RESEARCH ARTICLE

Quality control and processing of nascent RNA profiling data

Jason P. Smith[1,4], Arun B. Dutta[4], Kizhakke Mattada Sathyan[4], Michael J. Guertin[1,4,✉], and Nathan C. Sheffield[1,2,3,4,✉]

[1]Center for Public Health Genomics, University of Virginia
[2]Department of Public Health Sciences, University of Virginia
[3]Department of Biomedical Engineering, University of Virginia
[4]Department of Biochemistry and Molecular Genetics, University of Virginia
✉ Correspondence: guertin@virginia.edu
✉ Correspondence: nsheffield@virginia.edu

Experiments that profile nascent RNA are growing in popularity; however, there is no standard analysis pipeline to uniformly process the data and assess quality. Here, we introduce PEPPRO, a comprehensive, scalable workflow for GRO-seq, PRO-seq, and ChRO-seq data. PEPPRO produces uniform processed output files for downstream analysis, including alignment files, signal tracks, and count matrices. Furthermore, PEPPRO simplifies downstream analysis by using a standard project definition format which can be read using metadata APIs in R and Python. For quality control, PEPPRO provides several novel statistics and plots, including assessments of adapter abundance, RNA integrity, library complexity, nascent RNA purity, and run-on efficiency. PEPPRO is restartable and fault-tolerant, records copious logs, and provides a web-based project report for navigating results. It can be run on local hardware or using any cluster resource manager, using either native software or our provided modular Linux container environment. PEPPRO is thus a robust and portable first step for genomic nascent RNA analysis.
Availability: BSD2-licensed code and documentation: `https://peppro.databio.org`.

Background

Steady-state transcription levels are commonly measured by RNA-seq, but there are many advantages to quantifying *nascent* RNA transcripts: First, it measures the transcription process directly, whereas steady-state mRNA levels reflect the balance of mRNA accumulation and turnover. Second, nascent RNA profiling measures not only RNA polymerase occupancy, but also orientation, and can be used to determine pausing and accumulation within any genomic feature. Third, nascent RNA profiling measures unstable transcripts, which can be used to infer regulatory element activity and identify promoters and enhancers *de novo* by detecting bidirectional transcription and clustered transcription start sites (TSSs)[1,2]. These advantages have led to widespread adoption of global run-on (GRO-seq), precision run-on (PRO-seq), and chromatin run-on (ChRO-seq) experiments[3–5]. With increasing data production, we require comprehensive analysis pipelines for these data types. While tools are available for downstream analysis, such as to identify novel transcriptional units and bidirectionally transcribed regulatory elements[1,6–10], there is no comprehensive, unified approach to initial sample processing and quality control.

Here, we introduce PEPPRO, an analysis pipeline for uniform initial sample processing and novel quality control metrics. PEPPRO features include: 1) a serial alignment approach to remove ribosomal DNA reads; 2) nascent transcription-specific quality control outputs; and 3) a modular setup that is easily customizable, allowing modification of individual command settings or even swapping software components by editing human-readable configuration files. PEPPRO is compatible with the Portable Encapsulated Projects (PEP) format, which defines a common project metadata description, making it easier to share projects across pipelines, computing environments, and analytical teams while facilitating interoperability with other PEP-compatible tools. PEPPRO can be easily deployed across multiple samples in any compute environment, including locally, with any cluster resource manager, in containers, or in a cloud. We have also produced a computing environment with all the command-line tools required to run PEPPRO using either docker or singularity with the bulker multi-container environment manager[11]. Thus, PEPPRO provides a unified, cross-platform pipeline for nascent RNA profiling projects.

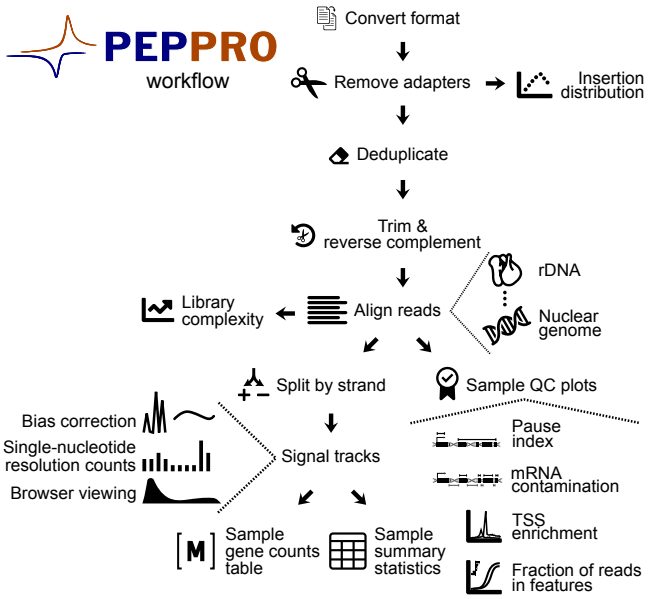

Fig. 1: ***PEPPRO provides a comprehensive set of steps to process genomic run on data.*** *PEPPRO performs basic data processing starting from raw sequencing reads and produces a variety of quality control plots and processed output files that facilitate more detailed downstream analysis.*

Results

Pipeline overview and data description

PEPPRO is a comprehensive pipeline that starts from raw, unaligned reads, and produces a variety of output formats, plots, and quality control metrics. Briefly, pre-alignment steps include removing adapters, deduplicating, trimming, and reverse complementation (Fig. 1). PEPPRO then uses a serial alignment strategy to siphon off unwanted reads from rDNA, mtDNA, and any other user-provided decoy sequences. It aligns reads and produces signal intensity tracks as both single-nucleotide counts files and smoothed normalized profiles for visualization. PEPPRO also provides a variety of plots and statistics to assess several aspects of library quality, such as complexity, adapter abundance, RNA integrity and purity, and run-on efficiency (See Methods for complete details).

To evaluate PEPPRO on different library types, we assembled a test set of run-on libraries with diverse characteristics (Fig. 2A). Our test set includes 7 previously published libraries: 2 ChRO-seq, 2 GRO-seq, and 3 PRO-seq[5,12–14]. We ran each of these samples through PEPPRO as a test case and visualized the data in a genome browser (Fig. 2B). To demonstrate PEPPRO's setup for differential expression analysis, we also generated paired-end PRO-seq libraries from H9 cell culture samples either naive or treated with romidepsin, a histone deacetylase inhibitor (HDACi). This test set therefore provides a range of qualities, protocols, and issues, providing a good test case for demonstrating the novel quality control features of PEPPRO and how to distinguish high-quality samples.

To demonstrate how PEPPRO responds to mRNA contamination, we also generated a set of 9 samples built from a single PRO-seq library (GSM1480327) that we spiked with increasing amounts of RNA-seq data (GSM765405) (Fig. S1). We ran PEPPRO on our public test set, our differential expression test set, and our spike-in set. Results of PEPPRO can be explored in the PEPPRO HTML-based web report, which displays all of the output statistics and QC plots (see PEPPRO documentation). Here, we describe each plot and statistic produced by PEPPRO.

