Bioinformatics, 2022, pp. 1–11

doi: https://www.doi.org/
Advance Access Publication Date: 2025
BMIF 898 MASTER'S PROJECT SUPERVISED BY DR. MARK ORMISTON (FINAL REPORT)

BMIF 898 MASTER'S PROJECT SUPERVISED BY DR. MARK ORMISTON (FINAL REPORT)

Development of a Bioinformatics Pipeline for Ribosome Profiling (Ribo-Seq) Data Analysis

Amir Ranjbar[1,*]

[1]School of Computing and Department of Biomedical and Molecular Sciences, Queen's University, 99 University Ave, K7L 3N6, Ontario, Canada
[*]Corresponding author. amir.ranjbar@queensu.ca

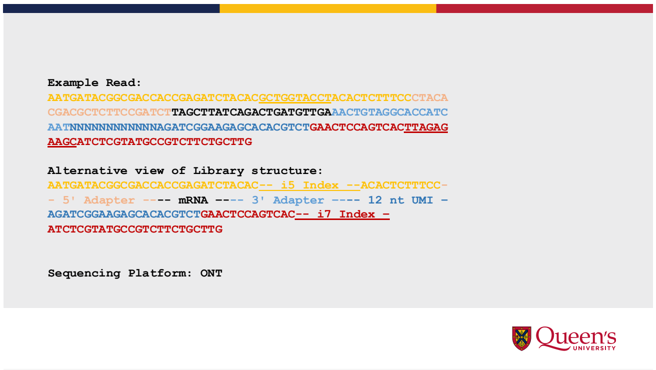
Abstract

A bioinformatics pipeline was developed to assess ribosome profiling (Ribo-Seq) data quality. After filtering and alignment, 101,014 high-quality 30-nucleotide ribosome-protected fragment (RPF) reads were recovered, most of which were mapped to coding regions. Over 22,000 actively translated open reading frames (ORFs) were detected, confirming good library quality and strong triplet periodicity. This pipeline provides a reliable way to process and evaluate Ribo-Seq data.

Key words: Ribosome Profiling, Bioinformatics Pipeline, Data Quality

1. Introduction

Microarray and RNA sequencing are commonly used to measure gene expression by quantifying mRNA levels. However, mRNA abundance doesn't always correlate with protein levels because translation also plays a role in gene regulation (Aeschimann et al., 2015). This means that the amount of protein produced can differ from what is expected based on mRNA alone. To get a more accurate picture of gene expression, it's crucial to measure proteins directly. Although mass spectrometry provides protein data, it doesn't yet capture the full range of proteins in a cell despite advancements in the field (Aeschimann et al., 2015). Ribosome profiling, a newer technique, offers a way to study translational activity with similar precision to RNA sequencing. Ribosome profiling uses deep sequencing of ribosome-protected mRNA fragments to map translation with near-codon-level precision across the transcriptome (Ingolia et al., 2009). It allows for a detailed look at translation but is limited to measuring translation rates, not actual protein levels, which also depend on protein turnover (Aeschimann et al., 2015). Before ribosome profiling, researchers used polysome profiling to examine ribosome distribution on mRNA. However, polysome profiling doesn't provide information about where ribosomes are located on the mRNA, and it can't differentiate between ribosomes translating the main open reading frame (ORF) and those on upstream ORFs (uORFs) (Aeschimann et al., 2015). This is important because ribosomes on uORFs, typically located in 5' UTRs, don't contribute to the main gene product and can even interfere with the translation of the primary ORF (Ingolia et al., 2009; Aeschimann et al., 2015). In some cases, translation termination at uORFs can induce ribosomal stalling, contributing to mRNA decay and further downregulating gene expression (Wethmar, 2014). Ribosome profiling addresses these limitations by giving a more accurate and detailed measure of ribosome occupancy across genes. Since its introduction in 2009, it has been widely adopted across many organisms and experimental systems. Each translating ribosome footprint protects an mRNA segment of roughly 30 nucleotides (More on this is discussed in Section 5.2.), which can be recovered and sequenced (Ingolia et al., 2009). By sequencing and mapping the mRNA regions bound by ribosomes, ribosome profiling offers a comprehensive view of translation at a specific moment in time (Aeschimann et al., 2015).



```
Example Read:
AATGATACGGCGACCACCGAGATCTACACGCTGGTACCTACACTCTTTCCCTACA
CGACGCTCTTCCGATCTTAGCTTATCAGACTGATGTTGAAACTGTAGGCACCATC
AATNNNNNNNNNNNNAGATCGGAAGAGCACACGTCTGAACTCCAGTCACTTAGAG
AAGCATCTCGTATGCCGTCTTCTGCTTG

Alternative view of Library structure:
AATGATACGGCGACCACCGAGATCTACAC-- i5 Index --ACACTCTTTCC-
- 5' Adapter ---- mRNA ---- 3' Adapter ---- 12 nt UMI -
AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC-- i7 Index -
ATCTCGTATGCCGTCTTCTGCTTG


Sequencing Platform: ONT
```

Fig. 1. Sequences Used in Library Preparation Protocol.

2. Materials

This project involved developing a computational analysis pipeline for libraries generated from primary pulmonary endothelial cells using the QIAseq miRNA UDI Library Kit. The libraries were sequenced using Oxford Nanopore Technologies. This pipeline can also be applied in future studies exploring the role of BMPR2 transcripts. By analyzing Ribo-Seq data, this pipeline will help determine how the loss of these transcripts affects mRNA translation. Understanding this regulation may reveal how BMPR2 contributes to pulmonary arterial hypertension (PAH), a severe lung disease linked to abnormal endothelial cell growth.

2.1 Hardware

Linux Environment

2.2 Software

FastQC (v0.12.1)	`https://www.bioinformatics.babraham.ac.uk/projects/download.html`
UMI-tools (v1.0.0)	`https://github.com/CGATOxford/UMI-tools`
Cutadapt (v4.9)	`https://github.com/marcelm/cutadapt`
Bowtie2 (v2.5.4)	`https://github.com/BenLangmead/bowtie2`
STAR (v2.7.11a)	`https://github.com/alexdobin/STAR`
SAMtools (v1.20)	`https://github.com/samtools/samtools`
subread (v2.20.0)	`http://subread.sourceforge.net/`
ribotricer (v1.4.0)	`https://github.com/smithlabcode/ribotricer`
ribotish (v0.2.7)	`https://github.com/zhpn1024/ribotish`

2.3 Input Files

2RW9VV_1.fastq	Raw sequencing file for Ribo-seq
Homo_sapiens_rRNA_tRNA.fa	Combined file containing ribosomal RNA (rRNA) and transfer RNA (tRNA) sequences for Homo sapiens. Ribosomal RNA was downloaded from the Silva database and tRNA sequences were obtained from GtRNAdb.
Homo_sapiens.GRCh38.113.gtf	Gene Transfer Format (GTF) file containing the gene annotations for Homo sapiens (GRCh38 version 113) from Ensembl.
Homo_sapiens.GRCh38.dna.primary_assembly.fa	Primary DNA assembly sequence for Homo sapiens (GRCh38 version) from Ensembl.
Homo_sapiens.GRCh38.ncrna.fa	Transcript sequences of non-coding RNA (ncRNA) genes for Homo sapiens from Ensembl.

