Heath TDP-43 Fibroblast Report

Claire Green

10/10/2017

Aim

The aim of this experiment was to analyse RNA-seq data provided by the Heath lab that was taken from patients with TDP-43 mutations. The dataset consists of RNA-seq data taken from fibroblasts of 7 patients and 3 controls. Each sample contained data from a nuclear fraction, a cytoplasmic fraction, and whole cell transcriptome.

Condition	ID	Mutation Type	Mutation	Gender
Patient	192	Truncation	N/A	M
Patient	193	Truncation	N/A	M
Patient	194	Truncation	N/A	M
Patient	51	Missense	M337V	M
Patient	55	Missense	G2875	M
Patient	48	Missense	A321V	F
Control	155	N/A	N/A	M
Control	2303	N/A	N/A	M
Control	170	N/A	N/A	M
Control	159	N/A	N/A	F

The aim of the experiment was to conduct differential expression analysis, followed by functional enrichment, on the various permutations that exist within the dataset. This includes:

- Patient vs Control
  - Cytoplasm
  - Nucleus
  - Whole Cell Transcriptome
- Patients
  - Cytoplasm vs Nucleus
  - Cytoplasm vs WCT
  - Nucleus vs WCT
- Controls
  - Cytoplasm vs Nucleus
  - Cytoplasm vs WCT
  - Nucleus vs WCT

Patient vs control conditions are to find any disease specific genes/processes that are dysfunctional in the patient fibroblasts. The cytoplasm/nucleus/WCT comparisons are too see if there are location-specific dysfunctional genes. For example, if a gene is downregulated in the cytoplasm as compared to the nucleus in patients, but not in controls, this could suggest issues with nuclear export.

Pre-processing

Pre-processing was conducted using the RNA-seq pipeline from the python package bcbio. This was run on Iceberg using the following run script:

```
#!/bin/bash
#$ -pe openmp 8
#memory requests are per-core
#$ -l rmem=8G -l mem=8G
#Prefer the hidelab queue but spill over to over queues if it is full
#$ -P hidelab
#$ -j y

module load apps/gcc/5.2/bcbio/0.9.6a
work_dir='/shared/hidelab2/user/mdp15cmg/TDP-43/PH_Fibroblasts'

#Seq.Reads file directories
r1_files=$work_dir/input/Read1
r2_files=$work_dir/input/Read2

#Read in seq reads
r1=($(find $r1_files -type f -name "*.gz"|sort -n))
r2=($(find $r2_files -type f -name "*.gz"|sort -n))

#Download the best-practice template file for RNAseq experiment
echo "DOWNLOADING TEMPLATE"
bcbio_nextgen.py -w template illumina-rnaseq PH_bcbio

#Edit the template
echo "EDITTING TEMPLATE"
#Switch to using star
sed -i 's/tophat2/star/g' $work_dir/PH_bcbio/config/PH_bcbio-template.yaml

#Initialise the main analysis
echo "INITIALISING ANALYSIS"
bcbio_nextgen.py -w template $work_dir/PH_bcbio/config/PH_bcbio-template.yaml $work_dir/PH_bcbio.csv ${

#Perform the analysis
echo "PERFOMING ANALYSIS"
cd $work_dir/PH_bcbio/work
bcbio_nextgen.py -n 8 ../config/PH_bcbio.yaml
```

Bcbio includes steps for quality control, adapter trimming, alignment, variant calling, transcriptome reconstruction, and post-alignment quantitation of genes and gene isoforms.

Quality Control

Once the count matrix had been generated, the next step was to take a look at the data. To start with I looked at the FastQC results generated by bcbio. This includes an html report for each sample, including information on sequence quality and GC content. Review of all samples suggested there was no obvious issue with technical quality.

Next I analysed the spread of the data using Principle Component Analysis. I used both 2D PCA and 3D.