Adapter ratio

A common source of unwanted reads in PRO/GRO/ChRO-seq libraries results from adapter-adapter ligation. These methods require two independent ligation steps to fuse distinct RNA adapters to each end of the nascent RNA molecule. The second ligation can lead to adapter-adapter ligation products that are amplified by PCR. The

A

#	Sample name	Description	Source
1	K562 PRO-seq	high quality PRO-seq data	GSM1480327
2	K562 GRO-seq	high quality GRO-seq data	GSM1480325
3	HelaS3 GRO-seq	low quality across multiple metrics	GSM1558746
4	Jurkat ChRO-seq 1	high quality ChRO-seq data	GSM3309956
5	Jurkat ChRO-seq 2	relatively low complexity	GSM3309957
6	HEK PRO-seq	intact (6) vs degraded (7) RNA	GSM3618147
7	HEK ARF PRO-seq		GSM3618143
8	H9 PRO-seq 1	Differential expression analysis; treatment causes Δ pause index	GSM4214080
9	H9 PRO-seq 2		GSM4214081
10	H9 PRO-seq 3		GSM4214082
11	H9 treated PRO-seq 1		GSM4214083
12	H9 treated PRO-seq 2		GSM4214084
13	H9 treated PRO-seq 3		GSM4214085

B

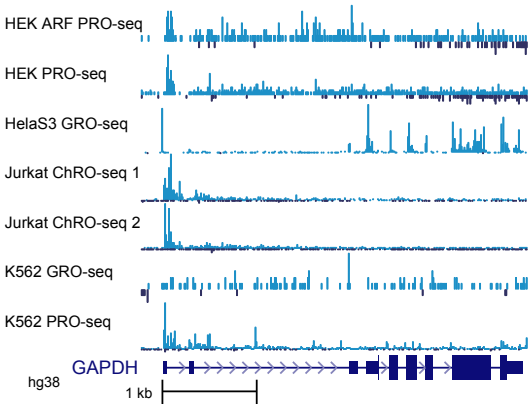

Fig. 2: ***PEPPRO test set data table and signal tracks.*** *A) Table showing the attributes of samples collected for our test set. Complete metadata is available from the PEPPRO website. B) Sample signal tracks from previously published test sets are visualized within a browser.*

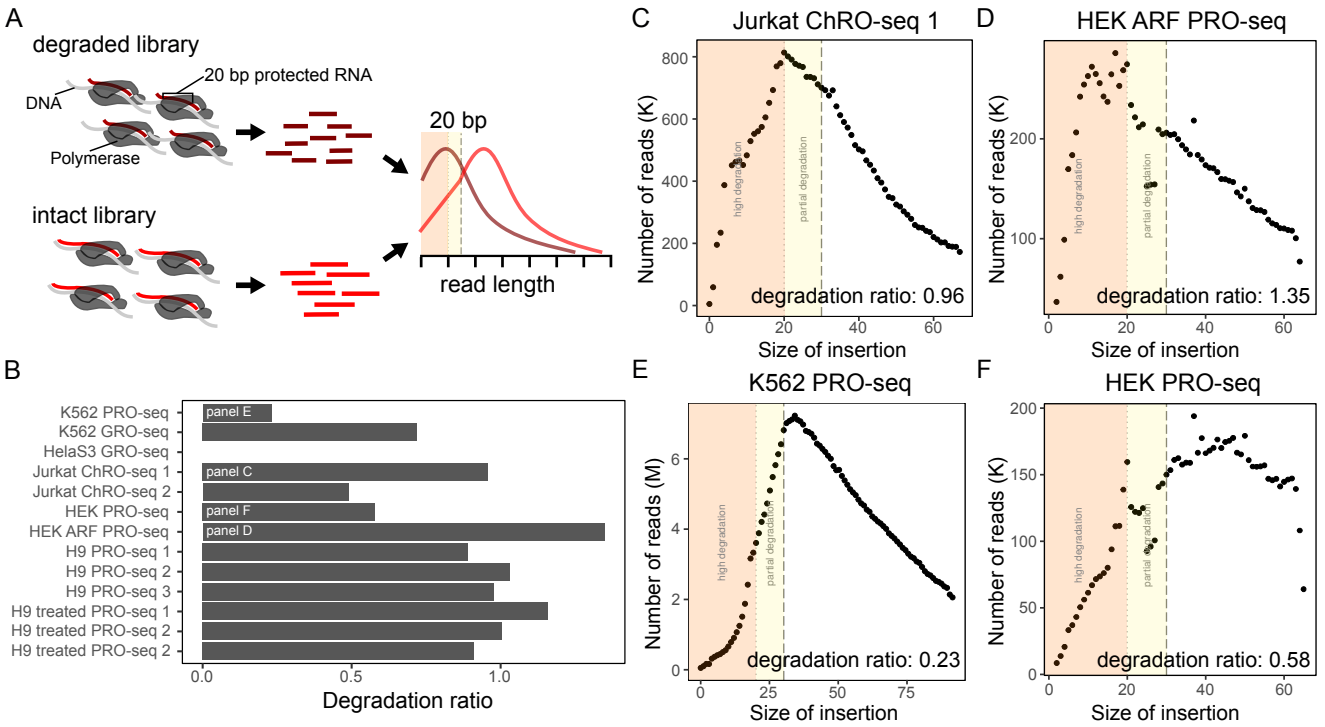

Fig. 3: ***RNA integrity is assessed with degradation ratios and insert sizes.*** *A) Schematic illustrating intact versus degraded libraries. B) Degradation ratio for test samples (HelaS3 GRO sample could not be calculated). C-F) Insert size distributions for: C, a degraded single-end library; D, a degraded paired-end library; E, a non-degraded single-end library; and F, a non-degraded paired-end library.*

frequency of adapter-adapter ligation can be reduced by molecular techniques (see Methods), but these are not always possible and many experiments retain adapters in high molar excess, leading to substantial adapter-adapter sequences. PEPPRO counts and reports the fraction of reads that contain adapter-adapter ligation products then removes adapter sequences and adapter-adapter ligation sequences before downstream alignment.

In our test, all samples had fewer than 50% adapter-adapter ligation reads (Fig. S2). Higher rates do not necessarily reflect lower quality samples, but rather indicate a suboptimal ratio of adapters during the library preparation or exclusion of the gel extraction size selection step. Excess adapters indicate that future sequencing will be less informative, leading to increased depth requirements, and therefore inform on whether to sequence a library deeper, tweak the adapter ratio in future samples, or include a size selection step. In our hands, we aim for adapter-adapter ligation abundance less than 50% with no size selection step, or around 5-20% if the final library is polyacrylamide gel electrophoresis (PAGE) purified. Libraries with no adapter-adapter ligation indicate that size selection was too stringent, and may actively select against short RNA insertions from specific classes of nascent RNA, such as RNAs from promoter-proximal paused polymerases[15].

RNA integrity

A common indicator of RNA sample quality is the level of RNA integrity. RNA integrity can be assessed by plotting the distribution of RNA insert sizes, which will be smaller when RNA is degraded. For a highly degraded library, we expect insert sizes below 20 nucleotides, which corresponds to the length of RNA between the RNA polymerase exit channel and 3′ RNA end. These nucleotides are sterically protected from degradation[16], so high frequency of insert sizes below 20 indicates that degradation occurred after the run-on step[5] (Fig. 3A).

PEPPRO uses a novel method to calculate the insert size distribution that applies to both single- and paired-end data (see Methods). PEPPRO reports the ratio of insert sizes from 10-20 nucleotides versus 30-40 nucleotides, which measures RNA integrity because more degraded libraries have higher frequency of reads of length 10-20, whereas less degraded libraries have more reads of length 30-40. In our test set, we found that high quality PRO-seq libraries have a ratio < 1 (Fig. 3B). A single-end ChRO-seq library that was intentionally degraded with RNase prior to the run on step[5] has a degradation ratio near 1 with a insertion distribution plot showing a peak at 20 nucleotides (Fig. 3C). A poor quality paired-end PRO-seq library contains many RNA species falling within the 10-20 range (Fig. 3D). High-quality libraries show plots that peak outside of the sub-20-nucleotide degradation zone (Fig. 3E, F).