3. Methods

This paper focuses on the bioinformatics methods used to assess the quality of ribosome profiling data.

3.1 Protocol

Ribosome profiling involves a few key steps:

- **Sample Collection and Preparation:** Biological samples are collected, and mRNA is digested with RNase I to generate ribosome-protected fragments (RPFs).
- **Purification:** The RPFs are then purified based on size and subjected to end healing repair to allow for adapter ligation and library preparation.
- **Library Preparation:** Using QIAseq mRNA UDI library kit, the RPFs are prepared for sequencing through adapter ligation and amplification.

This process usually takes a few days, with breaks possible at different stages (Ingolia et al., 2012). Once sequencing is done, bioinformatics tools are used to check the data quality (discussed in the next Section).

3.2 Bioinformatic Analysis and Quality Control

The analysis pipeline for ribosome-protected fragments is based on RNA-Seq methods, with specific adjustments for ribosome profiling. The main difference is the use of specialized tools for the analysis and quality control of Ribo-Seq data (Fig. 2).

3.2.1 FastQC

As a first step, before pre-processing, the following command is used to generate basic statistics of our ribo-seq fastq file (such as read length and total reads):

```
fastqc <input_file>
—o <output_directory>
```

3.2.2 Cutadapt

The Cutadapt tool is used for trimming adapter sequences from the reads. It is especially useful for removing unwanted sequences that might interfere with downstream analysis (Martin, 2011). The following command:

```
cutadapt —g <i5_index>
——untrimmed—output
<antisense_output_file>
<input_fastq_file>
```

has the following functions:

- **-g <i5_index>:** This option specifies the adapter sequence (i5 index) that will be searched for at the 5' end of the reads. In this case, the i5 index sequence used is GCTGGTACCT, which is associated with sense reads (Fig. 1).
- **–untrimmed-output <antisense_output_file>:** This option directs Cutadapt to output the reads that do not contain the specified i5 index to a separate file, identifying them as antisense reads. These reads are not trimmed and are stored in the output file specified.

The aim of using this command was to target the i5 index (GCTGGTACCT) to identify reads that lack this index near their 5' ends, which are our antisense reads. Essentially, by using the i5 index associated with sense reads, we were able to separate the antisense reads from the main fastq file.

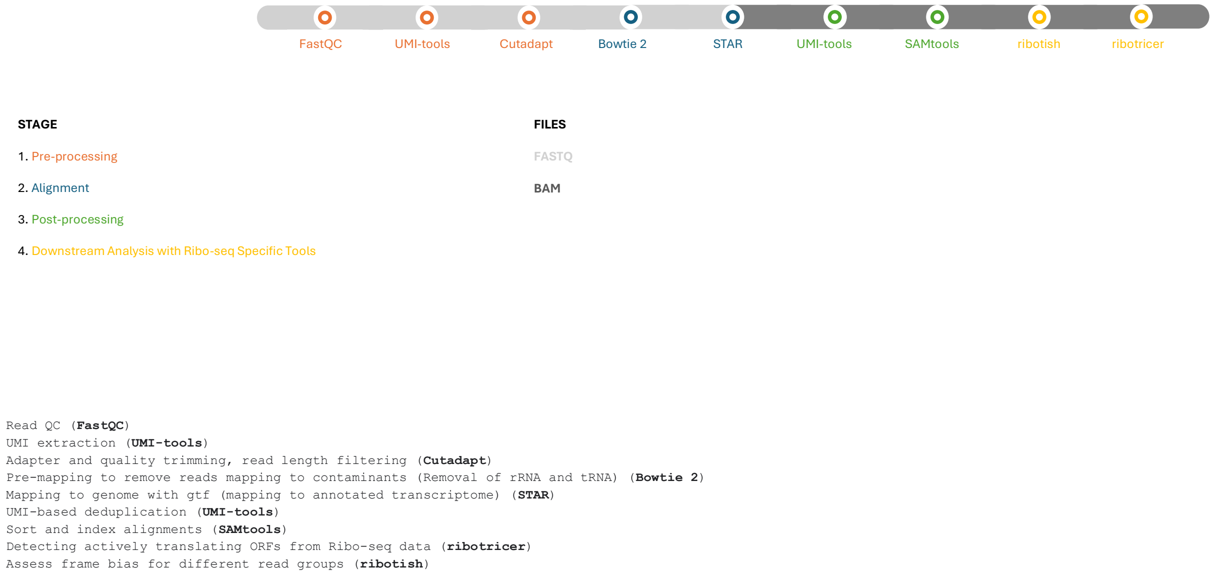
3.2.3 Seqkit

The Seqkit tool is an ultrafast toolkit for FASTA/Q file manipulation (Shen et al., 2016). In this step, Seqkit is used to reverse-complement antisense reads. The command used is:

```
seqkit seq —r —p —t DNA
<input_antisense_fastq> >
<output_reversed_antisense_fastq>
```

This command performs the following operations:

- **seq:** Calls the sequence manipulation subcommand.
- **-r:** Reverses each read in the file.
- **-p:** Complements the sequence after reversing, producing the reverse-complement.



```
Read QC (FastQC)
UMI extraction (UMI-tools)
Adapter and quality trimming, read length filtering (Cutadapt)
Pre-mapping to remove reads mapping to contaminants (Removal of rRNA and tRNA) (Bowtie 2)
Mapping to genome with gtf (mapping to annotated transcriptome) (STAR)
UMI-based deduplication (UMI-tools)
Sort and index alignments (SAMtools)
Detecting actively translating ORFs from Ribo-seq data (ribotricer)
Assess frame bias for different read groups (ribotish)
```

Fig. 2. Pipeline for Analyzing Ribosome Profiling Data.

- **-t DNA:** Specifies that the sequences are DNA, ensuring that reverse-complementing follows DNA base pairing rules (AT, CG).

