

Assessing the Impact of Different Tools on Transcriptomic Analysis Methods

Prasanth Kumar Thuthika

Background

Exposure to particulate matter (PM) especially fine particles measuring 2.5 micrometers or less (PM_{2.5}) has been linked to adverse respiratory outcomes, in part by perturbing gene expression in bronchial epithelial cells (Huang et al., 2021). Examining these changes at the transcriptomic level is critical for elucidating the molecular mechanisms underpinning pollutant-induced health effects. The transcriptome refers to the total collection of RNA molecules, both coding and non-coding, present in a given cell or tissue at a specific time (Conesa et al., 2016; Wang, Gerstein, & Snyder, 2009). Because RNA profiles can shift rapidly in response to environmental stressors, transcriptomic analyses have become indispensable for identifying differentially expressed genes and novel isoforms relevant to disease pathology (Mortazavi, Williams, McCue, Schaeffer, & Wold, 2008; Stark, Grzelak, & Hadfield, 2019).

High-throughput RNA sequencing (RNA-seq) is now the method of choice for capturing transcriptomic complexity, surpassing older techniques like microarrays in both sensitivity and resolution (Wang et al., 2009). However, the downstream computational workflows employed in RNA-seq data analysis are neither uniform nor universally agreed upon. Researchers commonly face decisions regarding read preprocessing (e.g., adapter trimming, removal of low-quality bases), transcriptome alignment or pseudoalignment, and quantification. Moreover, each of these steps can be performed using different tools each with varying speed, memory requirements, and sensitivity (Trapnell et al., 2010). Debate also persists over whether certain preprocessing steps such as removing overrepresented sequences are strictly necessary or risk discarding meaningful biological signals (Stark et al., 2019). These methodological discrepancies can influence the final outcome of transcriptome analyses, underscoring the importance of carefully evaluating multiple workflows to ensure reliable, reproducible insights into how PM_{2.5} exposure influences gene expression.

Objective

Numerous workflows have been proposed to analyse the transcriptome and fully capture the information carried by transcripts. However, choosing a single “best” workflow can be challenging, as many tools that claim superior performance may offer limited flexibility for customization, require significant computational power, or fail to fully uncover the nuances of transcript data. Although various “gold standard” tools are widely used in the field, newer pipelines—often inspired by these established methods—promise faster run times and reduced storage needs. Meanwhile, researchers continue to debate whether certain preprocessing steps are truly necessary for reliable downstream analyses, leading to inconsistencies in best practices. Ultimately, the selection of a workflow depends heavily on the specific research question and the nature of the data, making it difficult to define a universal standard.

This project aims to examine existing claims about which workflows and tools perform best under different computational, storage, and analytical constraints. Using data from the study titled *“Effect of concentration and duration of particulate matter exposure on the transcriptome and DNA methylome of bronchial epithelial*

cells,” we will compare our workflow’s outcomes to the original paper’s results as a form of validation. By employing different tools and adjusting various analytical steps, we seek to determine whether workflow choice and tool selection significantly influence the final findings. Our evaluation will clarify whether—when compromises in computation or storage are made—the overall quality of the analysis remains consistent with previously published conclusions.

