Leenah Hamdy
BF550
12/9/24

Project 2 Write Up

Introduction

The study “Comparative transcriptomic profiling reveals a role for Olig1 in promoting axon regeneration” by Fu et al. looked at regeneration of axons in mice, particularly focusing on finding genes that contribute to this regeneration. Through determining the mechanisms that facilitate axon regeneration, potential intervention targets could be identified for approaching the treatment of axon injury. Identifying these targets is particularly motivating because injuries to the adult mammalian central nervous system (CNS) typically leads to irreversible damage. CNS nerves do not have endogenous regeneration ability, and when paired with exogenous inhibitory factors, these neurons fail to regenerate on their own. Because axon regeneration is heterogeneous, with some types of axons having high regenerative capacity while other do not have this capacity, this study was motivated to determine genes that were highly upregulated in regenerative (R) spinal motor neurons (spMNs) compared to non-regenerative (G) spMNs. These neurons were identified as concentrations for this study as, while studies in the past only assumed heterogeneity across cells of different types, this study aimed to find another angle towards understanding molecular mechanisms for this regeneration by looking at heterogeneity of cells of the same type.

The cells from this study were collected from mice that had received a sciatic nerve crush injury. The spMNs were isolated and distinguished into the two groups based on their distinct regenerative abilities. Deep RNA-seq was performed on these cells, and used to generate the read counts matrix of detected genes for each type of cell, which was the data input for this project.

Methods

The RNA-seq read counts data in this study was analyzed using the iDEP (Integrated Differential Expression and Pathway Analysis) web application, which performed a differential expression analysis of this data and supplied many of the figures. The significance thresholds used for this analysis were indicated to be a mean expression level fold change (log2FoldChange) > 2 and an FDR-adjusted P-value < 0.1. iDEP is a R shiny application that uses the R DESeq2 package to perform differential expression analysis and supplies plots using ggplot2.

The plots I selected to recreate are figures 1E and S2C. I performed a differential expression analysis of the RNA-seq read counts data using the python pydeseq2 library. To create figure 1E, a volcano plot, I assigned significance thresholds of |log2FoldChange| > 1 and p-adjusted < 0.1 to accurately represent the significantly upregulated/downregulated figures indicated in the original plot. The p-adjusted were -log10 scaled and the data was plotted as a scatter plot using matplotlib.pyplot. To create figure S2C, a clustered heatmap, I sorted the results of the differential expression analysis by ascending p-adjusted and selected the top 1000 genes. I then normalized the read counts for this data using log(1+x) normalization to compress the range of values and reduce the influence of extreme outliers. Using seaborn clustermap, I plotted a heatmap using the Z-scores across the rows for the top 1000 differentially expressed genes.