This conversion is necessary to treat antisense reads equivalently to sense reads in the rest of the pipeline. **The resulting output contains only the reverse-complemented antisense reads.**

3.2.4 Cutadapt

In this step, Cutadapt is used again to process **the original FASTQ file**. The goal is twofold:

1. to remove antisense reads.
2. to trim the i5 index from the sense reads.

The command used is:

```
cutadapt -g <i5_index>
--discard-untrimmed -o
<output_sense_fastq>
<input_main_fastq>
```

- This operation, **–discard-untrimmed**, discards all reads that do not contain the specified i5 index. These are considered antisense reads and are excluded from this output.

The resulting output contains only the sense reads.

3.2.5 Cat

The UNIX cat command is used to concatenate files. In this step, it is applied to combine the two separate FASTQ files generated in the previous steps—one containing reverse-complemented antisense reads (from **Step 3.2.3**), and the other containing trimmed sense reads (from **Step 3.2.4**).

```
cat <input_antisense_reversed_fastq>
<input_sense_fastq> >
<output_combined_fastq>
```

3.2.6 Cutadapt

In this step, the Cutadapt tool is once again used to trim i5 index from our reads. Since the i5 index had already been trimmed from the sense reads in Step 3.2.4, the purpose of this step is to **remove the i5 index from the reverse-complemented antisense reads**, which still retained this sequence at their 5’ ends.

```
cutadapt -g <i5_index>
-o <output_trimmed_fastq>
<input_combined_fastq>
```

3.2.7 UMI-tools

UMI-tools is a specialized software suite designed for handling unique molecular identifiers (UMIs), which are short random sequences added to reads to differentiate between original RNA fragments and PCR duplicates (Smith et al., 2017). In this step, the extract function is used to locate and isolate UMIs embedded in each read, using a custom regular expression pattern to guide the extraction. The command used is:

```
umi_tools extract -I <input_fastq>
--extract-method=regex
--bc-pattern=”<regex_pattern >”
-S <output_fastq>
```

This command performs the following functions:

- **extract:** This subcommand identifies UMIs in each read and appends them to the read header for downstream deduplication.
- **–extract-method=regex:** Specifies that the UMI should be extracted using a regular expression (regex) pattern rather than a fixed format.
- **–bc-pattern=”<regex_pattern>”:** Defines the regular expression used to identify the barcode (UMI) location within the read.

UMI Extraction Strategy: In our dataset, each read contains a **12-nucleotide UMI sequence** immediately following the **3’ adapter sequence AACTGTAGGCACCATCAAT**. To specifically detect and extract these UMIs, the second half of this adapter — the sequence CACCATCAAT — was used as an anchor. The regex pattern was designed to:

- **Locate** the CACCATCAAT motif.
- **Extract** the 12 nucleotides that immediately follow it (i.e., the UMI).
- **Remove** both the anchor (CACCATCAAT) and the UMI from the read sequence.
- **Append** the extracted UMI to the read header, allowing for accurate PCR duplicate removal in later steps.

3.2.8 Cutadapt

In this step, Cutadapt is used to remove the 5’ adapter from the reads.

```
cutadapt -g <5_prime_adapter_sequence>
-o <output_trimmed.fastq>
<input_umi_extracted.fastq>
```

The adapter used in our library preparation protocol has a sequence of CTACACGACGCTCTTCCGATCT (Fig. 1).

3.2.9 Cutadapt

In this step, Cutadapt is used to remove the first half of the 3’ adapter from the reads.

```
cutadapt -a <3_prime_adapter_sequence>
-o <output_trimmed.fastq>
<input_trimmed.fastq>
```

During Step 3.2.7 (UMI-tools), the second half of the adapter sequence (CACCATCAAT) was used as an anchor for UMI extraction and was removed along with the 12-nucleotide UMI. However, the first half of the adapter, AACTGTAGG, still remains on reads and needs to be trimmed.

3.2.10 Cutadapt

In this step, Cutadapt is used to filter reads by length, retaining only those that fall within the desired size range for ribosome-protected fragments (RPFs).

```
cutadapt -m <min_read_length>
-M <max_read_length>
-o <output_length_filtered.fastq>
<input_trimmed.fastq>
```

For our Ribo-Seq data, valid reads were initially set as those between 17 and 34 nucleotides in length (The reasoning behind this choice is discussed in Section 5.2.).

3.2.11 Bowtie2

In this step, Bowtie2 is used to remove contaminating reads that align to ribosomal RNA (rRNA) and transfer RNA (tRNA) sequences (Langmead & Salzberg, 2012; Chothani et al., 2019). These non-coding RNA species are abundant in total RNA and must be removed to enrich for ribosome-protected fragments in Ribo-Seq analysis. A Bowtie2 index is built using a combined FASTA file of Homo sapiens rRNA and tRNA sequences, with rRNA downloaded from the Silva database and tRNA sequences from GtRNAdb. The command used for this step is:

```
bowtie2 -x <index_prefix>
-U <input_trimmed_fastq>
-S <output_mapped_sam>
--un <output_filtered_fastq>
```

Explanation of command components:

- **-x <index_prefix>:** Prefix of the Bowtie2 index built from the rRNA and tRNA reference FASTA.
- **-U <input_trimmed_fastq>:** Input FASTQ file containing reads that have been length-filtered (from Step 3.2.10).