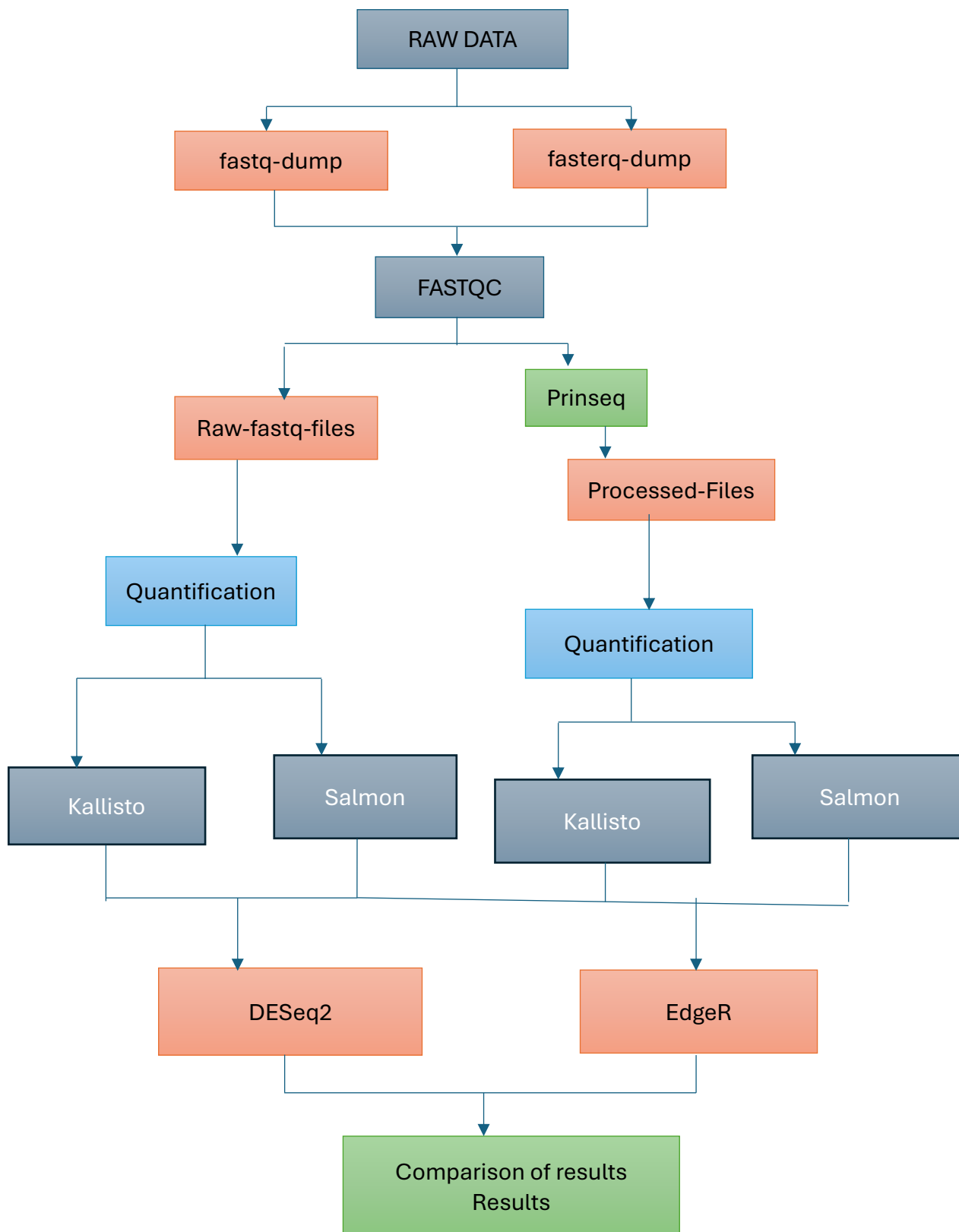
Data

The transcriptomic dataset for this project was obtained from the study by Huang et al. (2021) from the paper “*Effect of concentration and duration of particulate matter exposure on the transcriptome and DNA methylome of bronchial epithelial cells*” in which BEAS-2B bronchial epithelial cells were exposed to different doses of PM_{2.5} (1 or 30 µg/cm²) under acute (24-hour) and chronic (seven-day) conditions, vehicle-treated cells served as controls, and each condition was tested in biological triplicates. The raw RNA-Seq data, generated on an Illumina HiSeq 4000 with single-end reads, is publicly accessible in the NCBI Sequence Read Archive (SRA) under accession number SRP275647 aligned them to the human transcriptome (GRCh38), which was also obtained from Ensembl, using the corresponding annotation file. The sample types include Control_24h, Control_7d, PM_24_1dose, PM_24_30_dose, PM_7d_1_dose each three replicates.

Methodology

A variety of tools were used for this analysis, all of which were installed in a dedicated Conda environment, with most tools obtained from the Bioconda channel. These tools are open-source, and the differential expression analysis was performed in R. Further details on how these tools were utilized, along with the complete analysis workflow, are provided in the Results section. The flow chart below illustrates the workflow used to compare various steps in transcriptome analysis and assess the performance of different tools, highlighting how subsequent downstream analyses can be influenced by each step.

The data was downloaded and converted using two methods, followed by quality control for both sets of files. One set was left as raw files, while the other was processed using the Prinseq tool. Various tools were then employed for quantification, and the resulting files from each tool were used for differential expression analysis using different packages. The results were compared to understand the performance of the tools and the impact on the outcomes.



Results

The raw data were obtained from the NCBI Sequence Read Archive (SRA) using the SRA Toolkit, a reliable method for downloading large sequencing datasets. The *.sra* files were converted to FASTQ using either *fastq-dump* or *fasterq-dump*, with the latter demonstrating notably faster performance which has an advantage of using the multi-threading option for faster conversion. To assess data quality, FastQC <https://github.com/s-andrews/FastQC.git> was employed, generating an HTML report that provides detailed metrics and visual

summaries, while there are many other tools to assess the quality of the NGS data, FastQC is one of the most widely used for its comprehensive reporting. MultiQC <https://github.com/MultiQC/MultiQC.git> was subsequently used to aggregate these individual quality reports into a single file for easier comparison.

The FastQC report for one of the samples (others followed a similar pattern; see Figure 1) showed mostly green check marks, indicating a “pass” status, except for sequence duplication levels and per-base sequence content, sometimes the quality control reports can include multiple “fail” or “warn” flags that prompt mandatory preprocessing, often guided by the specific issues highlighted in the report. While some researchers view these artifacts as benign, others argue that they may affect downstream analyses. To assess their impact, we ran two parallel workflows: one using the raw files and another using data pre-processed with Prinseq <https://github.com/uwb-linux/prinseq.git> to remove overrepresented sequences. Fig 2 illustrates the FastQC report after preprocessing, which also applied to other samples. The sequence duplicates were removed after the pre-processing and it passed the check and while the per base sequence content passed warn from fail and suggests the files are different from the raw files. Because this step is frequently omitted in some pipelines, our aim was to determine whether these artifacts significantly influence subsequent analyses. In some cases, sequencing data may contain adapter sequences that must be removed using popular tools such as Trimmomatic <https://github.com/usadellab/Trimmomatic.git>, TrimGalore <https://github.com/FelixKrueger/TrimGalore.git> , or Cutadapt <https://github.com/marcelm/cutadapt.git> , while low-quality bases can be trimmed with fastp <https://github.com/OpenGene/fastp.git> , the FASTX Toolkit https://github.com/agordon/fastx_toolkit.git , or other similar programs.

