DATA SCIENCE IN LIFE SCIENCE

Identify epigenetic alterations associated with Alzheimer's disease and classification of gene expressions between healthy and sick patients Project Week 04 - Group 02

Pushpa Koirala, Christina Kirschbaum and Melika Moradi

Full list of author information is available at the end of the article

1 Project

This week, we worked on the Data Analysis and started with the Machine learning part. The RNA-seq analysis is now mainly finished, with only minor tasks left. The ChIP-seq analysis further advanced and the final part of the is planned for the coming week. Finally, the Machine Learning was started.

Although we are still slightly behind our original timeline, we managed to catch up to it again.

2 RNA-seq Data Analysis

The RNA-sequence analysis was performed with the R package DESeq2. The group of the young, old and diseased groups were compared in several steps. The DESeq() function revealed in the summary, that there are huge differences in the gene regulation between the young and the diseased group. However, the differences between young and old were smaller than the differences between the old and the diseased group. This was the case for the STAR and the kallisto counts.

```
out of 23525 with nonzero total read count
adjusted p-value < 0.05
LFC > 0 (up)       : 62, 0.26%
LFC < 0 (down)     : 358, 1.5%
outliers [1]       : 0, 0%
low counts [2]     : 10950, 47%
(mean count < 86)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

(a) Young vs Old

```
out of 23525 with nonzero total read count
adjusted p-value < 0.05
LFC > 0 (up)       : 408, 1.7%
LFC < 0 (down)     : 411, 1.7%
outliers [1]       : 0, 0%
low counts [2]     : 2285, 9.7%
(mean count < 2)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

(b) Old vs AD

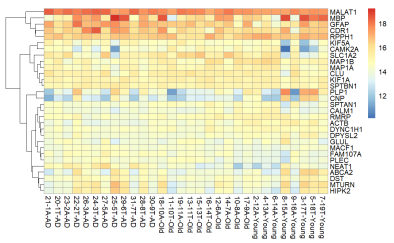
```
out of 23525 with nonzero total read count
adjusted p-value < 0.05
LFC > 0 (up)       : 2230, 9.5%
LFC < 0 (down)     : 1783, 7.6%
outliers [1]       : 0, 0%
low counts [2]     : 1829, 7.8%
(mean count < 1)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

(c) AD vs Young

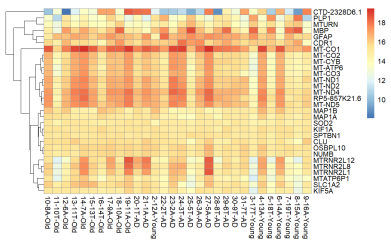
Figure 1: Results from DESeq with the STAR counts

In the following Figure 2 the gene expression heatmaps for the Top 30 genes can be seen. Only some genes like KIF1A, KIF5A and CDR1 were in the Top 30 of both sets. The kallisto heatmap shows in general more genes with a high expression. The high expressed genes are also detected for two of the people from the young group, otherwise they could have been related to age.

Before the PCA was done, a Variance Stabilizing Transformation was applied to the data. In the PCA shown in Figure 3 the groups were overlapping in both cases. The PCA from the kallisto counts showed a smaller variance. Additionally, it looks

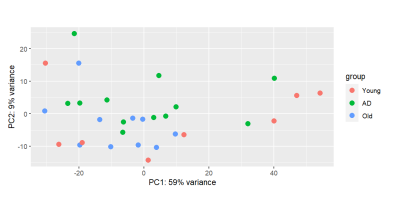


(a) Heatmap with counts from STAR files

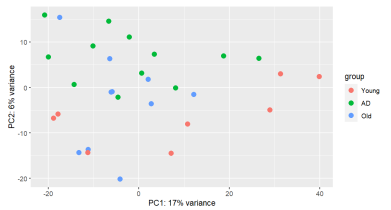


(b) Heatmap with counts from kallisto files

Figure 2: Comparison of the gene expression heatmaps of the Top 30 genes



(a) PCA with counts from STAR files



(b) PCA with counts from kallisto files

Figure 3: Comparison of the PCA plots

more like young and old as well as old and diseased are overlapping, which were also more similar than young and diseased in the DESeq() summary.

Additional to the two visualizations shown here, MA-Plots, heatmaps for sample distance and some more were generated.

3 ChIP-seq Data Analysis

The read.table function was used to conduct the analysis with bed files as the data source. It begins by reading data from an input file and then transforming that data into a dataframe. By default, lines that begin with are disregarded, and the presence of the option header value = TRUE indicates that the first line contains the column's name. Since data exploration needs to be done, we have come to the conclusion that we should perform analysis on both a healthy and a sick patient. This will allow us to have a comprehensive understanding of the data within both patients, as well as a clear picture of the differences between them. Installing the rtracklayer package will allow you to compare the healthy sample with the sick sample. This package will directly import the peaks as GRanges objects. After including the actual values as text labels on the barplot that we created, we were able to determine the total number of peaks.

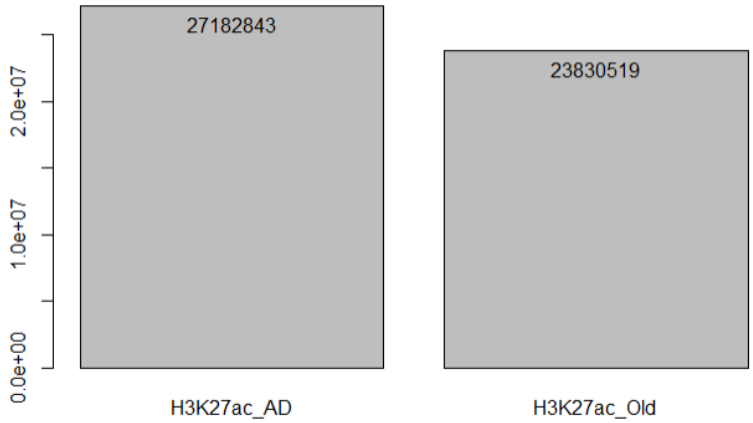


Figure 4: Number of peaks in healthy and sick sample.

After that, one more goal of ours was to use the findOverlaps function contained within the GenomicRanges package to compute the overlaps that exist between two different sets of regions. It was possible to obtain the subset of H3K27acAD peaks that overlap with the H3K27acOld peaks by making use of the function known as subsetByOverlaps.