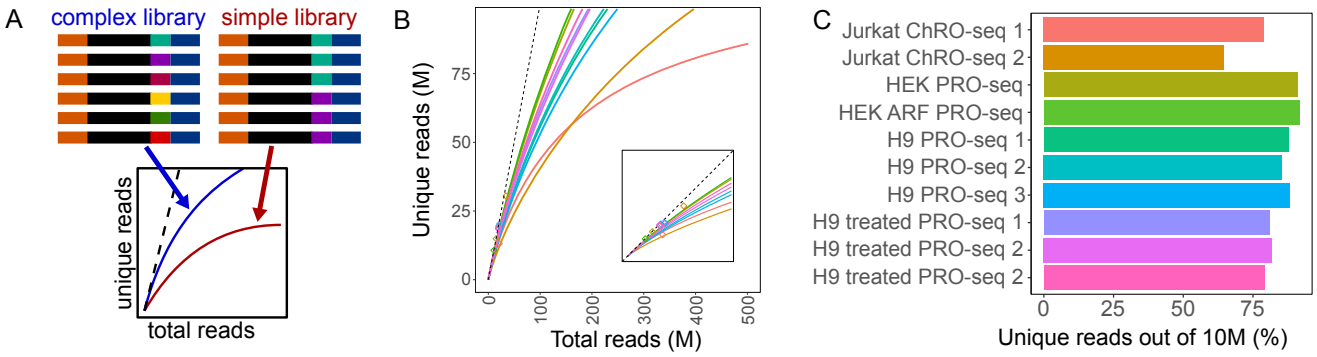

Fig. 4: ***Library complexity is measured with unique read frequency distributions and projections.*** *A) Schematic demonstrating PCR duplication and library complexity (dotted line represents completely unique library). B) Library complexity traces plot the read count versus externally calculated deduplicated read counts. Deduplication is a prerequisite, so these plots may only be produced for samples with UMIs. Inset zooms to region from 0 to double the maximum number of unique reads. C) The position of curves in panel B at a sequencing depth of 10 million reads.*

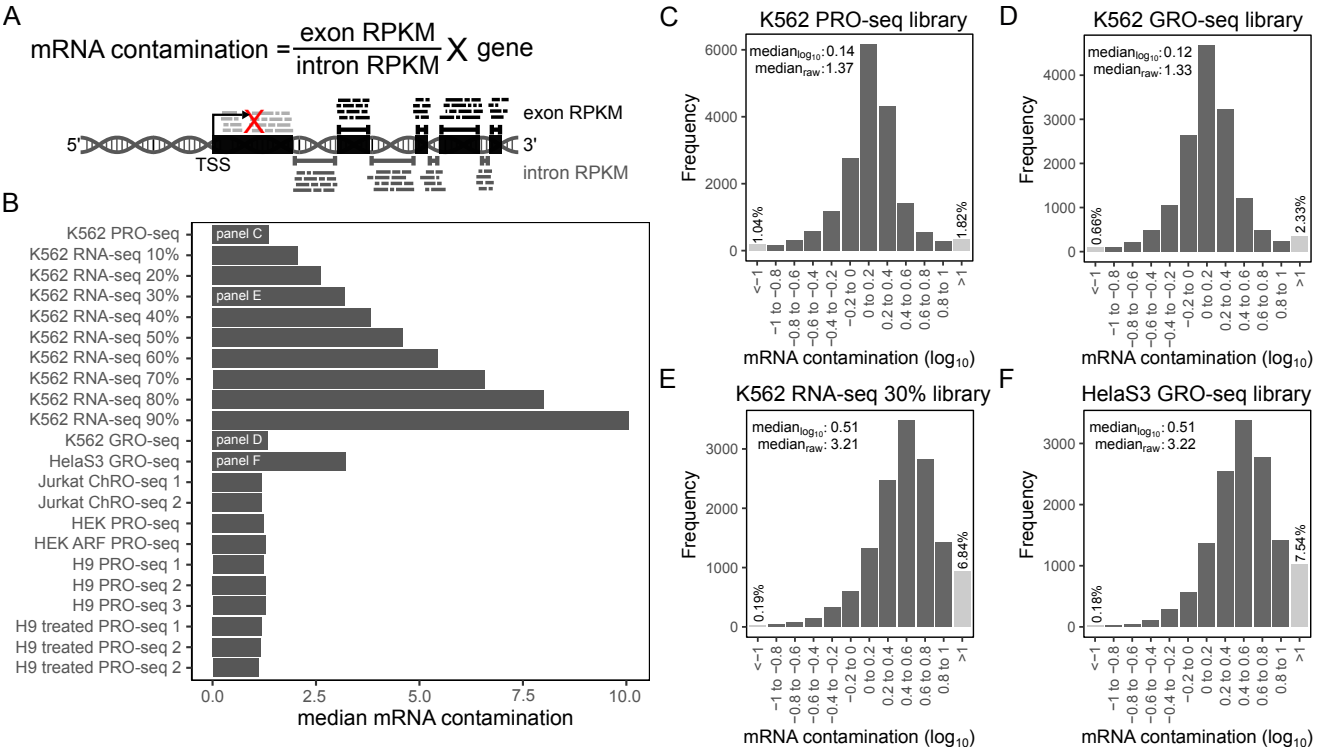

Fig. 5: ***Nascent RNA purity is assessed with the exon-intron ratio.*** *A) Schematic demonstrating mRNA contamination calculation. B) Median mRNA contamination metric for test set samples. C) Histogram showing the distribution of mRNA contamination score across genes in the K562 PRO-seq sample D) As in panel C for a GRO-seq library. E) mRNA contamination distribution for K562 PRO-seq spiked with 30% K562 RNA-seq. E) mRNA contamination distribution for HelaS3 GRO-seq is comparable to the 30% RNA-seq spike-in sample.*

Library complexity

Library complexity measures the uniqueness of molecules in a sequencing library (Fig. 4A). In PRO-seq, transcription start sites account for many of the 5′ RNA ends, and promoter proximal pause sites can focus the 3′ end of the RNA[4], so independent insertions with the same end points are not necessarily PCR duplicates. To solve this, PRO-seq protocols incorporate a unique molecular identifier (UMI) into the 3′ adapter, which PEPPRO uses to distinguish between PCR duplicates and independent RNA molecules with identical ends. PEPPRO accommodates multiple software packages for read deduplication, including `seqkit`[17] and `fqdedup`[18]. PEPPRO calculates library complexity at the current depth, reporting the percentage of PCR duplicates. In our test samples, we found that high quality libraries have at least 75% of reads unique at a sequencing depth of 10 million (Fig. 4C). PEPPRO also invokes `preseq`[19] to project the unique fraction of the library if sequenced at higher depth (Fig. 4B). These metrics provide a direct

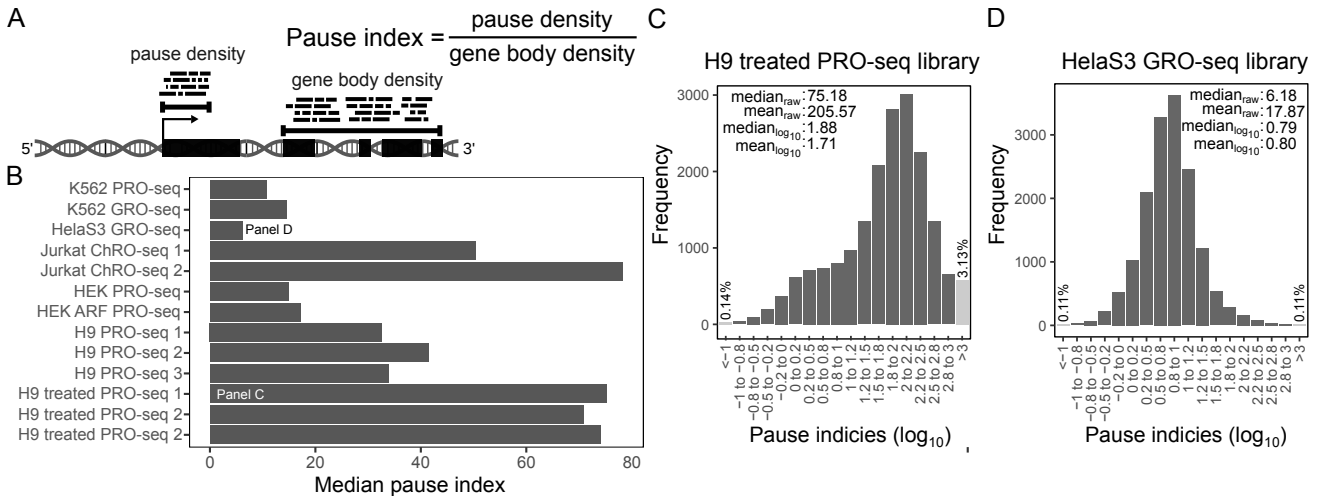

Fig. 6: ***Run-on efficiency is measured with pause indices****. A) Schematic demonstrating pause index calculation. B) Pause index values for test set samples. C) High pause index from H9 treated PRO-seq. D) Low pause index from HelaS3 GRO-seq.*

measure of library complexity and allow the user to determine value of additional sequencing.