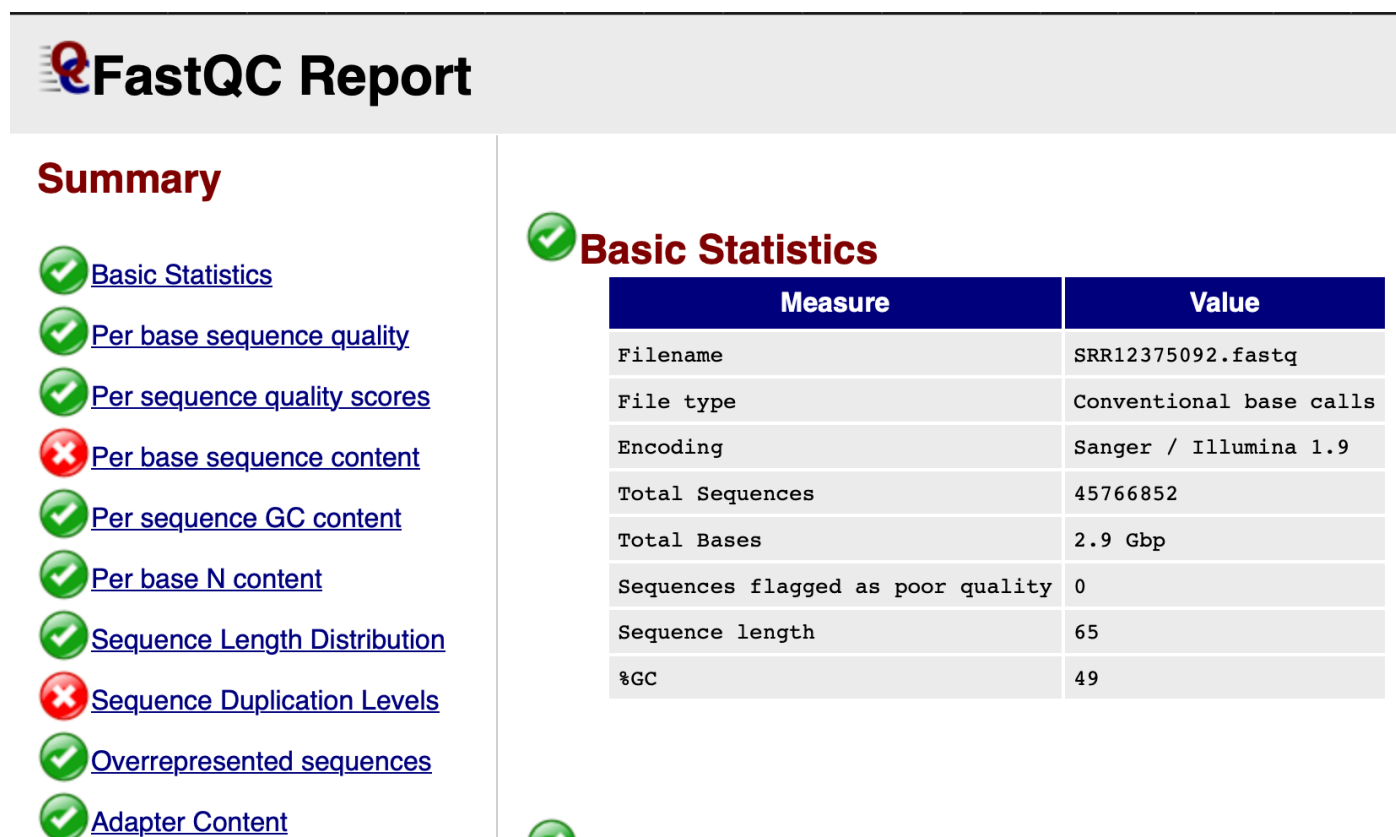












Fig:1 FastQC report for raw fastq files

FastQC Report

Summary

-  [Basic Statistics](#)
-  [Per base sequence quality](#)
-  [Per sequence quality scores](#)
-  [Per base sequence content](#)
-  [Per sequence GC content](#)
-  [Per base N content](#)
-  [Sequence Length Distribution](#)
-  [Sequence Duplication Levels](#)
-  [Overrepresented sequences](#)
-  [Adapter Content](#)

Basic Statistics

Measure	Value
Filename	SRR12375092_filtered.fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	19202769
Total Bases	1.2 Gbp
Sequences flagged as poor quality	0
Sequence length	65
%GC	48