```
```{r}
For computing overlaps
ovlHits <- findOverlaps(H3K27ac_AD, H3K27ac_Old)
Showing the result in Hits object
ovlHits
```

The results of the overlap between healthy and sick patients are presented here.

```
Hits object with 27448420 hits and 0 metadata columns:
             queryHits subjectHits
             <integer>   <integer>
         [1]         3           2
         [2]        20          22
         [3]        21          22
         [4]        22          22
         [5]        23          23
         ...       ...         ...
  [27448416]  27182832    23830516
  [27448417]  27182832    23830517
  [27448418]  27182833    23830517
  [27448419]  27182834    23830517
  [27448420]  27182834    23830518
  -------
  queryLength: 27182843 / subjectLength: 23830519
```

Figure 5: Overlap healthy and sick sample.

The overlap between the healthy and sick samples has been presented in the form of a Venn diagram in order to make it easier to understand. The goal of this presentation was to make it clearer which samples were healthy and which were sick. Taking the subset of peaks that are unique to H3K27ac AD and H3K27ac Old was the first thing that we did in this process. In order to accomplish this, we constructed a Venn object with the help of the Vennerable library. This object contained the number of peaks that were discovered in each of the subgroups. This object can be passed in as a parameter to the plot function if it's given to it. code can be seen in our GitHub repos.
```

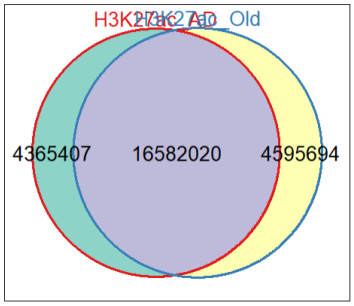


Figure 6: Venn-diagram showing peak overlap of H3K27acAD and H3K27acOld.

3.1 **Genomic Functional Annotation**.

An understanding of how transcription factors work requires an understanding of the genes and genomic characteristics they interact with. TxDb objects are intended to be used in this case. Prefabricated databases can be accessed via this object, which is part of an annotation package. In the TxDb.Hsapiens.UCSC.hg19.knownGene package, all known human genes transcripts are included, based on the UCSC coordinates of the hg19 genome assembly.

We have got the subset of genes that have a peak in their promotor regions as an output. We found altogether, 558398 Alzheimer Diesease peaks that overlap a promotor region. We have first restricted the genes to only that start coordinate (TSS). Then we used the function distanceetoNearest () from the GenomicRanges packages to calculate the distance from peaks to TSS. WE also used Hits object which was resulted from distancetoNearest() function and used the distances to subset the peaks to those that are less than 10Kb away from the TSS.

4 Machine Learning Model-Initiation

Before man use machine learning algorithms, data must be prepared for analysis by removing unwanted attributes and missing values. We preprocessed the RNA data and ChIP-seq were already preprocessed.

To apply a machine-learning model, we should split the data into young and old, healty and Alzheimer to visualize the difference between gene expression by young and old patient and also healthy and patients with Alzheimer's.

In our case, we need to use semi-supervised learning because we want to find something about our data, and clustering method that can we apply to partially labeled data.

As part of our strategy to develop some models based on the dataset that we already have. Because it only uses a small amount of labelled data and a large amount of unlabelled data, the semi-supervised machine learning algorithm is an appropriate choice for this situation. This algorithm provides the advantages of both unsupervised and supervised learning while avoiding the difficulties associated with locating a significant amount of labelled data. We choose to do the machine learning in python and the script will be available in GitHub.

At the end we will have a machine learning model which shows the difference between our outcome of interest and we can use it to make predictions.

Member	Last week's work
Christina	RNA-seq analysis for the three groups
Pushpa	Chip-seq analysis between healthy and sick sample
Melika	Machine learning

5 Contributions

6 Plans for the upcoming week

- Part 2 of ChIP-seq analysis
- Clean up and comment R notebook
- Machine Learning