(available at https://www.novocraft.com/products/novoalign/)

0. Quality control

Both datasets were tested using FastQC to ensure that all sample files were suitable for use in downstream analyses.

0.1. Polysome profiling data

```
# Generate a list of files and their directories to be checked
ls ~/directory/with/data/files | while read line; do echo
~/directory/of/data/file/$line; done > files_list.txt

# Run FastQC
cat files_list.txt | while read file; do fastqc -o 0_fastqc --noextract
$file
```

0.2. Ribosome profiling data


```
# Generate a list of files and their directories to be checked
ls ~/directory/with/data/files | while read line; do echo
~/directory/of/data/file/$line; done > files_list.txt

# Run FastQC
cat files_list.txt | while read file; do fastqc -o 0_fastqc --noextract
$file
```


The QC results were parsed using a custom python script and visualized using an R script.

1. Alignment of reads

1.1. Polysome profiling data

1.1.1. Building index

The GRCh38 assembly of the human genome from the Genome Reference Consortium and the Gencode 43 release of the human transcriptome annotation (Ensembl release 109) was used to build a reference index for the alignment, following the protocol provided in the HISAT2 documentation.

Unfortunately, the indexing of a human genome with transcript annotations required over 160GB of memory and could not be completed. Therefore, a prebuilt index used previously for a SCANB project was acquired and used. The index was created using GRCh38 assembly, gencode v41 annotation for the transcriptome annotation, and the dbSNP build 155 for the SNP data.

1.1.2. Alignment

The reads from the polysome profiling were aligned to the reference index, using HISAT2. The alignment output was directly piped into samtools to sort and convert them from SAM to BAM formats.

```
cat files_list.txt | while read line;do file_dir=$(echo ../data/$line);
file_name=$(echo $line | cut -d "/" -f 2); file_folder=$(echo $line | cut
-d "/" -f 1); mate_1=$(echo ${file_dir}_R1_001.fastq.gz); mate_2=$(echo
${file_dir}_R2_001.fastq.gz); echo $file_name; hisat2 -p 15 -q --fr --new-
summary --summary-file 1_alignments/summary/$file_name.sam.summary --dta -
-rna-strandness RF --non-deterministic --max-intronlen 2000000 -x
```

```
../reference/hisat2/genome_snp_tran -1 $mate_1 -2 $mate2 | samtools sort -
o 1_alignments/$file_folder/$file_name.bam; done;
```

1.1.3. Summary statistics from alignment

A list of summary files and their directories was generated using bash.

```
ls directory/with/summary/files | while read file; do echo
summary_files/$file; done > sample_list.txt
```

A custom Python script was used to parse the summary files with the list and visualized using an R script.

1.2. Ribosome profiling data

1.2.1. Building the index

While Novoalign includes the protocol for generating an index with known transcripts, the protocol was extremely long and poorly documented. It was decided that the benefits of attempting to create an index with transcriptome annotation was not great enough to outweigh the time and resources it would require. Therefore, an index was created with only the genome assembly. The same GRCh38 genome assembly used for the HISAT2 index was used.

```
# Index is created
novoindex ~/reference/novoalign/GRCh38_no_alt_maskedGRC.nix
~/reference/raw/GCA_000001405.15_GRCh38_no_alt_analysis_set_maskedGRC_excl
usions_v2.fasta

# A soft link to the index was created in the directory where novoalign
will be run
ln -s ~/reference/novoalign/GRCh38_no_alt_maskedGRC.nix
~/ribosome/index_link.nix
```

In addition to the reference genome, another index was created for the complete human ribosomal DNA repeating units (U13369.1).

```
# Index is created
novoindex ~/reference/novoalign/rRNA.nix
~/reference/raw/human_complete_rRNA.fasta

# A soft link to the index is created
ln -s ~/reference/novoalign/rRNA.nix ~/ribosome/rRNA_link.nix
```

1.2.2. Alignment with Novoalign

Novoalign was run using the index file created above. The alignment parameters were set to be extra sensitive compared to regular RNA seq alignments, due to the short read lengths of the RPFs. Novoalign also allows the clipping of adapter sequences that may be present in the 3' ends of the reads. Therefore, the adapter sequence used for the Ribo-seq procedure was provided to the aligner.

```
ls data | while read file; do sample_name=$(echo $file | sed
"s/.fastq.gz//"); echo $file; novoalign -c 15 -d index_link.nix -f
data/$file -F STDFQ -a AGATCGGAAGAGCACACGTCT -l 17 -h -1 -1 -t 90 -g 50 -x
15 -o SAM -o FullNW -r All 51 -e 51 2>
1_alignments/novoalign/genome/$file.summary | samtools sort -o
1_alignments/novoalign/genome/$sample_name.bam; done;
```

The ribosome dataset was aligned again to the rDNA repeats to determine the approximate levels of rRNA still remaining in the data. It was determined that while it is probably not necessary to remove such reads mapping as rRNA, it would still be beneficial to see how successful the rRNA depletion step was. '

```
ls data | while read file; do sample_name=$(echo $file | sed
"s/.fastq.gz//"); echo $file; novoalign -c 15 -d rRNA_link.nix -f
data/$file -F STDFQ -a AGATCGGAAGAGCACACGTCT -l 17 -h -1 -1 -t 90 -g 50 -x
15 -o SAM -o FullNW -r All 51 -e 51 2>
1_alignments/novoalign/rRNA/$file.summary | samtools sort -o
1_alignments/novoalign/rRNA/$sample_name.bam; done;
```

The alignment results were parsed with a python script and visualized using an R script

1.2.3. Alignment with HISAT2

Another set of alignments were produced using HISAT2 in order to compare the two mapping softwares. It was believed that Novoalign would be better for the alignment of the Ribo-Seq dataset due to the short read lengths.