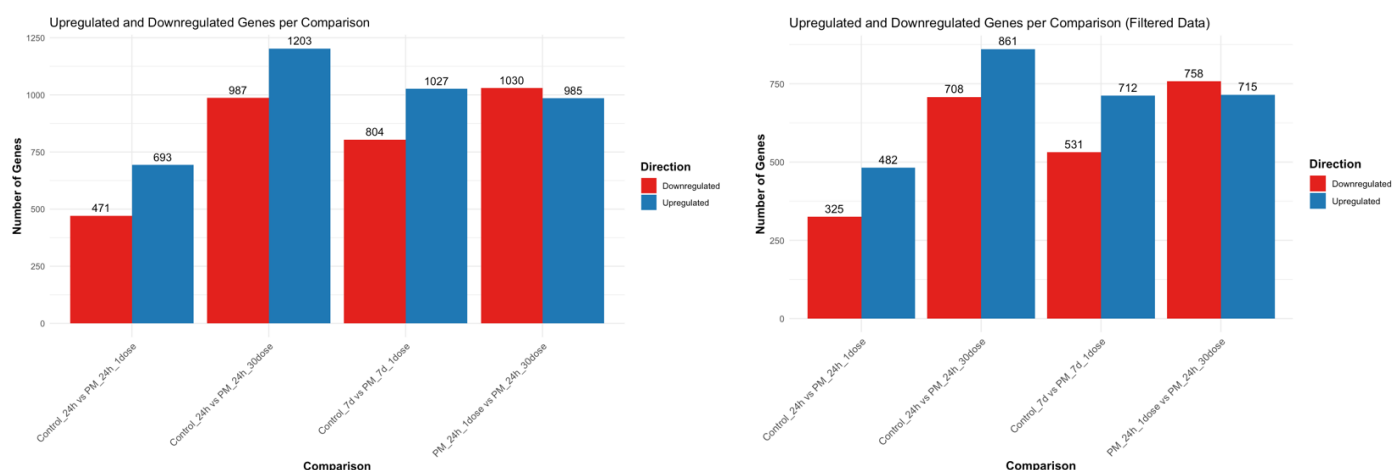
Fig:2 FastQC report for Prinseq-processed fastq files

While the primary goal was differential expression analysis rather than the identification of novel transcripts, the next step involved transcript quantification using Kallisto and Salmon. Each tool required indexing of the transcriptome before quantification could proceed. Among the two, Salmon took the longest time to process, whereas Kallisto completed its run in significantly less time, and Kallisto consumed the most memory during the indexing step while the other used less. Subsequent downstream analyses will help determine which tool offers the most advantages for accurate and efficient transcript quantification. The quantification of the transcripts was done in different ways and each serve different purposes. Transcript quantification commonly uses RPKM (reads per kilobase per million), FPKM (fragments per kilobase per million), or TPM (transcripts per million) to account for gene length and sequencing depth, facilitating more accurate comparisons of expression levels under diverse experimental conditions (Mortazavi, Williams, McCue, Schaeffer, & Wold, 2008; Li & Dewey, 2011). While RPKM is frequently used for single-end data, FPKM extends this approach to paired-end data, and TPM modifies the calculation so that total transcript abundances sum to the same value across samples (Wagner, Kin, & Lynch, 2012). These normalization strategies help control for technical biases and enhance the reliability of cross-sample gene expression analyses.

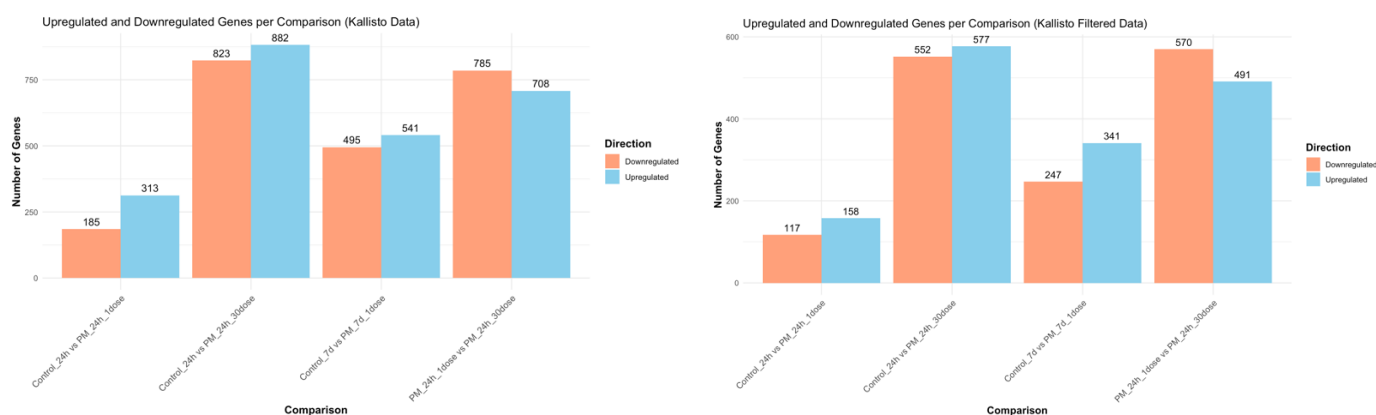
Kallisto and Salmon were used for quantification of both processed and raw RNA-Seq files to compare their accuracies and the time taken for quantification. It was observed that Salmon took approximately two more hours and consumed more memory compared to Kallisto. Both tools provided the count values, which were subsequently merged into one dataset. These merged datasets were then used for differential expression

analysis. The analysis was performed using two widely used and popular R libraries, DESeq2 and EdgeR, and the results from both methods were compared. (The detailed plots for the analysis are provided in the supplementary material. Most of the information regarding the plots and results is thoroughly summarized here, with a selection of notable plots included for comparison).