2D PCA

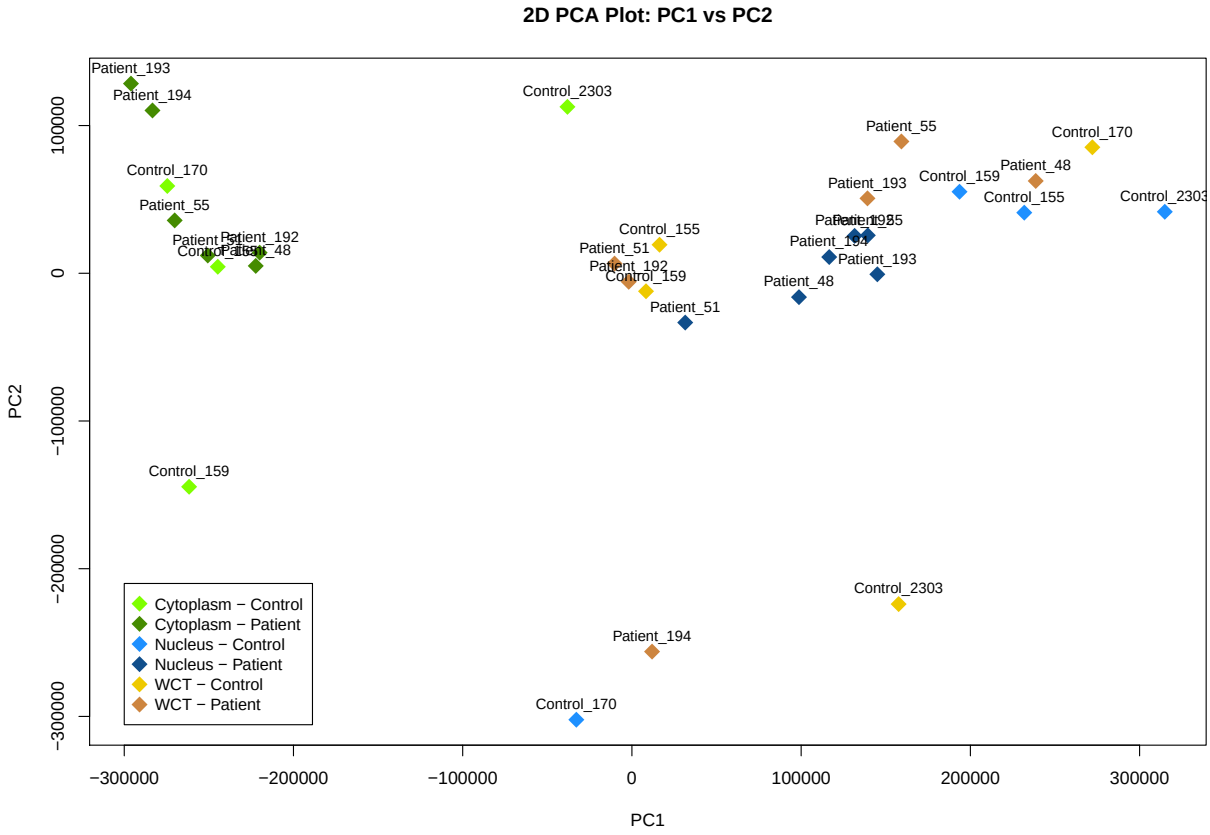


Figure 1:

As you can see from the PCA plot, the clustering of the samples seems to be based largely on where the RNA was sampled from, and in some cases clustering into patients and controls. There is a clear overlap between nuclear and whole cell transcriptomic samples, suggesting nuclear expression has a stronger influence in overall RNA expression.

There are four samples that seem to separate from the rest - Control 159 (Cytoplasm), Control 170 (Nucleus), Patient 194 (WCT), and Control 2303 (WCT). Because there is a mix of backgrounds here, it's unlikely that the separation is due to a particular patient or a particular condition. These will be further analysed later

3D PCA

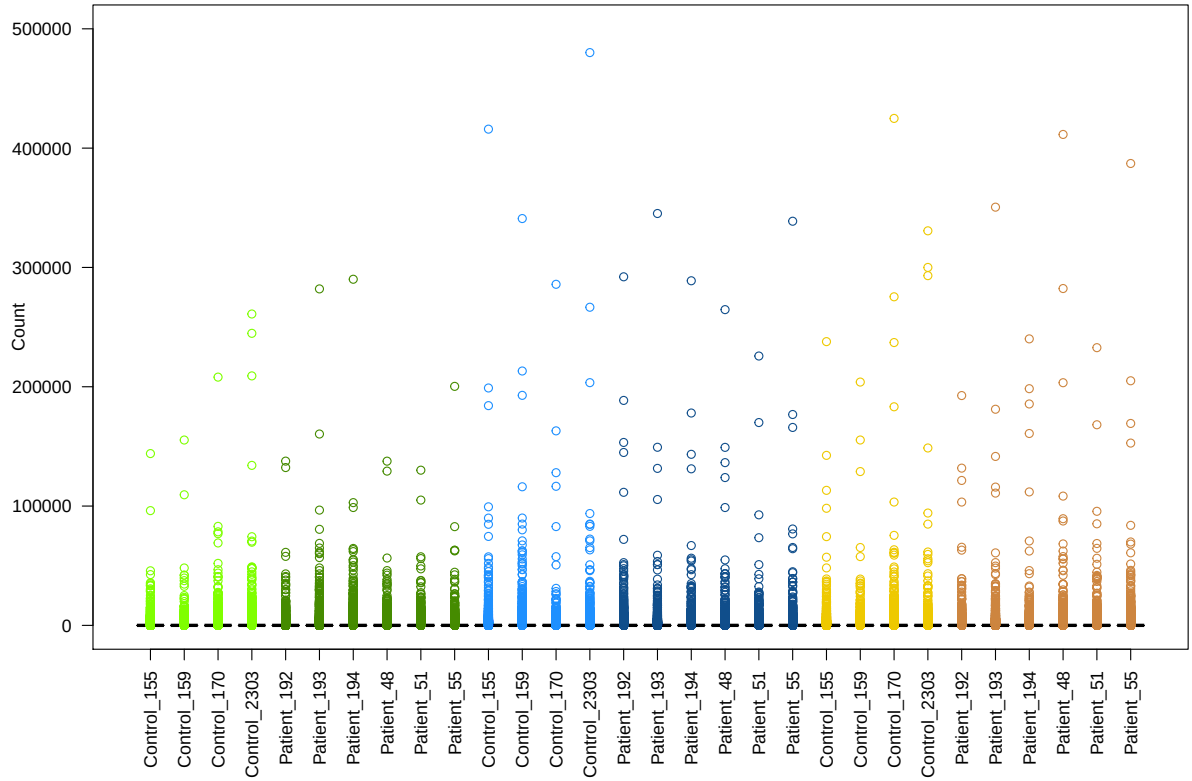
3D PCA is much harder to report, due to the medium, but it is a really useful way of visualising the spread of data in 3D space. I have recorded the output and it can be found in the video below