Trimming

Before the alignment could be performed, the adapter sequences were trimmed as HISAT2 is not capable of adapter trimming on its own, unlike Novoalign.

```
ls data | while read file; do sample_name=$(echo $file|sed
"s/.1.fastq.gz//"); echo $file; trimmomatic SE -threads 15 -phred33 -
summary 0_trimming/$sample_name.trim.summary data/$file
0_trimming/$sample_name.trimmed.fastq
ILLUMINACLIP:$HOME/bin/miniconda3/pkgs/trimmomatic-0.39-
hdfd78af_2/share/trimmomatic-0.39-2/adapters/TruSeq3-SE.fa:2:30:10
SLIDINGWINDOW:4:25 MINLEN 17; done;
```

The trimmed sequences were passed to FastQC to evaluate the trimming results.

```
ls 0_trimming/ | grep -v ".summary" | while read file; do echo $file;
fastqc -o 0_fastqc/trimmed/ --noextract 0_trimming/$file; done;
```

The trimming results were parsed using a python script and visualized using an R script.

Alignment

The trimmed sequences were then aligned to the reference genome.

```
ls data | while read file; do sample_name=$(echo $file | sed
"s/.1.fastq.gz//"); echo $file; hisat2 -p 15 -q --phred33 --new-summary --
summary-file 1_alignments/hisat2/summary/$sample_name.sam.summary --dta --
rna-strandness R --non-deterministic --max-intronlen 2000000 -x
../reference/hisat2/genome_snp_tran -U data/$file | samtools sort -o
1_alignments/hisat2/$file_folder/$sample_name.bam; done;
```

NOTE: The alignments with HISAT2 were extremely poor (<1% alignment).

2. Gene counts

2.1. Polysome data

2.1.1. Read counts of polysome data with HTSeq-count

```
# Generate a list of alignment (BAM) files
ls directory/with/alignment/files | while read file; do echo
directory/to/files/$file; done > alignments_list.txt

# Run HTSeq-count
files=$(cat alignments_list.txt | awk '{print}' ORS=" "); htseq-count -f
bam -r pos -s reverse -t exon -m intersection-strict --nonunique=all
$files
../reference/raw/gencode.v41.primary_assembly.annotation.ucsc.filtered.gtf
```

2.2. Ribosome data

2.2.1. Read counts of ribosome data with HTSeq-count

```
# Generate a list of alignment (BAM) files
ls directory/with/alignment/files | while read file; do echo
directory/to/files/$file; done > alignments_list.txt

# Run HTSeq-count
files=$(cat alignments_file.txt | awk '{print}' ORS=" "); htseq-count -f
bam -r pos -s yes -t exon -m intersection-strict --nonunique=all $files
```

```
../reference/raw/gencode.v41.primary_assembly.annotation.ucsc.filtered.gtf
> gene_counts_no.txt
```

3. Differential expression analysis

3.1. Polysome data

The R package anota2seq was used for the differential expression analysis of the polysome data, generating log2FC values for the total mRNA and translated (polysome associated) mRNA.

An R script was used for the analysis.

In order to make the polysome data comparable to the ribosome data, the analysis was conducted again using DESeq2.

A different R script was used for the DESeq2 (deltaTE) analysis.

3.2. Ribosome data

Anota2seq could not be used for the analysis of the Ribo-seq data, as it requires at least 3 replicates per condition and one of the samples was removed due to a significantly low total number of reads. Therefore, a similar analysis package, DESeq2, was used.

An R script was used for the DESeq2 (deltaTE) analysis.

4. Gene set enrichment analysis (GSEA)

The R package clusterProfiler was used to carry out a GSEA against genesets from KEGG and Gene Ontology, as well as transcription factor target genes, TargetScan predicted targets for miR-4728-3p/miR-21-5p, 5' terminal oligopyrimidine (5'TOP) genes and genes with internal ribosomal entry sites (IRES).

Before the GSEA could be run, gene sets not available through the msigdbr package were created with the following data using an R script:

```
- TargetScan miRNA target predictions downloaded from [TargetScan]
([https://](https://www.targetscan.org/vert_80/)).
- 5'TOP genes taken from the Supplementary Materials of [Cottrell
et.al.,2020](https://www.nature.com/articles/s41598-020-79379-8#Sec20).
- IRES genes downloaded from [IRESbase](http://reprod.njmu.edu.cn/cgi-
bin/iresbase/index.php)
- UniBind transcription factor binding site data downloaded from [UniBind]
(https://unibind.uio.no)
```

The UniBind data only provided chromosomal coordinates of the TFBSs, which had to be converted to Ensembl gene IDs using a Python script

The enrichment analysis were run with an R script and visualized using another script