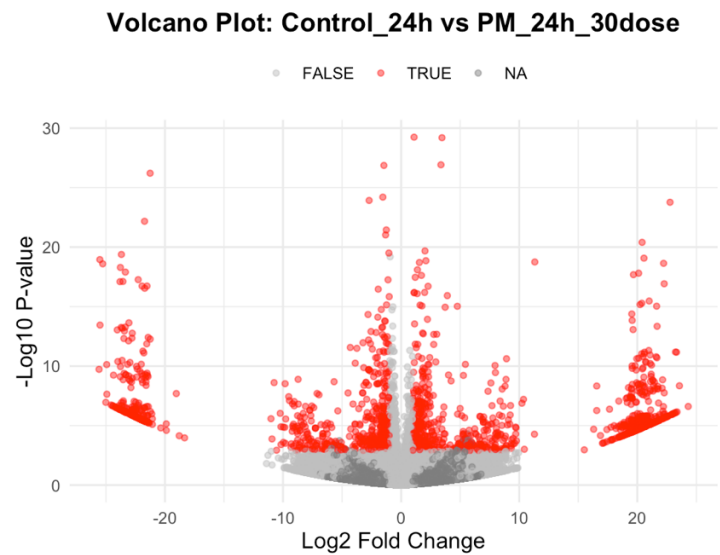
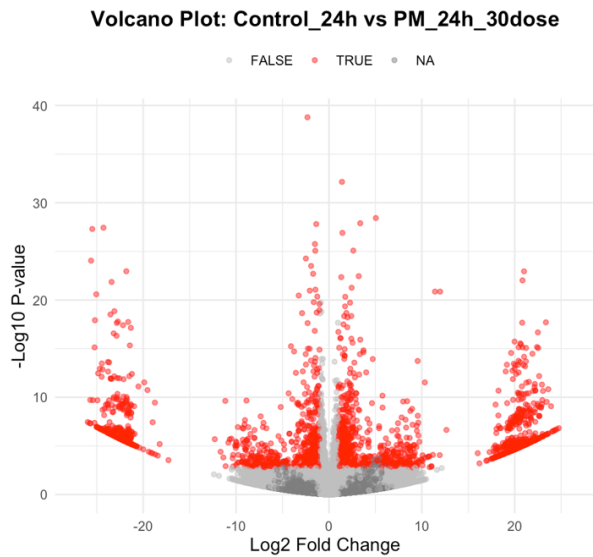
The differential expression analysis was performed using DESeq2 on both the processed and raw count files generated by the Salmon and Kallisto tools. Four comparisons were considered: Control_24h vs PM_24h_1dose, Control_24h vs PM_24h_30dose, Control_7d vs PM_7d_1dose, and PM_24h_1dose vs PM_24h_30dose. Among these, the **Control_24h vs PM_24h_30dose** condition was specifically analysed to evaluate the effects of data preprocessing and tool selection. The DESeq2 analysis results clearly highlight differences in differential expression between the filtered and raw samples. Additionally, the analysis results from Salmon and Kallisto also varied in terms of differential expression patterns. The figures below illustrate these variations in the analysis.



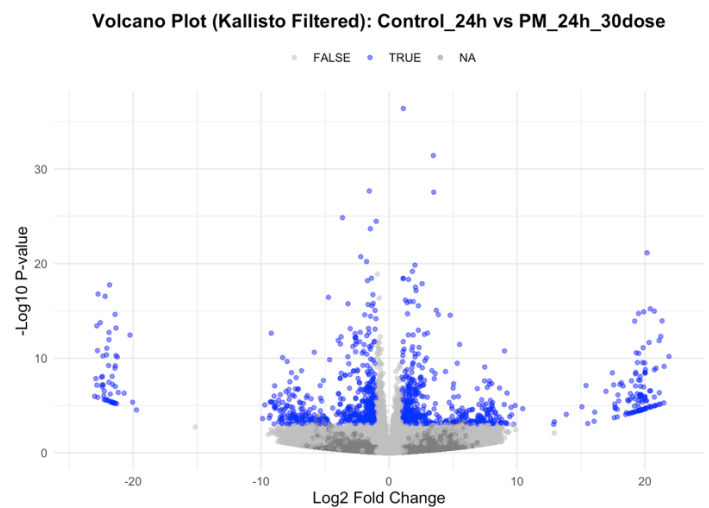
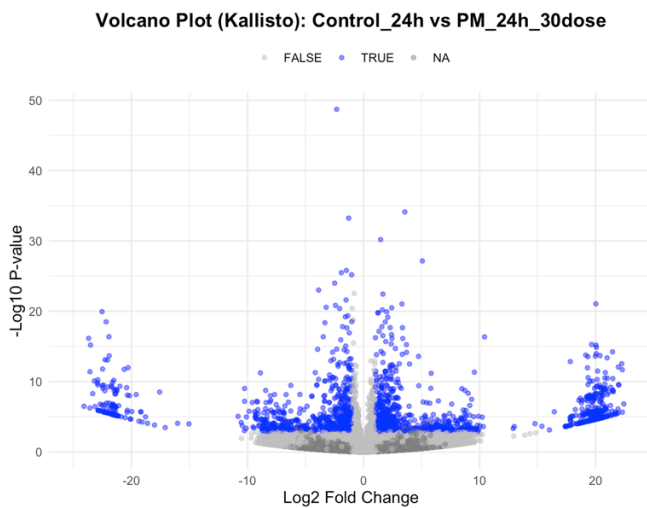
Figures illustrating the DESeq2 analysis of raw (left) and processed data (right) using count files generated by the Salmon tool.