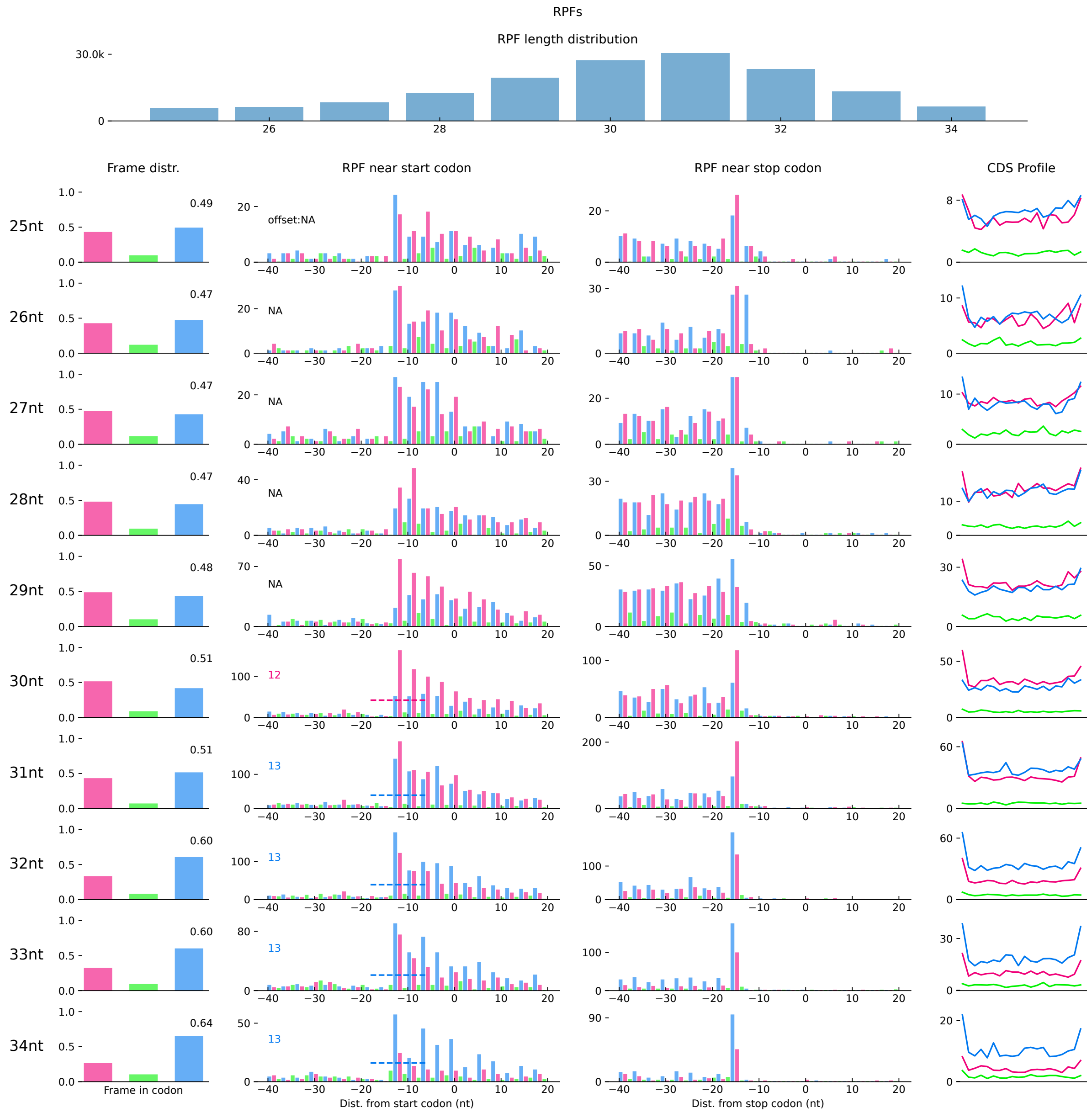


Fig. 3. Metagene Plots from Ribotish. Lower Panel includes a frame-wise distribution of read coverage, a positional summary of RPF counts surrounding translation initiation and termination sites, and an aggregate density profile of read coverage across coding regions for all genes.

- **-S <output_mapped_sam>:** Output file for reads that **do** map to rRNA/tRNA (not used in downstream analysis).
- **–un <output_filtered_fastq>:** Output FASTQ file containing **only reads that did not map** to the rRNA/tRNA database. This filtered FASTQ is used for downstream genome alignment.

3.2.12 Star

In this step, STAR is used to align the contaminant-filtered reads to the Homo sapiens GRCh38 reference genome and annotated transcriptome (Dobin et al., 2013; Chothani et al., 2019). the following reference files from Ensembl are used:

- Homo_sapiens.GRCh38.dna.primary_assembly.fa: Reference genome (primary assembly).
- Homo_sapiens.GRCh38.113.gtf: Gene annotation file (GTF) for GRCh38, version 113.

```
STAR --runThreadN <number_of_threads>
--genomeDir <genome_directory>
--readFilesIn <input_filtered_fastq>
--outFileNamePrefix <output_prefix>
--outSAMtype BAM SortedByCoordinate
--alignEndsType <alignment_end_type>
--outFilterMultimapNmax <max_multimaps>
--outSAMattributes All
```

Key parameters:

- **–runThreadN <number_of_threads>:** Number of threads to speed up processing on HPC.
- **–genomeDir <genome_directory>:** Directory containing the STAR genome index, built from the reference FASTA and GTF.
- **–alignEndsType EndToEnd:** Requires full-length read alignment with no soft clipping, which is appropriate for Ribo-Seq data.
- **-outFilterMultimapNmax 1:** Retains only uniquely mapping reads.

3.2.13 Samtools

Samtools is used to index the aligned BAM file, enabling efficient access to specific regions of the genome during further analyses.

```
samtools index <input_bam_file>
```

3.2.14 UMI-tools

UMI-tools is used to perform deduplication based on the unique molecular identifiers (UMIs), removing PCR duplicates and ensuring that only one representative read is retained for each original transcript fragment.

```
umi_tools dedup -I <input_bam_file>
-S <output_deduplicated_bam>
```

3.2.15 Samtools

Samtools is used to sort the deduplicated BAM file by genomic coordinates and then indexed the sorted BAM file for enabling efficient access to specific regions of the genome during further analyses (Danecek et al., 2021).

```
samtools sort <input_deduplicated_bam>
-o <output_sorted_bam>
samtools index <input_sorted_bam>
```

3.2.16 Subread (Optional)

FeatureCounts from the Subread package is used to count reads mapping to specific genomic features using the aligned and sorted BAM file, outputting the results as a CSV file (Chothani et al., 2019).

```
bam_file <- "<path_to_bam_file>";
gtf_file <- "<path_to_gtf_file>";
counts <-
featureCounts(files = bam_file,
annot.ext = gtf_file,
isGTFAnnotationFile = TRUE,
GTF.featureType = "<feature_type>",
GTF.attrType = "<attribute_type>",
useMetaFeatures = TRUE);
write.csv(counts$counts,
"<path_to_output_csv>")
```

3.2.17 Ribotricer

In this step, Ribotricer, a specialized tool for detecting actively translating open reading frames (ORFs) from ribosome profiling (Ribo-Seq) data, is used (Choudhary et al., 2020).

```
ribotricer detect-orfs
--bam <input_bam_file>
--ribotricer_index <ribotricer_index_file>
--prefix <output_prefix>
```

Ribotricer classifies detected ORFs into eight distinct types, each representing different genomic and transcriptomic features:

1. **Annotated ORF:** These are coding sequences (CDS) that are already identified and annotated in the provided GTF file.
2. **Super Upstream ORF (super_uORF):** This type refers to an upstream ORF located before the main CDS of a gene, with no overlap with other CDS of the same gene. It represents the first (or most upstream) uORF.
3. **Super Downstream ORF (super_dORF):** This ORF is found downstream of the annotated CDS of a gene, with no overlap with other CDS of the same gene, marking it as the last (or most downstream) dORF.
4. **Upstream ORF (uORF):** A translation initiation site located upstream of the main CDS, but not overlapping with it, representing a potential regulatory element.
5. **Downstream ORF (dORF):** A downstream ORF that does not overlap with the primary CDS, potentially indicating a separate translation event.
6. **Overlapping Upstream ORF (overlap_uORF):** This is an upstream ORF that overlaps with the main CDS of the same gene, suggesting a possible regulatory role in translation initiation.
7. **Overlapping Downstream ORF (overlap_dORF):** A downstream ORF that overlaps with the main CDS, which may indicate a complex translational mechanism or regulatory interaction.
8. **Novel ORF:** These ORFs are found in non-coding regions or in non-coding transcripts of genes that are typically classified as coding. They may represent unannotated or previously unidentified translation events.

