RNA-Sequencing

Introduction

RNA sequencing (RNA-seq) has revolutionized molecular biology, offering a powerful and high-throughput approach to study the transcriptome. By capturing a detailed snapshot of gene expression, RNA-seq allows researchers to explore not only which genes are active but also how they are regulated—through alternative splicing, post-transcriptional modifications, and the roles of non-coding RNAs. This technology has become indispensable in drug discovery, where it helps uncover new drug targets, unravel the complexities of disease mechanisms, and predict how patients might respond to treatments, including identifying patterns of drug resistance.

What makes RNA-seq particularly valuable is its ability to provide unbiased, quantitative data about the entire transcriptome, enabling researchers to see the bigger picture of cellular activity. This has made it a key tool in the push toward precision medicine, where therapies are tailored to individual patients based on their unique genetic makeup.

In this review, we'll dive into how RNA-seq works, from the technical aspects of the methodology to the statistical models used to analyze the data—such as DESeq2 for identifying differentially expressed genes. We'll also explore its transformative impact on drug discovery and personalized medicine, highlighting how this technology is shaping the future of healthcare.

RNA-Seq methodology

Library preparation and sequencing platforms

The RNA-seq process kicks off with the extraction of RNA from sources like tissues, cell cultures, or biofluids such as blood or cerebrospinal fluid. To ensure the RNA is free from contamination, it's treated with DNase to remove any lingering genomic DNA. The quality of the RNA is then checked using tools like the Bioanalyzer or TapeStation, which provide an RNA Integrity Number (RIN). Generally, samples with a RIN above 7 are considered good to go for RNA-seq.

Next, the focus shifts to isolating messenger RNA (mRNA). This is often done using poly(A) selection, where oligo-dT beads pull out mRNA by binding to their poly-A tails. For a broader view that includes non-coding RNAs, ribosomal RNA (rRNA) depletion methods like Ribo-Zero or NEBNext can be used instead. Once the RNA is prepped, it's broken down into smaller pieces, around 200 base pairs long, using either heat and divalent cations or enzymes.

From there, the RNA fragments are converted into complementary DNA (cDNA). This involves two steps: first-strand synthesis using reverse transcriptase, followed by second-strand synthesis to create double-stranded cDNA. The cDNA is then prepped for sequencing—end-repaired, A-tailed, and fitted with sequencing adapters. These adapters often include unique molecular identifiers (UMIs), which help correct errors and reduce biases later in the process. Finally, the library is amplified using PCR and ready for high-throughput sequencing on platforms like Illumina's NovaSeq or HiSeq, PacBio, or Oxford Nanopore.

This workflow, while technical, is the backbone of RNA-seq, enabling researchers to explore the transcriptome in incredible detail.

Read Alignment and Quality Control

Once the sequencing is done, the resulting reads—usually between 50 and 150 base pairs long—are put through a quality check. Tools like FastQC are used to evaluate the reads, looking at factors like base quality, GC content, and whether there's any adapter contamination. If low-quality reads are spotted, they're trimmed away using software like Trimmomatic or Cutadapt to clean up the data.

Next, the cleaned-up reads are aligned to a reference genome. This step uses specialized tools called splice-aware aligners, which are designed to handle the complexities of RNA splicing. Some popular options include:

- STAR (Spliced Transcripts Alignment to a Reference): Known for its speed and efficiency, especially with longer reads.

- HISAT2: A memory-efficient aligner that's great for large datasets.

- Salmon: This tool takes a different approach, using pseudo-alignment for fast transcript quantification without needing full alignment.

After alignment, it's important to check the quality of the results. Metrics like the mapping rate (how many reads aligned successfully), duplication rate (how many reads are redundant), and transcript coverage uniformity (how evenly the reads are distributed across the transcript) are assessed. Reads that map to multiple locations—called multi-mapped reads—are either discarded or assigned to the most likely location based on alignment confidence scores.

This careful quality control ensures that the data is reliable and ready for downstream analysis, like identifying differentially expressed genes or exploring transcript isoforms.

Gene Expression Quantification

Gene expression levels are quantified using two primary strategies:

1. Raw Read Counting – Counting the number of reads mapped to each gene or transcript.
2. Transcript Per Million (TPM) and Fragments Per Kilobase Million (FPKM) – Normalized measures that account for transcript length and sequencing depth.

Gene-level quantification is typically performed using HTSeq or FeatureCounts, while transcript-level quantification can be done using Salmon or Kallisto.

Differential Gene Expression Analysis Using Pydeseq2

PyDESeq2 is a Python implementation of the widely used DESeq2 package, designed for differential gene expression analysis in RNA-seq data. RNA-seq experiments measure gene expression levels by counting the number of reads (sequences) that align to each gene in a genome. PyDESeq2 helps researchers identify genes that are differentially expressed between experimental conditions, such as drug-treated versus control samples, while accounting for technical and biological variability.

At its core, PyDESeq2 uses statistical models to analyze count data, which is inherently noisy and variable. By modeling this data appropriately, PyDESeq2 can provide reliable insights into which genes are significantly affected by the experimental conditions being studied.