Figures illustrating the DESeq2 analysis of raw (left) and processed data (right) using count files generated by the Kallisto tool.

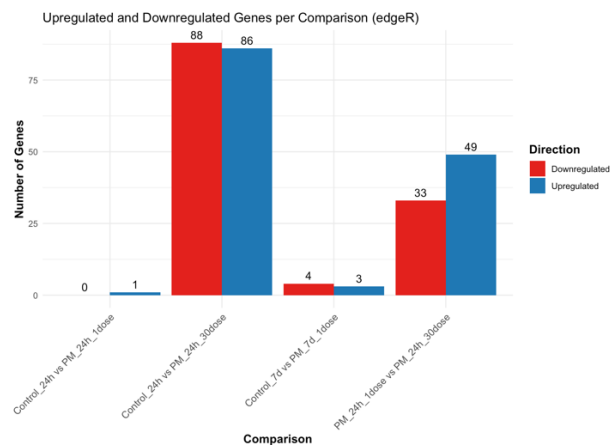
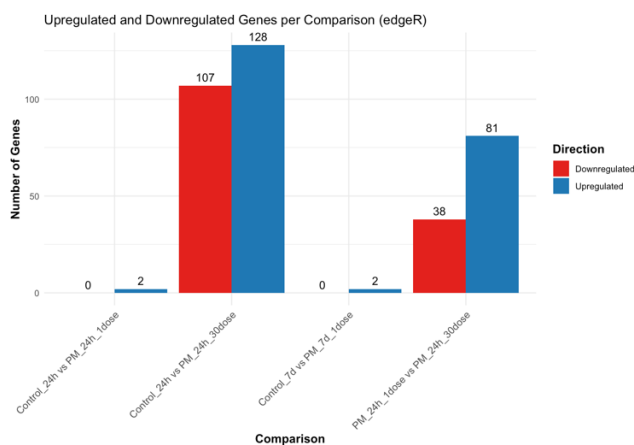


Volcano Plots illustrating the DESeq2 analysis of raw (left) and processed data (right) using count files generated by the Salmon tool.

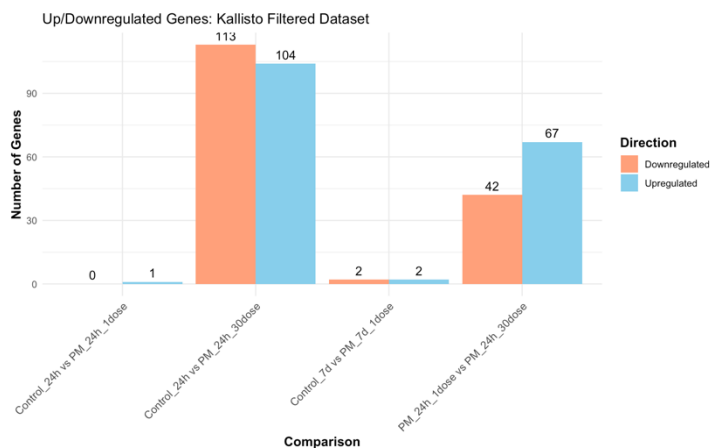
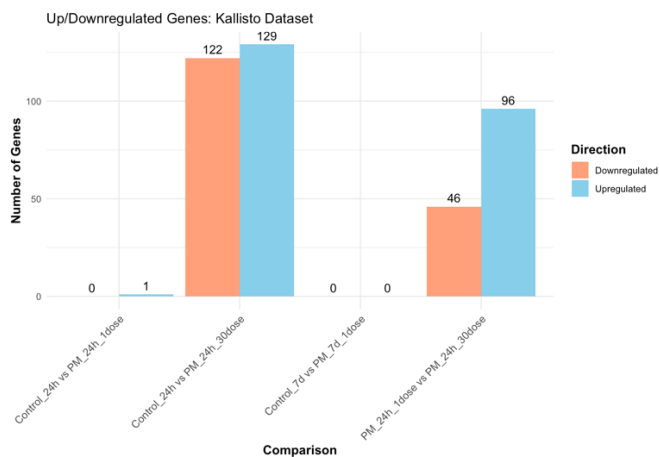


Volcano Plots illustrating the DESeq2 analysis of raw (left) and processed data (right) using count files generated by the Kallisto tool.