Nascent RNA purity

One challenge specific to nascent RNA sequencing is ensuring that the library targets nascent RNA specifically, which requires eliminating the more abundant processed rRNA, tRNA, and mRNA transcripts. Early run-on protocols included 3 successive affinity purifications, resulting in 10,000-fold enrichment over mRNA and over 98% purity of nascent RNA[3,4]. Newer run-on protocols recommend fewer affinity purifications[14]. Therefore, assessing the efficiency of nascent enrichment is a useful quality control output.

PEPPRO provides rDNA alignment rate and an mRNA contamination metric to estimate the *nascent purity* of RNA. First, since rRNA represents the vast majority of stable RNA species in a cell, overrepresentation of rDNA reads indicates poor nascent RNA enrichment. The fraction of nascent rRNA transcription is likely to be distinct among cell lines, but most of our high-quality PRO-seq samples found rDNA-alignment rates between 10% and 20% (Fig. S3). A second measure of nascent purity is to evaluate messenger RNA abundance. PEPPRO assesses this by calculating the exon to intron read density ratio (Fig. 5A). A nascent RNA sequencing library without polymerase pausing would have a ratio of exon density to intron density of ˜1. Because promoter-proximal pausing inflates this ratio, PEPPRO excludes the first exon from this calculation. In our test samples, the median exon-intron ratio is between 1.1 and 1.4 for high quality libraries (Fig. 5B). Our *in silico* spike-in of conventional RNA-seq increases this ratio proportionally to the level of mRNA contamination (Fig. 5B). This ratio varies substantially among genes and PEPPRO produces histograms to compare in more detail among samples (Fig. 5C-F). By comparing these values to the spike-in experiment, we can estimate the level of mRNA contamination of a library (Fig. 5E, F)

Run-on efficiency

Another quality metric for run-on experiments is run-on efficiency. Typically, gene-body polymerases extend efficiently during the nuclear run-on step, but promoter-proximal paused polymerases require either high salt or detergent to do so[20,21]. Because these treatments vary, PEPPRO employs two methods to assess run-on efficiency: *pause index* and *TSS enrichment*. First, PEPPRO calculates the pause index, which is defined as the ratio of the density of reads in the *pausing* region versus the density in the corresponding gene body (Fig. 6A; see Methods). PEPPRO plots the frequency distribution of the pause index across genes. A greater pause index indicates a more efficient run-on, as a higher value indicates that paused polymerases efficiently incorporate the modified NTPs. We found in our test samples that an efficient run-on process has a median pause index greater than 10 (Fig. 6B). For more detail, PEPPRO produces frequency distribution plots that show an exponential distribution among genes for an efficient library (or a normal distribution on a log scale, Fig. 6C) and a shifted distribution for an inefficient run-on (Fig. 6D).

As a second assessment of run-on efficiency, PEPPRO aggregates sequencing reads at TSSs to plot and calculate a TSS enrichment score. PEPPRO plots aggregated reads 2000 bases upstream and downstream of a reference set of TSSs. Efficient TSS plots show a characteristic PRO-seq pattern with an upstream peak for divergently transcribing polymerases and a prominent peak representing canonical paused polymerases (Fig. S4). PEPPRO also summarizes these values across samples.

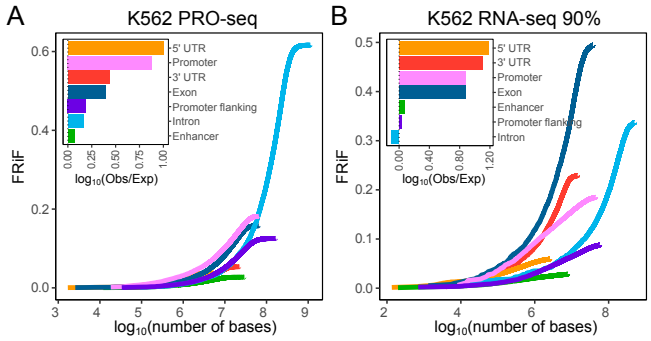

Fig. 7: ***Fraction of reads in genomic features****. A) K562 PRO-seq represents a "good" cumulative fraction of reads in features (cFRiF) and fraction of reads in features (FRiF) plot. B) K562 PRO-seq with 90% K562 RNA-seq spike-in represents a "bad" FRiF/PRiF.*

Read feature distributions

PEPPRO also produces plots to visualize the *fraction of reads in features*, or FRiF. The cumulative FRiF (cFRiF) plot provides an information-dense look into the genomic distribution of reads relative to genomic features. This analysis is a generalization of the more common *fraction of reads in peaks* (FRiP) plots produced for other data types[22] with two key differences: First, it shows how the reads are distributed among different features, not just peaks; and second, it uses a cumulative distribution to visualize how quickly the final read count is accumulated in features of a given type. To calculate the FRiF, PEPPRO overlaps each read with a feature set of genomic annotations, including: enhancers, promoters, promoter flanking regions, 5′ UTR, 3′ UTR, exons, and introns (Fig. 7). The individual feature elements are then sorted by read count, and for each feature, we traverse the sorted list and calculate the cumulative sum of reads found in that feature divided by the total number of aligned reads. We plot the read fraction against the log_{10} transformed cumulative size of all loci for each feature. This allows the identification of features that are enriched for reads with fewer total features and total genomic space. Additionally, PEPPRO calculates the non-cumulative FRiF by taking the log_{10} of the number of observed bases covered in each feature over the number of expected bases in each feature to identify enriched genomic features (Fig. 7).

In our test samples, high-quality libraries have a characteristic pattern with slow accumulation but high total of reads in introns, and fast accumulation but lower total of reads in promoter elements. ChRO-seq libraries have an increased promoter emphasis and higher mRNA contamination indicated by an increase in reads in promoters and exons at the cost of reads in introns and promoter flanking regions (Fig. S5). Additionally, the RNA-seq spike-in samples demonstrate the increasing prevalence of exonic reads and 3′ UTR at the cost of intronic sequences (Fig. S6). These plots are therefore a useful general-purpose quality control tool that reveal substantial information about a sample in a concise visualization.

Differential expression

The focus of PEPPRO is in the pre-processing relevant for any type of biological project. The output of PEPPRO sets the stage for downstream analysis specific to a particular biological question. Probably the most common type of specific downstream analysis is a differential expression analysis, so PEPPRO produces all the necessary results to immediately enter a differential analysis with dedicated tools. To demonstrate this, we included 3 samples treated with romidepsin and 3 untreated control samples. PEPPRO sets the user up to easily run a differential comparison using dedicated software, like the DESeq bioconductor package[23].

To facilitate differential expression analysis, PEPPRO produces a project-level counts table that may be loaded in R using `pepr`, and, in a few lines of code, converted quickly into DEseq data sets ready for downstream

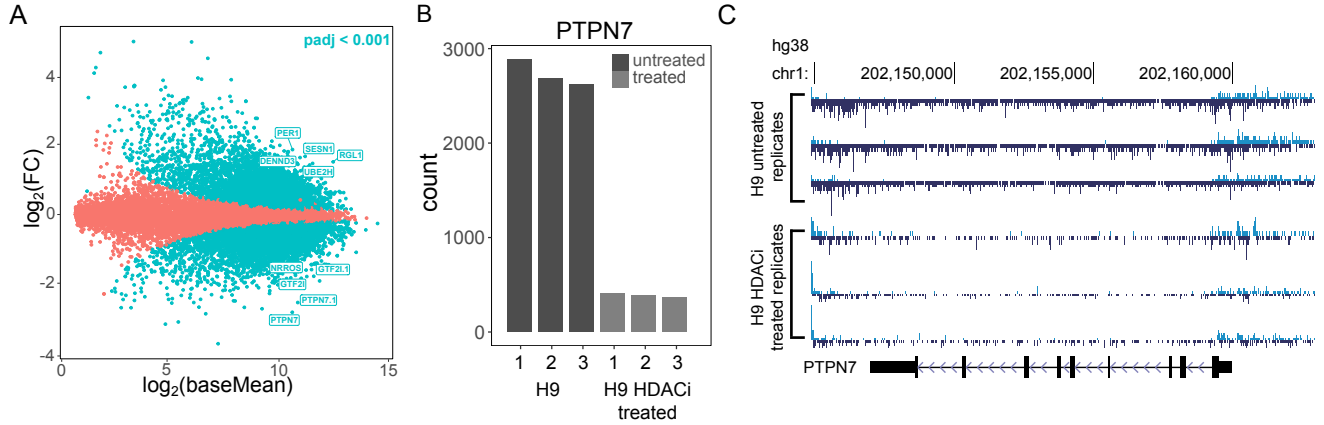

Fig. 8: ***Differential analysis with the PEPPRO counts matrix****. A) MA plot between H9 DMSO versus H9 200nM romidepsin treated PRO-seq libaries (top 10 most significant genes labeled; n=3/treatment). B) Most significantly differential gene count differences. C) Signal tracks from the differential analysis.*