Results

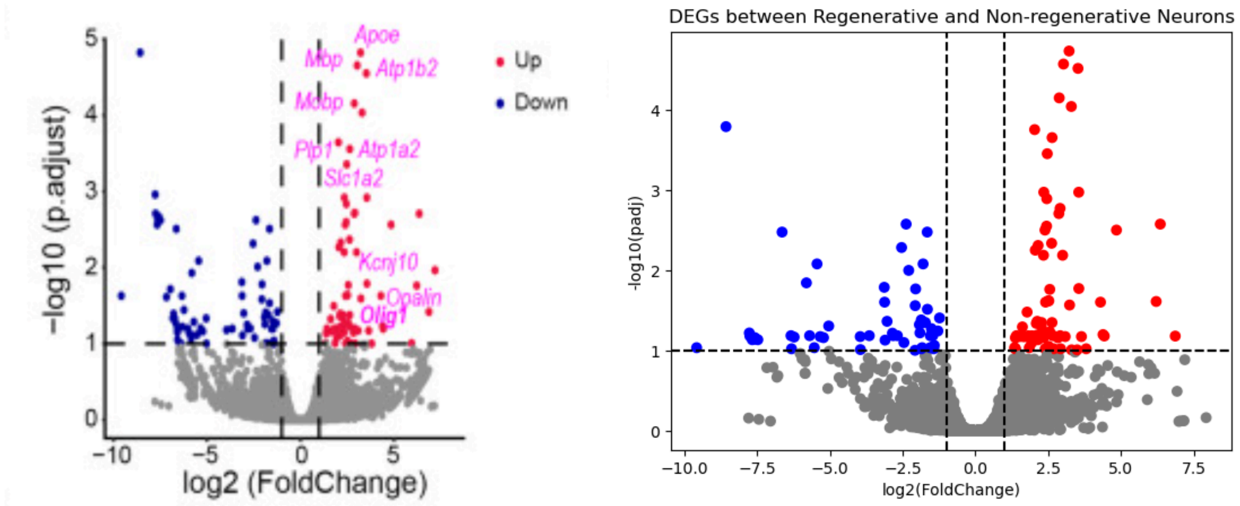

Figure 1: Original volcano plot (figure 1E) from the paper (left) created using R vs. recreation (right) created using python.

While I was able to yield similar results to the original paper (figure 1), the results do not line up perfectly. For example, one of the downregulated genes in the original plot has a -log10(padj) value of almost 5, in the original plot, while in the recreation this same gene has a -log10(padj) value of a little less than 4. This may be due to the very slight differences between the differential expression calculations that R's DESeq2 and python's pydeseq2 make. This may be due to the differences in how the different languages handle numerical calculations, particularly with the small values that require high precision that show up in differential expression analysis. When these differences are compounded through multiple steps of calculations, causing the overall results to be similar but not precisely the same. Using R's DESeq2 to run the differential expression analysis and inputting these results into my python plotting script, I was able to recreate the exact figure from the original paper (figure 2). Additionally, while the paper indicated in the methods section that it had used significance thresholds of |log2FoldChange| > 2 and p-adjusted < 0.1, using these thresholds did not yield a volcano plot that matched that of the original paper. Changing the log2FoldChange threshold to be greater than 1 instead yielded a more accurate result. I tested the volcano plotting on the iDEP tool and found that changing the thresholds using the provided menu did not change the coloring of the significantly differentially expressed genes on the plot. This indicates that the thresholds are hard coded on the back end of the software, so user input of what thresholds should be used for significance are not reflected in the plots. On the iDEP website, they indicate that the software has not been thoroughly tested and that results should be verified by further analysis on the part of the researcher.

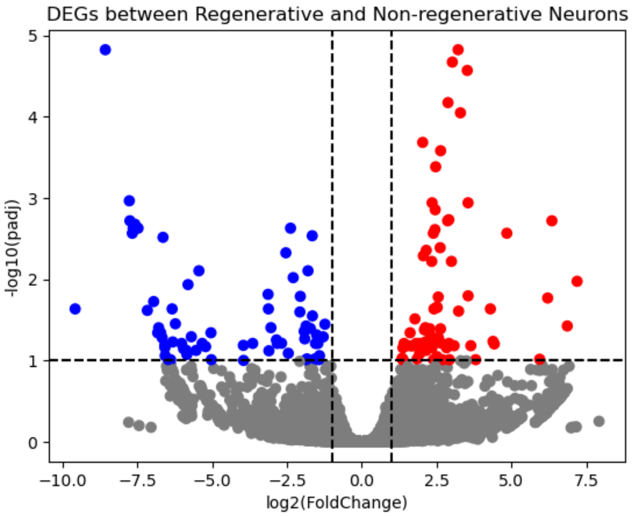

Figure 2: Volcano plot generated using R DESeq2 generated data.

Similarly, when plotting the heatmap, the metric for ranking the differentially expressed genes to pull the top 1000 genes was not indicated in the paper, so I ranked them using the FDR (p-adjusted) value. The recreated heatmap shows slightly different results than the original figure (figure 3), but the overall clustering remains consistent with the original findings of the paper. Particularly, because the ranking metric used in the original paper is unknown, small clusters of genes for each type of spMN may show up as being upregulated in the original paper and downregulated in the recreation, and vice versa. However, this may also be due to inconsistency in how the p-adjusted values were calculated in R as compared to python, as these values tend to be very small and require fine-tuned calculations, as mentioned previously. Because of this, python may have produced different rankings than R, particularly for the middle sections of the genes where the differences in how the genes show up as having a positive or negative Z-score are most obvious.