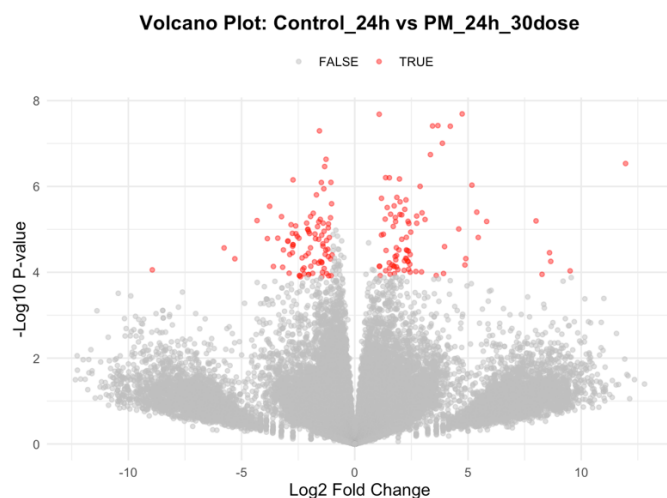
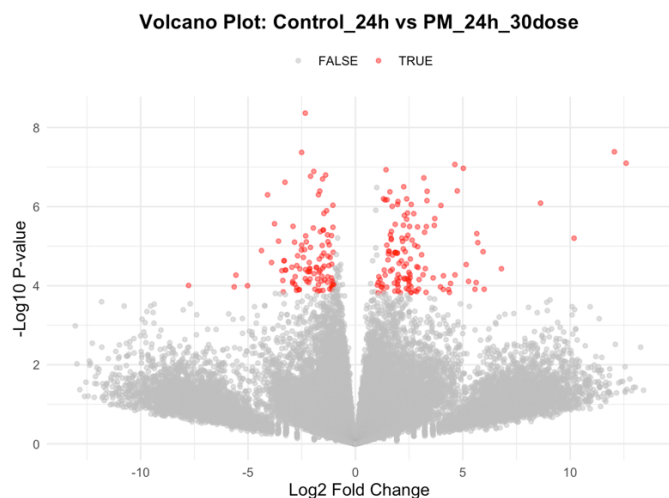
The same analysis was performed using edgeR to compare whether the results align with those generated by DESeq2. The findings indicate differences in the results for both the raw and processed data, as well as discrepancies with the DESeq2 outcomes. The figures below illustrate these differences.



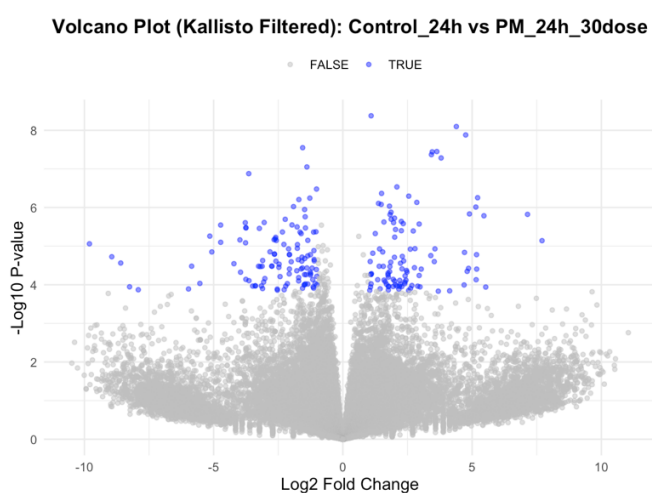
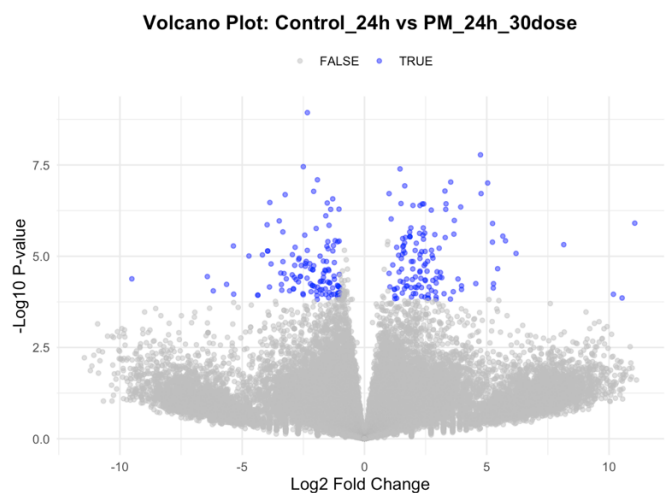
Figures illustrating the EdgeR analysis of raw (left) and processed data (right) using count files generated by the Salmon tool.



Figures illustrating the EdgeR analysis of raw (left) and processed data (right) using count files generated by the Kallisto tool.