Metric	Recommended value
Degradation ratio	< 1
rDNA alignment rate	< 20%
Pause index	> 10
mRNA contamination	1 - 1.5
% uniformative adapter reads	< 25%
TSS enrichment (coding)	> 10
TSS enrichment (non-coding)	> 5
% unique at 10M reads	> 75%

Fig. 9: ***Recommendation table.*** *Based on our experience processing both high- and low-quality nascent RNA libraries, these are the recommended values for high-quality PRO-seq libraries.*

DESeq analyses (See Supplemental text). Using this approach, we ran a differential expression analysis comparing romidepsin-treated against untreated samples (Fig. 8A). We identified many genes with significantly different read coverage. As an example, the PTPN7 gene showed clear differences in counts (Fig. 8B), which we can further visualize using the browser track outputs generated by PEPPRO (Fig. 8C). This analysis demonstrates how simple it is to ask a downstream biological question starting from the output produced by PEPPRO.

Conclusions

PEPPRO is an efficient, user-friendly PRO/GRO/ChRO-seq pipeline that produces novel, integral quality control plots and signal tracks that provide a comprehensive starting point for further downstream analysis. The included quality control metrics inform on library complexity, RNA integrity, nascent RNA purity, and run-on efficiency with theoretical and empirical recommended values (Figure 9). PEPPRO is uniquely flexible, allowing pipeline users to serially align to multiple genomes, to select from multiple bioinformatic tools, and providing a convenient configurable interface so a user can adjust parameters for individual pipeline tasks. Furthermore, PEPPRO reads projects in PEP format, a standardized, well-described project definition format, providing an interface with Python and R APIs to simplify downstream analysis.

PEPPRO is easily deployable on any compute infrastructure, including a laptop, a compute cluster, or the cloud. It is thereby inherently expandable from single to multi-sample analyses with both group level and individual sample level quality control reporting. By design, PEPPRO enables simple restarts at any step in the process should the pipeline be interrupted. At multiple steps within the pipeline, different software options exist creating a swappable pipeline flow path with individual steps adaptable to future changes in the field. PEPPRO is a rapid, flexible, and portable PRO/GRO/ChRO-seq project analysis pipeline providing a standardized foundation for more advanced inquiries.

Availability

Documentation on the Portable Encapsulated Project (PEP) standard may be found at pepkit.github.io.

`Refgenie` documentation and pre-built reference genomes are available at refgenie.databio.org.

The HTML report for the test samples is located at `http://big.databio.org/papers/peppro/PEPPRO_summary.html`.

Methods

Pipeline implementation

The PEPPRO pipeline is a python script (`peppro.py`) runnable from the command-line. PEPPRO is built using the python module `pypiper`[24], which provides restartability, file integrity protection, copious logging, resource monitoring, and other features. Individual pipeline settings can be configured using a pipeline configuration file (`peppro.yaml`), which enables a user to specify absolute or relative paths to installed software and parameterize adapter clipping, deduplication, read trimming, and signal track generating software tools. This configuration file comes with sensible defaults and will work out-of-the-box for research environments that include required software in the shell PATH, but may be configured to fit any computing environment and is adaptable to project-specific parameterization needs.

Refgenie reference assembly resources

Several PEPPRO steps require generic reference genome assembly files, such as sequence indexes and annotation files. For example, alignment with `bowtie2` requires `bowtie2` indexes, and feature annotation to calculate fraction of reads in features requires a feature annotation. To simplify and standardize these assembly resources, PEPPRO uses *refgenie*. Refgenie is a reference genome assembly asset manager that streamlines downloading, building, and using data files related to reference genomes[25]. Refgenie includes recipes for building genome indexes and genome assets as well as downloads of pre-indexed genomes and assets for common assemblies. Refgenie enables easy generation of new standard reference genomes as needed. For a complete analysis, PEPPRO requires a number of refgenie managed assets. Those assets as defined by refgenie are: fasta, bowtie2_index, ensembl_gtf, ensembl_rb, refgene_anno, and feat_annotation. If building these assets manually, they separately require a genome fasta file, a gene set annotation file from RefGene, an Ensembl gene set annotation file in GTF format, and an Ensembl regulatory build annotation file. Finally, using PEPPRO with `seqOutBias` requires the additional refgenie tallymer_index asset of the same read length as the data.

Adapter-adapter ligation product abundance

Adapter-adapter ligation products show up in run-on libraries because there are two independent ligation steps. Sequencing these products is uninformative, and so there are several molecular approaches used to reduce their abundance in a sequencing library. All protocols include an inverted dT on the 3′ end of the 3′ adapter, and also do not phosphorylate the 5′ end of the 5′ adapter. Many protocols include a size-selection gel extraction step to purify the library from a prominent adapter-adapter ligation species.

PEPPRO calculates adapter-adapter ligation products directly from `cutadapt` output, and the default `-m` value for this step is the length of the UMI plus two nucleotides. Therefore, if RNA insertions fewer than three nucleotides in length are present in the library, these are treated as adapter-adapter ligation products.

RNA insert size distribution and degradation

For both single and paired end data, the RNA insert size distribution is calculated prior to alignment. For single end data, the calculation is derived only from sequences that contain adapter sequence, which is output directly from `cutadapt`[26]. PEPPRO plots the inverse `cutadapt` report fragment lengths against the `cutadapt` fragment counts. If there is a known UMI, based on user input, that length is subtracted from reported `cutadapt` fragment lengths. As a consequence of this distribution, we can establish a measure of library integrity by evaluating the sum of fragments between 10-20 bases versus the sum of fragments between 30-40 bases in length. The higher this degradation ratio, the more degraded the library.

Paired end sequencing files often have shorter reads because a standard 75 base sequencing cartridge can be used for two paired end reads that are each 38 nucleotides in length. Therefore, many fewer of the reads derived from either end of the molecule extend into the adapter sequence. To address this issue, we incorporate a step that fuses overlapping reads using `flash`[27]. Therefore, if two paired end reads contain overlapping sequence, the reads are combined and the insert size is calculated directly from the fused reads and output directly from `flash`. This distribution is plotted identically to the single end reads and degradation is calculated in the same manner. This degradation ratio metric is uniform between single-end or paired-end libraries and is reported prior to any alignment steps, minimizing influences from extensive file processing or alignment eccentricities.

Excluding size selection skews metrics

Recent PRO-seq protocols, including the H9 libraries we generated, exclude the PAGE size selection step that removes adapter-adapter ligation products[14]. Size selection can potentially bias against small RNA insertions. The previous two metrics: adapter-adapter abundance and degradation ratio are naturally skewed toward the undesirable range if libraries are constructed without size selection. Adapter abundance is skewed because the sole purpose of size selection is to remove the adapter species, but these uninformative reads are of minimal concern and can be overcome by increasing sequencing depth. Degradation ratio is skewed higher because the size selection is not perfect and insert sizes in the range of 10-20 are preferentially selected against relative to those in the 30-40 range. Therefore, while we provide recommendations for optimal degradation ratios, this metric is not necessarily comparable between library preparation protocols and a higher ratio is expected for protocols that exclude size selection.