The counts for each category will be reported in the results section (Fig. 6).

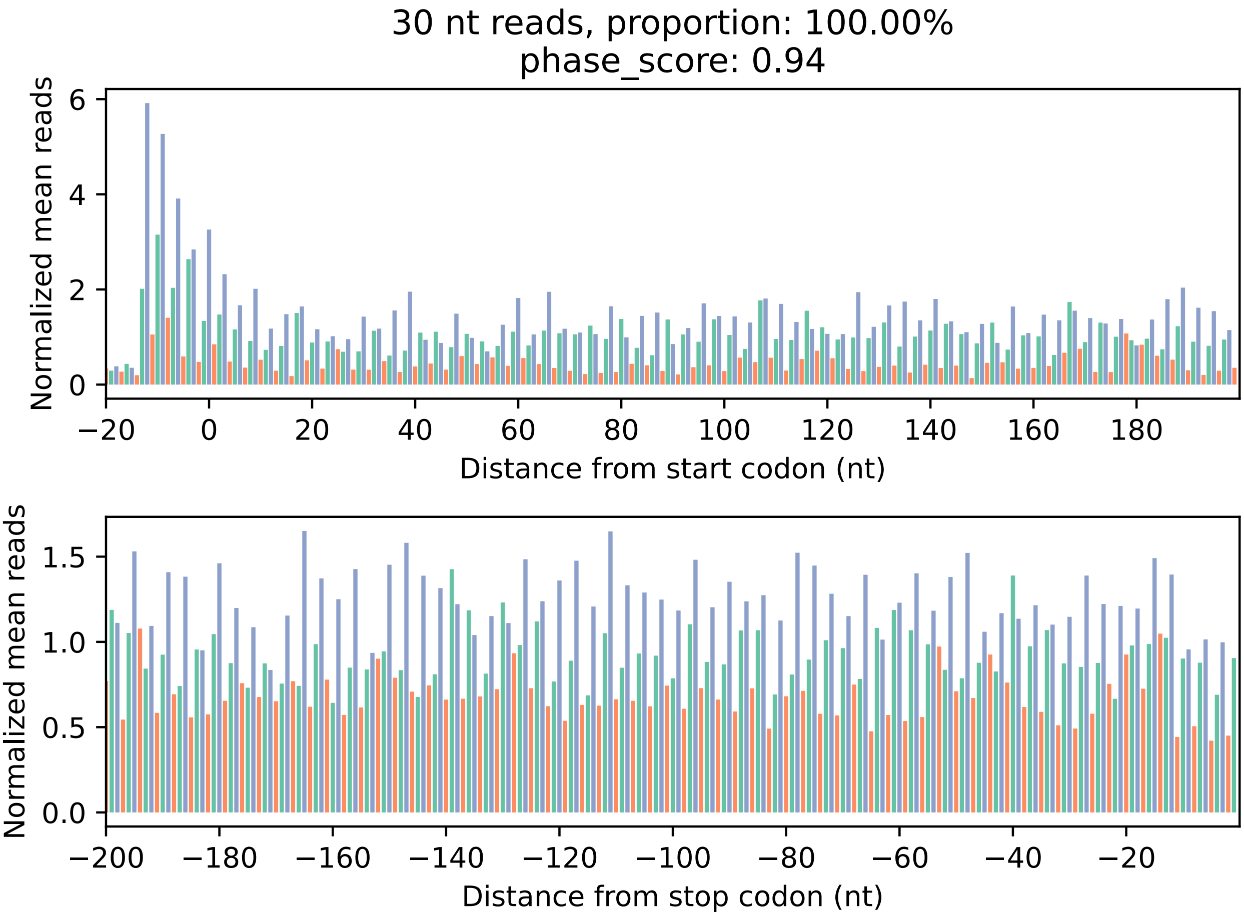


Fig. 4. Metagene Plots from Ribotricer. A positional summary of 30-nt RPF counts surrounding translation initiation and termination sites.

3.2.18 Ribotish

To evaluate the quality of ribosome profiling (Ribo-Seq) data, the quality module from RiboTISH is used (Zhang et al., 2017; Chothani et al., 2019). This module analyzes alignment data in BAM format to assess read distributions across annotated coding regions and estimate P-site offsets. For optimal performance, as recommended in the RiboTISH manual, the input BAM file consisted of uniquely mapped reads (as prepared in Step 3.2.12) that were trimmed to approximately 17–34 nucleotides, matching the expected length of ribosome-protected fragments (RPFs; see Step 3.2.10). Alignments were performed in end-to-end mode (Step 3.2.12) to avoid soft-clipping artifacts that could compromise frame precision. The command used was:

```
ribotish quality
—b <input_bam_file>
—g <gtf_file>
```

This generated a report with multiple visualizations summarizing key Ribo-Seq quality metrics:

1. **Upper Panel:** Displays the overall length distribution of RPFs uniquely mapped to annotated protein-coding regions.
2. **Lower Panel:** Provides frame-specific quality assessments for each read length:

   a. **Column 1:** Proportion of RPF 5' ends mapping to each of the three reading frames. The percentage of reads aligning to the dominant reading frame is used to evaluate frame periodicity.

   b. **Column 2:** Distribution of 5' ends around annotated translation initiation sites (TIS), used to estimate the optimal P-site offset and assess TIS alignment accuracy.

   c. **Column 3:** Distribution of 5' end counts near annotated stop codons, used to assess termination site precision.

   d. **Column 4:** RPF coverage across coding regions in each of the three frames.

The detailed frame bias and P-site offset estimations are presented in the results section, with accompanying figures for visual reference (Fig. 3).

Table 1. Steps and Counts.