Volcano Plots illustrating the EdgeR analysis of raw (left) and processed data (right) using count files generated by the Salmon tool.



Volcano Plots illustrating the EdgeR analysis of raw (left) and processed data (right) using count files generated by the Kallisto tool.

The question arises as to why the differential expression analysis using edgeR shows significantly fewer upregulated or downregulated genes compared to DESeq2 for the same dataset. Which method produces the correct results, and does the data processing step truly matter? It is evident that proper preprocessing is crucial for obtaining more accurate and reliable results.

Conclusion

Kallisto uses a pseudo-alignment method that maps reads to transcripts without full alignment, whereas Salmon employs quasi-mapping combined with a probabilistic model to handle multi-mapping reads and correct for sequencing biases such as GC content and positional biases, providing more accurate transcript quantification. Both tools generate estimated counts, but Salmon includes more diagnostic features and advanced bias correction options. While Kallisto excels in speed and simplicity, Salmon offers enhanced flexibility and accuracy at the cost of slightly increased computational requirements (Bray et al., 2016). DESeq2 and edgeR are widely used tools for differential gene expression analysis, but they differ in statistical methods and normalization approaches. DESeq2 employs a shrinkage-based approach for dispersion estimation and fold-change estimation, enhancing stability for genes with low counts, and normalizes counts using a median of ratios method. In contrast, edgeR models gene expression using negative binomial distributions with empirical Bayes methods for dispersion estimation, offering flexibility for small sample sizes. Differences in results can arise due to these variations in normalization and statistical modeling, which affect sensitivity to outliers, low-expression genes, and the handling of variance across replicates (Love et al., 2014; Robinson et al., 2010). To address our objective of determining which tool performed better based on validation against the original results, none of the tools produced results that matched or closely resembled the original findings. This discrepancy could be due to the use of different techniques or methodologies (not explicitly detailed in the referenced paper) for expression analysis. Results are also likely to vary if alternative workflows or tools are used at different steps in the analysis pipeline. Therefore, there is a need to develop more robust and comprehensive pipelines to fully extract and interpret information from the transcriptome.

Discussion

This study evaluates commonly cited advantages of various tools and whether they truly perform as expected. However, the key factor lies in the specific research question one intends to address with the transcriptome data. Ultimately, the choice of tools depends on how quickly results are needed, as well as the available computing power and memory.

Data Availability

The data are publicly available from the NCBI Sequence Read Archive (SRA) under accession ID **SRP275647**.

Supplementary Data

Detailed plots can be found in the supplementary materials within the repository, and an additional spreadsheet includes information on time, memory, and CPU usage for each tool.

Code Availability

Code is available at: <https://github.com/PrasanthKT/Transcriptome-Analysis.git>

References

1. Conesa, A., Madrigal, P., Tarazona, S., Gomez-Cabrero, D., Cervera, A., McPherson, A., ... & Mortazavi, A. (2016). A survey of best practices for RNA-seq data analysis. *Genome Biology*, 17(1), 13. <https://doi.org/10.1186/s13059-016-0881-8>
2. Huang, X., Li, Y., & Zhang, C. (2021). *Effect of concentration and duration of particulate matter exposure on the transcriptome and DNA methylome of bronchial epithelial cells*. (Please insert specific journal citation and DOI once confirmed.)
3. 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. <https://doi.org/10.1038/nmeth.1226>
4. Stark, R., Grzelak, M., & Hadfield, J. (2019). RNA sequencing: the teenage years. *Nature Reviews Genetics*, 20(11), 631–656. <https://doi.org/10.1038/s41576-019-0150-2>
5. Trapnell, C., Williams, B. A., Pertea, G., Mortazavi, A., Kwan, G., van Baren, M. J., ... & Pachter, L. (2010). Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature Biotechnology*, 28(5), 511–515. <https://doi.org/10.1038/nbt.1621>
6. Wang, Z., Gerstein, M., & Snyder, M. (2009). RNA-Seq: a revolutionary tool for transcriptomics. *Nature Reviews Genetics*, 10(1), 57–63. <https://doi.org/10.1038/nrg2484>
7. Li, B., & Dewey, C. N. (2011). RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC Bioinformatics*, 12(1), 323. <https://doi.org/10.1186/1471-2105-12-323>
8. 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. <https://doi.org/10.1038/nmeth.1226>
9. Wagner, G. P., Kin, K., & Lynch, V. J. (2012). Measurement of mRNA abundance using RNA-seq data: RPKM measure is inconsistent among samples. *Theory in Biosciences*, 131(4), 281–285. <https://doi.org/10.1007/s12064-012-0162-3>
10. Bray, N. L., Pimentel, H., Melsted, P., & Pachter, L. (2016). Near-optimal probabilistic RNA-seq quantification. *Nature Biotechnology*, 34(5), 525–527. <https://doi.org/10.1038/nbt.3519>
11. Love, M. I., Huber, W., & Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, 15(12), 550. <https://doi.org/10.1186/s13059-014-0550-8>
12. Robinson, M. D., McCarthy, D. J., & Smyth, G. K. (2010). edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1), 139–140. <https://doi.org/10.1093/bioinformatics/btp616>