Removing UMI and reverse complementation

In a typical sequencing library, low library complexity is indicated by high levels of PCR duplicates. Conventional methods remove independent paired-end reads that map to the same genomic positions. This method works reasonably well for molecular genomics data sets with random nucleic acid cleavage. However, in PRO-seq, transcription start sites account for many of the 5′ RNA ends and polymerases pause downstream in a focused region[4]. Consequently, independent insertions with the same end points are common, especially in the promoter-proximal region. To solve this, PRO-seq protocols incorporate a unique molecular identifier (UMI) into the 3′ adapter to distinguish between PCR duplicates and independent insertions with shared ends. PEPPRO removes PCR duplicates only if UMIs are provided.

Following the removal of PCR duplicates, the UMI is trimmed. For run-on experiments where the sequencing primer sequences the 3′ end of the original RNA molecule, reverse complementation is performed. As only the first read contains a UMI in paired-end experiments, the second reads skip UMI trimming. Both steps are performed using either `seqtk` (https://github.com/lh3/seqtk) or `fastx` (https://github.com/agordon/fastx_toolkit), depending on user preference. Because reads are processed uniquely for first and second reads in a paired-end experiment, reads must be re-paired prior to alignment. PEPPRO uses the optimized implementation `fastq-pair`[28] to re-pair desynchronized read files.

Serial alignments

Following re-pairing, or starting from processed single-end reads, PEPPRO performs a series of preliminary,

serial alignments (prealignments) before aligning to the primary reference using `bowtie2`[29]. As a significant portion of nascent transcription includes rDNA, PEPPRO defaults to initially aligning all reads to the human rDNA sequence. Not only does this remove rDNA reads from downstream analysis, it improves computational efficiency by aligning the largest read pool to a small genome and reduces that read pool for subsequent steps. The user can specify any number of additional genomes to align to prior to primary alignment, which may be used for species contamination, dual-species experiments, repeat model alignments, decoy contamination, or spike-in controls. Following these serial alignments, any remaining reads are aligned to the primary genome. Alignment statistics (number of aligned reads and alignment rate) for all serial alignments and primary alignments are reported. For the primary alignment, PEPPRO also reports the number of mapped reads, the number removed for quality control (low-quality, multi-mapping, or unmapped paired reads), the total efficiency of alignment (aligned reads out of total raw reads), and the read depth. Prior to further downstream analysis, paired-end reads are split into separate read alignment files and only the first read is retained for downstream processing. For both paired-end and single-end experiments, this aligned read file is split by strand with both plus and minus strand aligned files further processed.

Processed signal tracks

Following read processing, alignment, strand separation, and quality control reporting, aligned reads are efficiently converted into strand-specific bigWig files. For PRO-seq and similar protocols, the 3′ ends of reads are reported. Optionally, PEPPRO can use `seqOutBias`[30] to correct enzymatic sequence bias by taking the ratio of genome-wide observed read counts to the expected sequence based counts for each k-mer. K-mer counts take into account mappability at a given read length using Genome Tools' `Tallymer` program[31]. Strand specific bigWigs may be visually analyzed using genomic visualization tools and provide a unified starting point for downstream analyses. For example, output bigWig files can be directly loaded into dREG to identify regulatory elements defined by bidirectional transcription[1].

Exon-intron ratio plots

PEPPRO provides both an mRNA contamination histogram for quick visual quality control, and a BED format file containing gene by gene exon:intron ratios for detailed analysis.

Pause index

Pause indices refer to the ratio of read density in the promoter proximal region and the gene body. Pause indices can vary widely depending on the defined pause window and how a pause window is determined (i.e. relative to a TSS or the most dense window proximal to a TSS). PEPPRO defines the density within the pause region as the single, most dense window +20-120 bp taken from all annotated TSS isoforms per gene. The gene body is defined as the region beginning 500 bp downstream from the TSS to the gene end. Finally, PEPPRO plots the distribution of pause indices for each gene in a histogram and provides a BED-formatted file containing each gene's pause index for more detailed analyses.

PRO-seq experiments

H9 PRO-seq experiments were conducted as described previously[14]. The HDACi-treated samples were incubated with 200nM romidepsin for 60 minutes prior to harvesting. The control "untreated" samples were treated with DMSO for 60 minutes.

Funding

This work was supported by the National Institute of General Medical Sciences grants GM128635 (MJG) and GM128636 (NCS). JPS was supported by the institutional training grant GM008136. ABD was supported by institutional training grants LM012416 and GM007267.

References

1. Wang, Z., Chu, T., Choate, L. A. & Danko, C. G. Identification of regulatory elements from nascent transcription using dREG. *Genome research* **29,** 293–303 (2019).

2. Scruggs, B. S. *et al.* Bidirectional transcription arises from two distinct hubs of transcription factor binding and active chromatin. **58,** 1101–1112 (2015).

3. Core, L. J., Waterfall, J. J. & Lis, J. T. Nascent rna sequencing reveals widespread pausing and divergent initiation at human promoters. *Science* **322,** 1845–1848 (2008).

4. Kwak, H., Fuda, N. J., Core, L. J. & Lis, J. T. Precise maps of rna polymerase reveal how promoters direct initiation and pausing. **339,** 950–953 (2013).

5. Chu, T. *et al.* Chromatin run-on and sequencing maps the transcriptional regulatory landscape of glioblastoma multiforme. *Nature genetics* **50,** 1553–1564 (2018).

6. Chae, M., Danko, C. G. & Kraus, W. L. GroHMM: A computational tool for identifying unannotated and cell type-specific transcription units from global run-on sequencing data. *BMC bioinformatics* **16,** 222 (2015).

7. Azofeifa, J. G., Allen, M. A., Lladser, M. E. & Dowell, R. D. An annotation agnostic algorithm for detecting nascent rna transcripts in gro-seq. *IEEE/ACM transactions on computational biology and bioinformatics* **14,** 1070–1081 (2017).

8. Allison, K. A., Kaikkonen, M. U., Gaasterland, T. & Glass, C. K. Vespucci: A system for building annotated databases of nascent transcripts. *Nucleic acids research* **42,** 2433–2447 (2014).

9. Wang, J. *et al.* Nascent rna sequencing analysis provides insights into enhancer-mediated gene regulation. *BMC genomics* **19,** 633 (2018).

10. Anderson, W. D., Duarte, F. M., Civelek, M. & Guertin, M. J. Defining data-driven primary transcript annotations with primaryTranscriptAnnotation in R. *Bioinformatics* (2020). doi:10.1093/bioinformatics/btaa011

11. Sheffield, N. C. Bulker: A multi-container environment manager. *OSF Preprints* (2019). doi:10.31219/osf.io/natsj

12. Core, L. J. *et al.* Analysis of nascent rna identifies a unified architecture of initiation regions at mammalian promoters and enhancers. *Nature genetics* **46,** 1311 (2014).

13. Duttke, S. H. C. *et al.* Human promoters are intrinsically directional. *Molecular cell* **57,** 674–684 (2015).

14. Sathyan, K. M. *et al.* An improved auxin-inducible degron system preserves native protein levels and enables rapid and specific protein depletion. **33,** 1441–1455 (2019).

15. Andersson, R. *et al.* Human gene promoters are intrinsically bidirectional. *Molecular cell* **60,** 346–347 (2015).

16. Choder, M. & Aloni, Y. RNA polymerase ii allows unwinding and rewinding of the dna and thus maintains a constant length of the transcription bubble. *Journal of Biological Chemistry* **263,** 12994–13002 (1988).

17. Shen, W., Le, S., Li, Y. & Hu, F. SeqKit: A cross-platform and ultrafast toolkit for fasta/q file manipulation. **11,** e0163962

18. Martins, A. fqdedup: Remove PCR duplicates from FASTQ files. (2018).

19. Daley, T. & Smith, A. D. Modeling genome coverage in single-cell sequencing. *Bioinformatics* **30,** 3159–3165 (2014).

20. Rougvie, A. E. & Lis, J. T. The rna polymerase ii molecule at the 5' end of the uninduced hsp70 gene of d. Melanogaster is transcriptionally engaged. **54,** 795–804 (1988).

21. Core, L. J. *et al.* Defining the status of rna polymerase at promoters. *Cell reports* **2,** 1025–1035 (2012).

22. Furey, T. S. ChIP-seq and beyond: New and improved methodologies to detect and characterize protein-dna interactions. *Nature reviews. Genetics* **13,** 840–852 (2012).