Stage	Step	Total Sequences/Reads	Notes
Before Quality Control			
	Total Sequences	9,098,834	Input reads before any preprocessing.
Quality Control			
	After UMI Extraction	8,370,039	
	After Trimming Adapters	8,370,039	
	Length Filter (17-34nt)	4,193,612	
	Final Reads After QC	4,193,612	Final reads at the end of QC stage.
Alignment Stage			
	Input to Ribosomal RNA Removal	4,193,612	
	Reads aligned exactly 1 time	212,008 (5.06%)	
	Reads aligned more than 1 time	1,444,054 (34.43%)	
	Overall alignment rate	39.49%	During rRNA removal step.
	Aligned 0 time	2,537,550 (60.51%)	
	Input to Genome Contaminants Removal	2,537,550	
	Reads aligned exactly 1 time	93,063 (3.67%)	
	Reads aligned more than 1 time	165,487 (6.52%)	
	Overall alignment rate	10.19%	During genome contaminants removal step.
	Aligned 0 time	2,279,000	
	Reads After Contaminants Removal Steps	2,279,000	
	Unique Genome Alignments	670,531	Reads uniquely mapped to the transcriptome.
	Reads mapped to too many loci	1,253,246	
	Reads unmapped (too short)	355,220	
	Reads unmapped (other reasons)	3	
	Deduplicated Reads (UMI-based)	658,446	Reads retained after UMI deduplication.
	PCR Duplicates Removed	12,085	
	Final Reads After Alignment	658,446	End of alignment stage.

4. Results

4.1 Pre-processing

The initial FASTQ file comprised 9,098,834 raw sequencing reads. After antisense and sense reads separation using Cutadapt (Section 3.2.2) and Seqkit (Section 3.2.3), 4,853,177 antisense reads were reverse-complemented. A subsequent Cutadapt step (3.2.4) retained 4,245,657 sense reads, yielding a combined total of 9,098,834 reads upon concatenation (3.2.5), confirming successful antisense and sense reads separation and concatenation. UMI-tools (3.2.7) was then used to extract 12-nt unique molecular identifiers (UMIs) from reads, anchored to the adapter motif CACCATCAAT. After this step, 8,370,039 reads remained, indicating a loss of 728,795 reads—likely due to reads lacking the expected UMI motif positioning. Subsequent 5' and 3' adapter trimming with Cutadapt (Sections 3.2.8 and 3.2.9) retained the entire set of 8,370,039 reads, indicating no read loss during adapter removal. However, length filtering (3.2.10) retained only reads between 17–34 nt in length, reducing the dataset to 4,193,612 ribosome-protected fragments (RPFs) appropriate for downstream analysis (Table 1).

4.2 Alignment

To eliminate abundant contaminants, Bowtie2 was used to remove reads mapping to rRNA and tRNA (3.2.11). Out of 4,193,612 reads, 2,279,000 passed this filter, suggesting that 1,656,062 reads (39.49%) were rRNA-derived and 258,550 reads mapped to other genomic contaminants. The remaining 2,279,000 reads were aligned to the human GRCh38 genome and annotated transcriptome using STAR (3.2.12). A total of 670,531 reads uniquely aligned (Table 1).

4.3 Post-processing

UMI-based deduplication using UMI-tools (3.2.14) removed 12,085 PCR duplicates, resulting in a final count of 658,446 uniquely mapped RPF reads. Given that 30-nucleotide RPFs exhibited the strongest triplet periodicity in metagene analysis (see Discussion 5.3.1), the pipeline was rerun with a tighter length filter (exactly 30 nt) at Step 3.2.10. This resulted in 101,014 30-nt uniquely mapped, deduplicated reads.

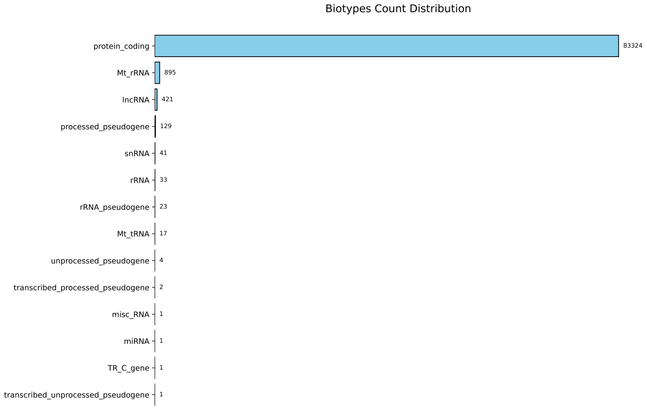


Fig. 5. Biotypes Count Distribution from Rsubread.

4.4 Downstream Analysis with Ribo-seq Specific Tools

To assess read counts at the gene level overlapping exons, featureCounts (3.2.16) assigned 84.0% (84,893) of the 101,014 filtered reads to annotated exonic features (Fig. 5). The distribution of mapped reads across biotypes confirmed expected results, with the vast majority corresponding to protein-coding genes (83,324). Minor proportions mapped to Mt_rRNA (895), lncRNA (421), and various pseudogenes and non-coding RNA types (Fig. 5). Using Ribotricer (3.2.17), 22,260 actively translating open reading frames (ORFs) from these 30-nt RPFs were detected (Fig. 6). The majority were annotated ORFs (7,063), followed by a substantial number of novel ORFs (8,750). Fewer upstream ORFs (uORFs; n = 22) were identified, as expected in our control sample without induced stress [The likelihood of translational repression increases with the number of uORFs present in a transcript (Wethmar, 2014).]. Cellular stress shifts translation toward non-canonical start sites, increasing the translation of uORFs and alternative protein isoforms (Wethmar, 2014). Previous work has shown that stress conditions such as starvation can drive a sixfold increase in ribosome occupancy within 5' UTRs, supporting enhanced uORF translation under these contexts (Ingolia et al., 2009). Although uORFs are often associated with translational repression, under certain conditions, they can enhance the translation of the main coding sequence (Wethmar, 2014). That is why uORFs can function as translational switches that enable fast cellular responses to environmental changes.

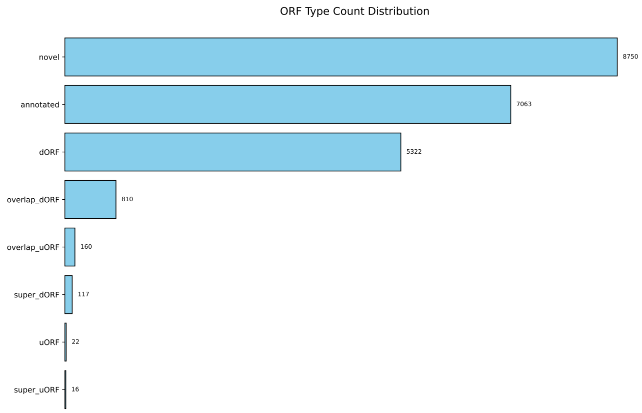


Fig. 6. ORF Type Count Distribution from Ribotricer.