The Count Matrix: Starting Point of Analysis

The analysis begins with a **count matrix**, which is a table where each row represents a gene, each column represents a sample, and the values are the raw read counts for each gene in each sample. These counts reflect how often a gene was "read" during sequencing. For example, if you're studying the effect of a drug on cells, your count matrix might include samples from both drug-treated and untreated (control) groups.

In Python, the count matrix is typically represented as a Pandas DataFrame or a NumPy array. It's important to ensure that the count matrix is properly formatted, with genes as rows and samples as columns, and that it includes all the samples you plan to analyze.

The Design Matrix: Defining the Experiment

The **design matrix** is a critical component of the analysis because it specifies the structure of your experiment. It tells PyDESeq2 which factors or variables to consider when modeling the data. For example, in a drug response study, you might have two main factors:

1. **Condition**: Whether a sample is from the drug-treated group or the control group.
2. **Batch**: Technical variation, such as differences in sequencing runs or sample preparation batches, which can introduce unwanted noise into the data.

In PyDESeq2, you define the design matrix using a formula-like syntax. For instance:

design = "~ condition + batch"

This formula tells PyDESeq2 to model the data based on the condition (e.g., drug-treated vs. control) while accounting for the batch effect. By including batch in the design, you can control for technical variability and focus on the biological differences of interest.

Statistical Modeling: How PyDESeq2 Works

PyDESeq2 uses a **negative binomial distribution** to model the raw counts. This distribution is well-suited for RNA-seq data because it accounts for overdispersion, which is the extra variability often observed in count data beyond what a simpler Poisson distribution can handle.

To improve the accuracy of the model, PyDESeq2 uses **shrinkage estimation** (via empirical Bayes methods) to stabilize the dispersion and fold change estimates. This is particularly important for genes with low counts, which can have highly variable estimates without shrinkage.

Output: Log2 Fold Changes and P-values

After fitting the model, PyDESeq2 calculates two key metrics for each gene:

1. **Log2 fold changes**: These indicate how much a gene's expression changes between conditions (e.g., drug-treated vs. control). A log2 fold change of 1 means the gene's expression has doubled in the treated group compared to the control.
2. **P-values**: These tell you whether the observed changes are statistically significant. A low p-value suggests that the difference in expression is unlikely to be due to random chance.

Because RNA-seq experiments typically involve testing thousands of genes simultaneously, PyDESeq2 adjusts the p-values using the **Benjamini-Hochberg method** to control the **false discovery rate (FDR)**. This adjustment helps reduce the likelihood of false positives, ensuring that the results are more reliable.

Sample Size and Power: Planning Your Experiment

Before conducting an RNA-seq experiment, it's important to estimate the **sample size** needed to detect meaningful differences in gene expression. While PyDESeq2 itself doesn't directly perform power analysis, you can use other Python tools (e.g., statsmodels or custom simulations) to estimate the minimum number of samples required to achieve a desired level of statistical power and control the FDR.

Power analysis helps ensure that your experiment is well-designed and has a high likelihood of detecting true biological differences. Without adequate sample size, you risk missing important findings or producing unreliable results.

Why Experimental Design Matters

Proper experimental design is crucial for obtaining meaningful results. By carefully defining your design matrix and accounting for factors like batch effects, you can ensure that PyDESeq2 accurately models the effects of interest (e.g., drug treatment) while controlling for confounding variables. This leads to more robust and interpretable results.

Additionally, understanding the statistical principles behind PyDESeq2—such as the use of negative binomial models and shrinkage estimation—helps you appreciate why the tool is so effective at handling the complexities of RNA-seq data.

RNA sequencing has revolutionized drug discovery, offering deep insights into gene expression, cellular function, and disease mechanisms. It's already transforming precision medicine through applications like target identification, biomarker discovery, and understanding drug resistance. While challenges remain in data interpretation and technical complexity, combining RNA-seq with multi-omics, AI, and single-cell technologies is unlocking new possibilities. As RNA-seq becomes more affordable and accessible, its role in drug development and personalized medicine will only expand, shaping the future of healthcare.

References

1. Mortazavi, A., Williams, B. A., McCue, K., Schaeffer, L., & Wold, B. (2008). Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature Methods*, 5(7), 621-628.
2. Liu, L., Li, Y., Li, S., Hu, N., He, Y., Pong, R., ... & Law, M. (2016). Comparison of next-generation sequencing systems. *Journal of Biomedicine and Biotechnology*, 2012, 251364.
3. Dobin, A., Davis, C. A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., ... & Gingeras, T. R. (2013). STAR: ultrafast universal RNA-seq aligner
4. Shaw, A. T., et al. (2013). Crizotinib in ROS1-rearranged non-small-cell lung cancer. *New England Journal of Medicine*, 368(9), 863-873.
5. Parker, J. S., et al. (2009). Supervised risk predictor of breast cancer based on intrinsic subtypes. *Journal of Clinical Oncology*, 27(8), 1160-1167.
6. Gautschi, O., et al. (2016). Adaptive resistance in melanoma. *Cancer Discovery*, 6(3), 221-223.
7. Sade-Feldman, M., et al. (2018). Resistance to checkpoint blockade. *Nature*, 562(7725), 568-573.