23. Love, M. I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome biology* **15,** 550 (2014).

24. Sheffield, N. C. Pypiper: A python toolkit for building restartable pipelines. (2017).

25. Stolarczyk, M., Reuter, V. P., Magee, N. E. & Sheffield, N. C. Refgenie: A reference genome resource manager. *Gigascience* (2020). doi:10.1101/698704

26. Martin, M. Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet.journal* **17,** 10–12 (2011).

27. Magoc, T. & Salzberg, S. L. FLASH: Fast length adjustment of short reads to improve genome assemblies. *Bioinformatics* **27,** 2957–2963 (2011).

28. Edwards, R. & Edwards, J. A. Fastq-pair: Efficient synchronization of paired-end fastq files. *BioRxiv* 552885 (2019).

29. Langmead, B. & Salzberg, S. L. Fast gapped-read alignment with bowtie 2. **9,** 357–359 (2012).

30. Martins, A. L., Walavalkar, N. M., Anderson, W. D., Zang, C. & Guertin, M. J. Universal correction of enzymatic sequence bias reveals molecular signatures of protein/dna interactions. *Nucleic acids research* **46,** e9 (2018).

31. Kurtz, S., Narechania, A., Stein, J. C. & Ware, D. A new method to compute k-mer frequencies and its application to annotate large repetitive plant genomes. *BMC genomics* **9,** 517 (2008).

Supplemental text

Gene counts table

PEPPRO provides a project level counts table that simplifies downstream analyses. Here, we import the PEPPRO project counts table and construct a DESeq data set in a few lines of code.

```
library(PEPPROr)
prj = Project("peppro_paper.yaml")
counts = read.csv(file.path(paste0(config(prj)$metadata$output_dir,
                            "/summary/PEPPRO_countData.csv")))
counts = counts[,c("geneName", "H9_PRO-seq_1", "H9_PRO-seq_2", "H9_PRO-seq_3",
                   "H9_treated_PRO-seq_1", "H9_treated_PRO-seq_2",
                   "H9_treated_PRO-seq_3")]
count_matrix = as.matrix(counts[,-"geneName"])
rownames(count_matrix) = counts$geneName
coldata = data.frame(condition=c(rep("untreated", 3), rep("treated", 3)))
rownames(coldata) = colnames(count_matrix)
library("DESeq2")
dds = DESeqDataSetFromMatrix(countData = count_matrix,
                             colData = coldata,
                             design = ~ condition)
featureData = data.frame(gene=rownames(count_matrix))
mcols(dds) = DataFrame(mcols(dds), featureData)
```

Supplemental figures

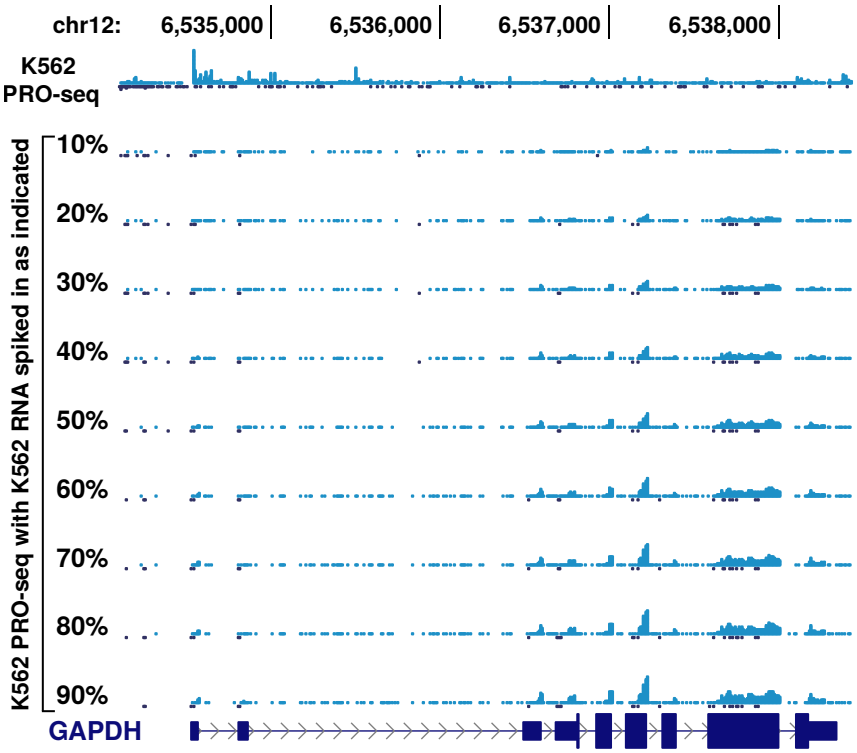

Fig. S1: ***K562 RNA-seq spike-in signal tracks show increasing exonic coverage.*** *GAPDH exonic coverage is enriched as the percentage of RNA-seq reads increases, and is visualized particularly well at exons 6 and 8.*

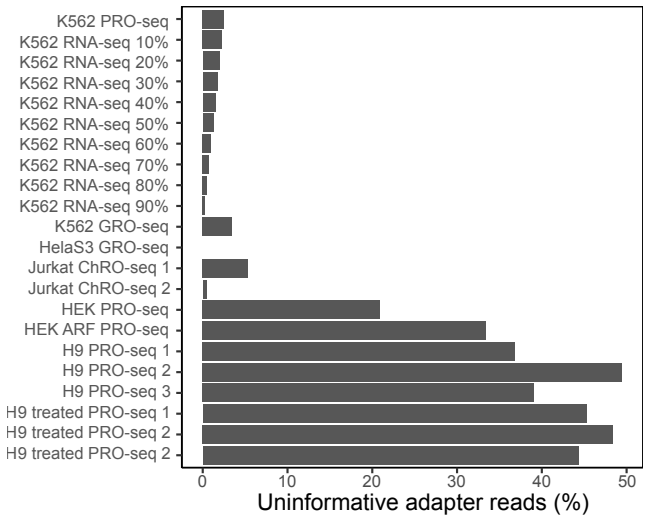

Fig. S2: ***Percentage of uninformative adapter reads following adapter removal for test set samples.*** *The HEK and H9 libraries contain more adapter-adapter reads because PAGE-mediated size selection was excluded from the protocol.*

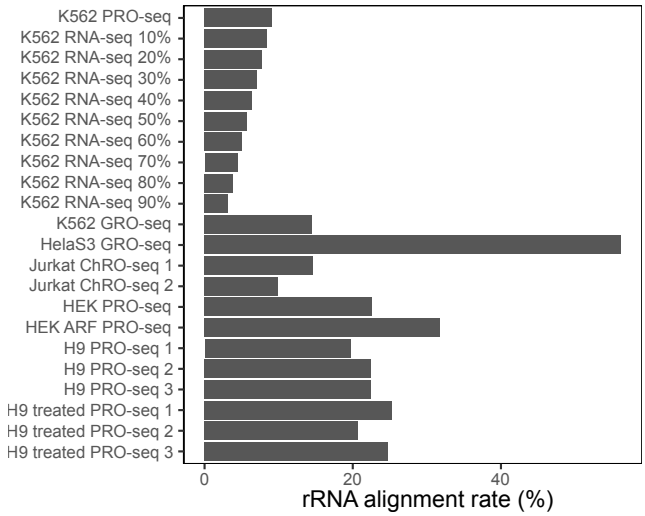

Fig. S3: ***Ribosomal DNA alignment rates for test set samples***. *The HelaS3 GRO-seq sample is highly enriched for ribosomal RNA transcripts compared to other test samples.*