5. Discussion

5.1 Read Coverage

5.1.1 RPF Read Coverage and SLC25A3 Gene Annotation Track

In Figure 7, the upper panel displays ribosome-protected fragment (RPF) read coverage across a randomly selected gene, while the lower panel shows the corresponding gene annotation. In this annotation track, narrow boxes at the 5' and 3' ends represent untranslated regions (UTRs), thick black boxes indicate coding sequences (CDSs), and connecting lines denote introns. Consistent with previous findings (98.8% in Ingolia et al. study), the majority of RPF reads (approximately 84% for 30-nt RPFs reads) in our dataset align predominantly with coding regions, and they are elevated near the 5' coding region, which is a common translational feature and not strongly influenced by transcript length or abundance (Ingolia et al., 2009).

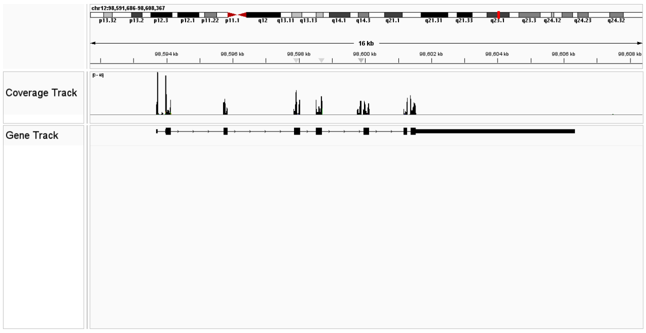


Fig. 7. RPF Read Coverage and SLC25A3 Gene Annotation Track.

5.2 Read Length Distribution

As outlined in Step 3.2.10, the length distribution of ribosome-protected mRNA fragments ranges from 17 to 34 nucleotides. For this analysis, the focus was specifically on 30-nucleotide reads, which exhibited the clearest triplet periodicity—corresponding to the one-codon-per-elongation-cycle pattern. Different versions of ribosome profiling use various translation inhibitors, and each one can freeze the ribosome at specific points in the elongation cycle (Lareau et al., 2014; Zhang et al., 2017). These points of arrest result in different fragment sizes, which reflect the underlying structural state of the ribosome. For instance, cycloheximide tends to trap ribosomes in a non-rotated state, where decoding or peptide bond formation is stalled. In this conformation, the ribosome protects approximately 28 to 30 nucleotides of mRNA (Lareau et al., 2014; Zhang et al., 2017). On the other hand, anisomycin and lincomycin favor the rotated state, with anisomycin stabilizing a unique configuration that leaves behind much shorter footprints, around 20 to 22 nucleotides in length (Lareau et al., 2014; Zhang et al., 2017). These shifts in fragment size across inhibitors highlight how the physical state of the ribosome directly affects the portion of mRNA it shields from RNase digestion. This connection becomes clearer when looking at how the ribosome moves during elongation. In each cycle—happening as fast as once every sixth of a second—the ribosome incorporates an amino acid, then advances exactly three nucleotides along the mRNA (Lareau et al., 2014). This process involves a chain of well-orchestrated structural changes. It starts with the delivery of an aminoacyl-tRNA to the A site by eEF1A, followed by codon recognition and peptide bond formation in the P site (Lareau et al., 2014). After that, the ribosome undergoes subunit rotation, entering the rotated state, during which tRNAs shift into hybrid positions—their anticodon loops stay in the A and P sites while their acceptor arms move to the P and E sites (Lareau et al., 2014). Further rearrangements prepare the ribosome for translocation, a step driven by eEF2 binding and GTP hydrolysis, which propels the ribosome forward and rotates the 40S head (Lareau et al., 2014). The process ends when the ribosome returns to the non-rotated state, ready to begin another cycle (Lareau et al., 2014). The ribosome's back-and-forth between these conformations is not only central to translation but also determines the fragment lengths recovered in ribosome profiling. Therefore, to capture a wider range of ribosome footprints, the fragment size selection was extended to include RNA pieces between 17 and 34 nucleotides following RNase I digestion.

5.3 Periodicity in RPFs

5.3.1 Metagene Plots from Ribotish and Ribotricer

Metagene analyses were conducted to assess the 5' end coverage of RPF reads around translation initiation and stop sites. This was restricted to transcripts with 5' UTRs no longer than 40 nucleotides and 3' UTRs no longer than 20 nucleotides (Fig. 3). In the Ribotish-generated plot, 5' ends of uniquely mappable RPF reads ranging from 25 to 34 nucleotides are shown. In contrast, the Ribotricer plot includes only uniquely mappable 30-nucleotide reads, focusing on regions spanning 20 nucleotides upstream of the initiation and stop codons (Fig. 4). The x-axis zero positions in both plots correspond to the start and stop codons. From the first column of the lower panel in the Ribotish figure, it is evident that RPFs of different lengths have distinct frame preferences. Notably, 30-nucleotide fragments display the strongest bias, with 51% dominance (consistent with Ingolia et al. study, 75% dominance was reported for their 28-nucleotide fragments) to the first reading frame (align and start with the first nucleotide of the start codon). Given its high informativeness, the 30-nucleotide RPF read length was emphasized in this study. Both Ribotish and Ribotricer plots reveal a clear peak of 5' ends located 12 nucleotides upstream of the start codon. This pattern indicates that, in 30-nucleotide reads, the P and A ribosomal sites are situated at positions 13–15 and 16–18, respectively, from the 5' end. Similar positional offsets have been documented in other studies (Ingolia et al., 2009; Aeschimann et al., 2015). Both Ribotish and Ribotricer plots of 30-nt RPF reads are positively skewed. This is consistent with prior findings that ribosome density often declines from initiation toward the end of translation, potentially due to acceleration of elongation or premature termination events (Ingolia et al., 2009). Previous work in plants revealed increased ribosome occupancy 16–17 nucleotides upstream of stop codons, suggesting a slowdown in ribosome progression during translation termination (Hou et al., 2016). In our data, this signal is also apparent, with 30-nt RPF reads showing a peak 15 nucleotides upstream of annotated stop codons.

5.4 The Problem of Contamination

Our ribosome profiling protocol did not include the treatment of the isolated RNA with any rRNA Removal Kit. As a result, the primary contaminants in our RPF library were ribosomal RNA (rRNA), which accounted for more than half of the total reads. Despite this, 39.49% of these contaminant reads were successfully removed. Not removing all the contaminants could lead to extensive loss of informative RPF reads. Interestingly, most rRNA-derived reads are longer than 32 nucleotides (Aeschimann et al., 2015). The Ribotish plot shows that 31- to 34-nucleotide reads are enriched in the third reading frame, possibly reflecting the influence of these contaminants or potential programmed ribosomal frameshifts that need to be studied further.