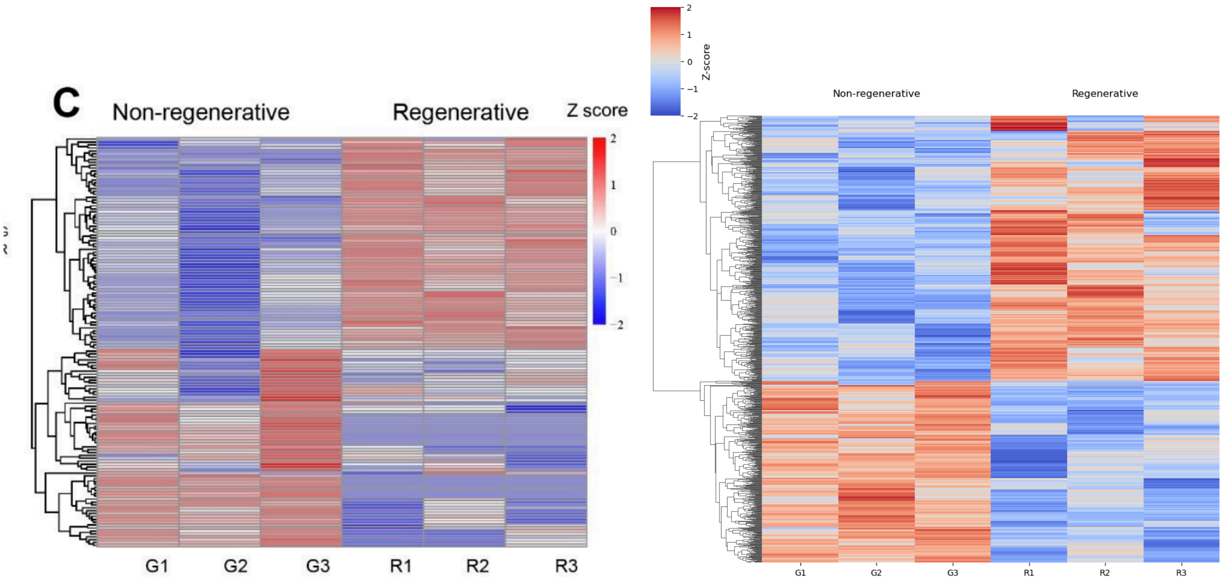

Figure 3: Original heatmap (supplementary figure 2C) from the paper (left) created using R vs. recreation (right) created using python.

Ultimately, because programming languages handle calculations with very small values differently, it can yield slightly different results when completing analysis of biological data. While these differences did not greatly affect the legitimacy of the findings of the study, researchers should be cautious when completing their analyses and may benefit from verifying their results using the different analysis tools available in R and Python. Additionally, when using online tools to analyze data, researchers should pay close attention to how it is being analyzed and make sure to report the accurate pipelines and thresholds being used so their results can be accurately reproduced.

Works Cited

Fu, X. Q., Zhan, W. R., Tian, W. Y., Zeng, P. M., & Luo, Z. G. (2024). Comparative transcriptomic profiling reveals a role for Olig1 in promoting axon regeneration. Cell reports, 43(7), 114514. https://doi.org/10.1016/j.celrep.2024.114514

Ge, S.X., Son, E.W. & Yao, R. (2018). iDEP: an integrated web application for differential expression and pathway analysis of RNA-Seq data. BMC Bioinformatics 19, 534. https://doi.org/10.1186/s12859-018-2486-6

Muzellec, B., Telenczuk, M., Cabeli, V., & Andreux, M. (2023). *PyDESeq2: a python package for bulk RNA-seq differential expression analysis*. PyDESeq2. 10.1093/bioinformatics/btad547

Data: https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE221115