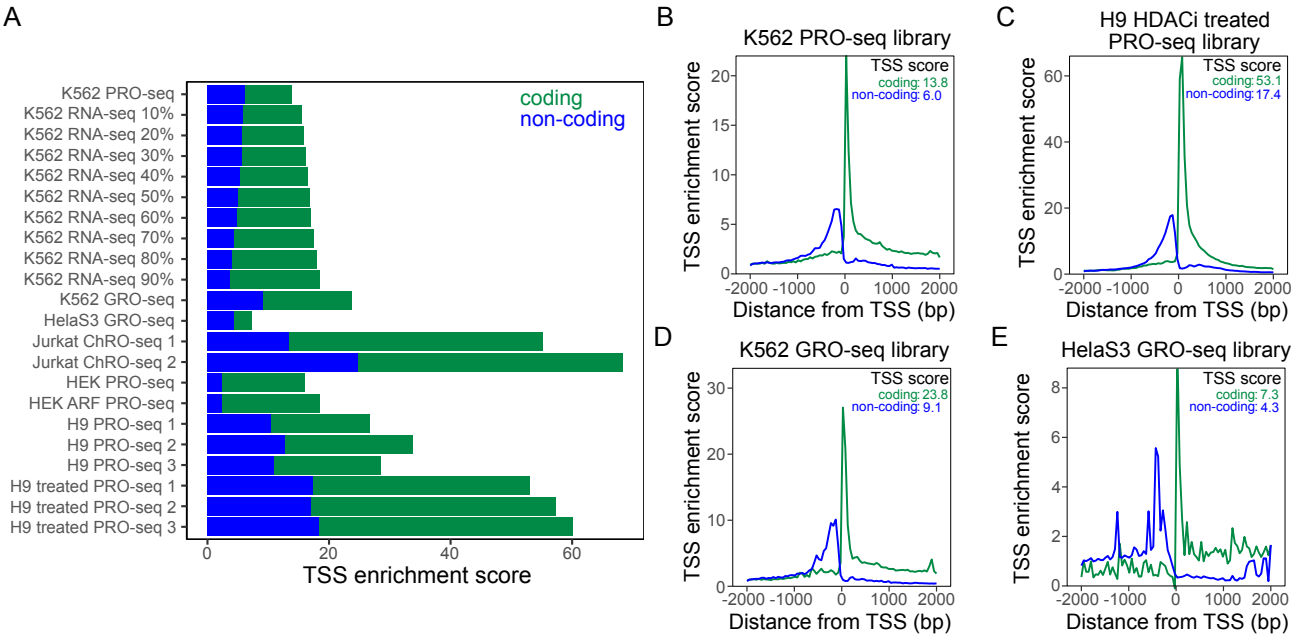

Fig. S4: ***TSS enrichment***. *A) TSS enrichment scores for test set samples. B) Representative high-quality PRO-seq TSS enrichment plot. C) TSS enrichment plot in romidepsin treated PRO-seq library. D) Representative high quality GRO-seq TSS enrichment plot E) Reprentative example of lower quality GRO-seq TSS enrichment plot.*

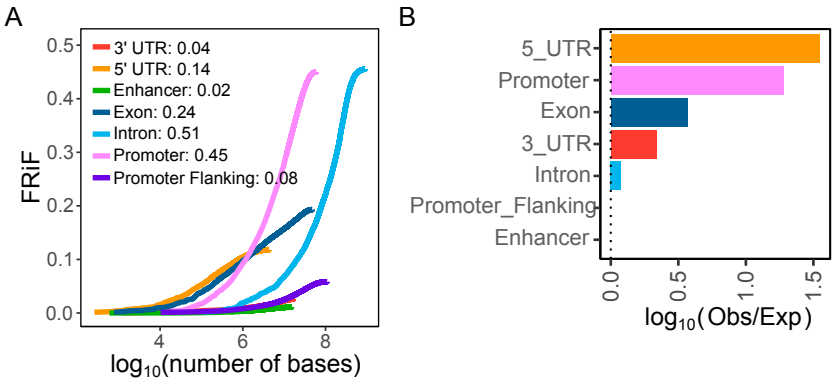

Fig. S5: ***Fraction of Reads in Features in ChRO-seq***. *A) Cumulative FRiF and B) FRiF plots for example Jurkat ChRO-seq 1 library test sample shows increased enrichment of promoter sequences.*

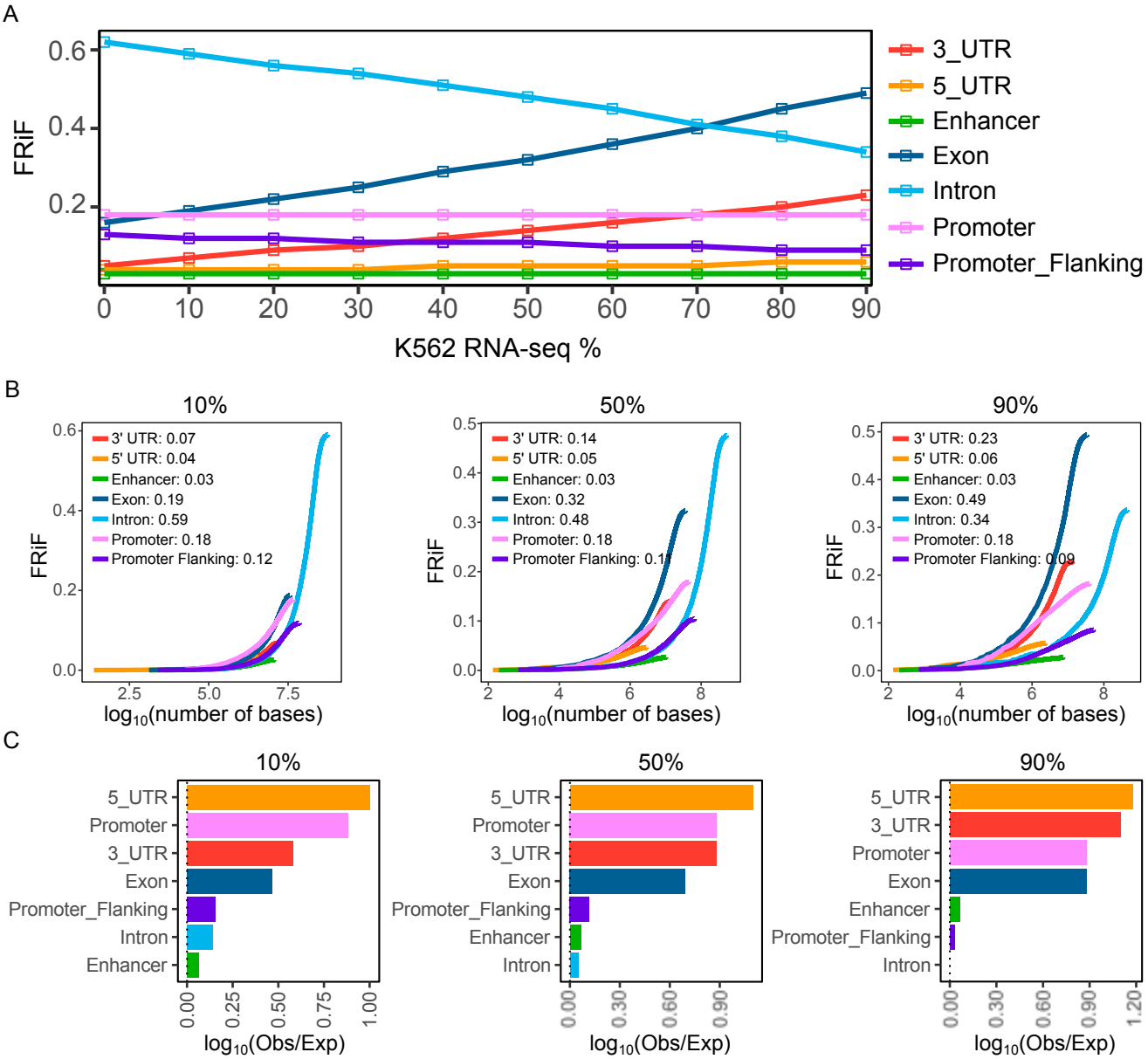

Fig. S6: ***RNA-seq spike-in test samples.*** *A) Increasing percentages of RNA-seq spike-in lead to changes in the fraction of reads in features (FRiF). B) Cumulative FRiF plots at 10%, 50%, and 90% RNA-seq spike-in. C) The expected versus observed fraction of reads in genomic features at 10%, 50%, and 90% RNA-seq spike-in.*