6. Conclusion

Our results show that the pipeline effectively filters and analyzes Ribo-Seq data, recovering high-quality 30-nt RPFs with strong coding region alignment. The detection of over 22,000 translated ORFs and clear triplet periodicity confirm good sample quality. This workflow offers a solid framework for evaluating Ribo-Seq data. The next step is to identify transcripts that are differentially translated between conditions, which requires matched RNA-seq data from the same samples. While DESeq2 is often used for this, specialized tools like RiboDiff, anota2seq (preferred choice), Babel, Riborex, RIVET, RiboDiPa, and Xtail are specifically designed for this task.

References

Aeschimann, F., Xiong, J., Arnold, A., Dieterich, C., & Großhans, H. (2015). Transcriptome-wide measurement of ribosomal occupancy by ribosome profiling. Methods, 85, 75–89. https://doi.org/10.1016/j.ymeth.2015.06.013 Chothani, S., Adami, E., Ouyang, J. F., Viswanathan, S., Hubner, N., Cook, S. A., Schafer, S., & Rackham, O. J. L. (2019). deltaTE: Detection of Translationally Regulated Genes by Integrative Analysis of Ribo-seq and RNA-seq Data. Current Protocols in Molecular Biology, 129(1), e108. https://doi.org/10.1002/cpmb.108 Choudhary, S., Li, W., & D. Smith, A. (2020). Accurate detection of short and long active ORFs using Ribo-seq data. Bioinformatics, 36(7), 2053–2059. https://doi.org/10.1093/bioinformatics/btz878 Danecek, P., Bonfield, J. K., Liddle, J., Marshall, J., Ohan, V., Pollard, M. O., Whitwham, A., Keane, T., McCarthy, S. A., Davies, R. M., & Li, H. (2021). Twelve years of SAMtools and BCFtools. GigaScience, 10(2), giab008. https://doi.org/10.1093/gigascience/giab008 Delahaye, C., & Nicolas, J. (2021). Sequencing DNA with nanopores: Troubles and biases. PLOS ONE, 16(10), e0257521. https://doi.org/10.1371/journal.pone.0257521 Dobin, A., Davis, C. A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M., & Gingeras, T. R. (2013). STAR: ultrafast universal RNA-seq aligner. Bioinformatics, 29(1), 15–21. https://doi.org/10.1093/bioinformatics/bts635 Hou, C., Lee, W., Chou, H., Chen, A., Chou, S., & Chen, H. (2016). Global Analysis of Truncated RNA Ends Reveals New Insights into Ribosome Stalling in Plants. The Plant Cell, 28(10), 2398–2416. https://doi.org/10.1105/tpc.16.00295 Ingolia, N. T., Brar, G. A., Rouskin, S., McGeachy, A. M., & Weissman, J. S. (2012). The ribosome profiling strategy for monitoring translation in vivo by deep sequencing of ribosome-protected mRNA fragments. Nature Protocols, 7(8), 1534–1550. https://doi.org/10.1038/nprot.2012.086 Ingolia, N. T., Ghaemmaghami, S., Newman, J. R. S., & Weissman, J. S. (2009). Genome-Wide Analysis in Vivo of Translation with Nucleotide Resolution Using Ribosome Profiling. Science, 324(5924), 218–223. https://doi.org/10.1126/science.1168978 Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. Nature Methods, 9(4), 357–359. https://doi.org/10.1038/nmeth.1923 Lareau, L. F., Hite, D. H., Hogan, G. J., & Brown, P. O. (2014). Distinct stages of the translation elongation cycle revealed by sequencing ribosome-protected mRNA fragments. ELife, 3, e01257. https://doi.org/10.7554/eLife.01257 Martin, M. (2011). Cutadapt removes adapter sequences from high-throughput sequencing reads. EMBnet.Journal, 17(1), 10. https://doi.org/10.14806/ej.17.1.200 McGlincy, N. J., & Ingolia, N. T. (2017). Transcriptome-wide measurement of translation by ribosome profiling. Methods, 126, 112–129. https://doi.org/10.1016/j.ymeth.2017.05.028 Ranasinghe, D., Jayadas, T. T. P., Jayathilaka, D., Jeewandara, C., Dissanayake, O., Guruge, D., Ariyaratne, D., Gunasinghe, D., Gomes, L., Wijesinghe, A., Wijayamuni, R., & Malavige, G. N. (2022). Comparison of different sequencing techniques for identification of SARS-CoV-2 variants of concern with multiplex real-time PCR. PLOS ONE, 17(4), e0265220. https://doi.org/10.1371/journal.pone.0265220 Shen, W., Le, S., Li, Y., & Hu, F. (2016). SeqKit: A Cross-Platform and Ultrafast Toolkit for FASTA/Q File Manipulation. PLOS ONE, 11(10), e0163962. https://doi.org/10.1371/journal.pone.0163962 Smith, T., Heger, A., & Sudbery, I. (2017). UMI-tools: modeling sequencing errors in Unique Molecular Identifiers to improve quantification accuracy. Genome Research, 27(3), 491–499. https://doi.org/10.1101/gr.209601.116 Wang, Q., & Mao, Y. (2023). Principles, challenges, and advances in ribosome profiling: from bulk to low-input and single-cell analysis. Advanced Biotechnology, 1(4), 6. https://doi.org/10.1007/s44307-023-00006-4 Wang, Y., Zhao, Y., Bollas, A., Wang, Y., & Au, K. F. (2021). Nanopore sequencing technology, bioinformatics and applications. Nature Biotechnology, 39(11), 1348–1365. https://doi.org/10.1038/s41587-021-01108-x Wethmar, K. (2014). The regulatory potential of upstream open reading frames in eukaryotic gene expression. WIREs RNA, 5(6), 765–768. https://doi.org/10.1002/wrna.1245 Wick, R. R., Judd, L. M., & Holt, K. E. (2019). Performance of neural network basecalling tools for Oxford Nanopore sequencing. Genome Biology, 20(1), 129. https://doi.org/10.1186/s13059-019-1727-y Zhang, P., He, D., Xu, Y., Hou, J., Pan, B.-F., Wang, Y., Liu, T., Davis, C. M., Ehli, E. A., Tan, L., Zhou, F., Hu, J., Yu, Y., Chen, X., Nguyen, T. M., Rosen, J. M., Hawke, D. H., Ji, Z., & Chen, Y. (2017). Genome-wide identification and differential analysis of translational initiation. Nature Communications, 8(1), 1749. https://doi.org/10.1038/s41467-017-01981-8