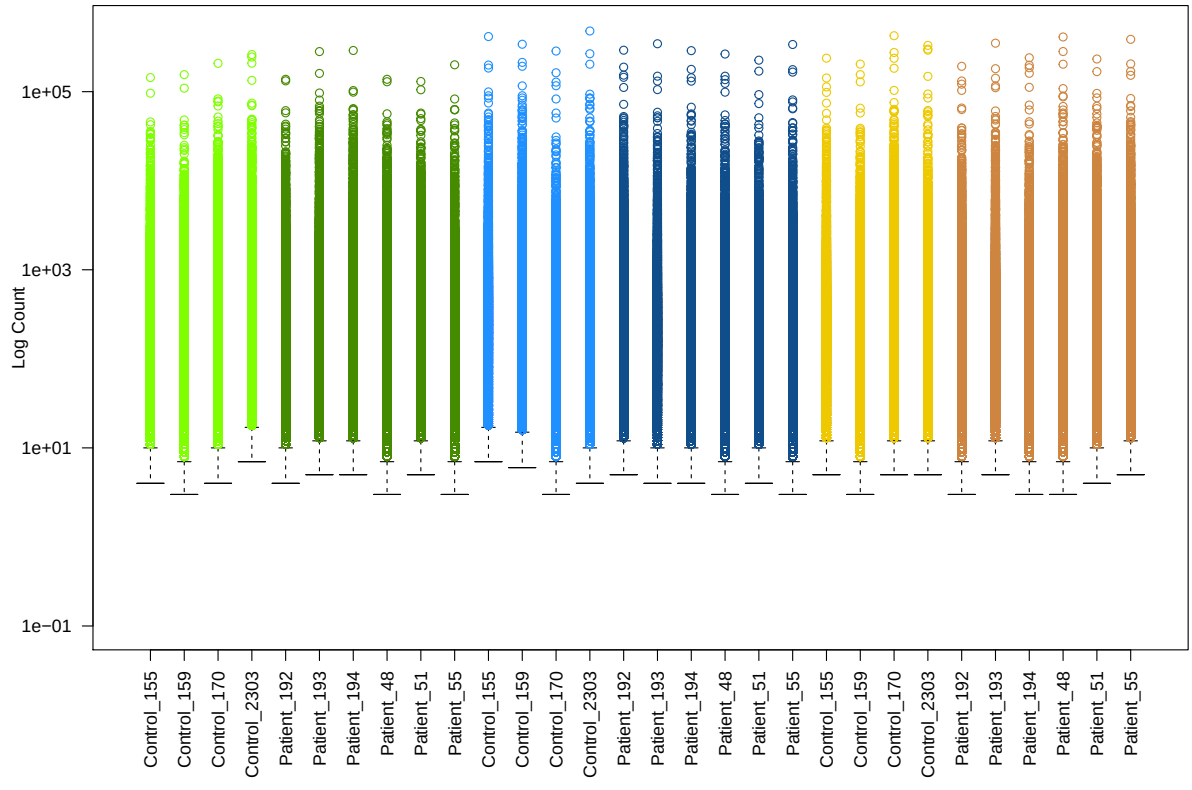
Link to Video

Again, you can see most samples lie flat across the PC1 and PC3 plane, however these four samples are distributed well into the PC2 plane.

Box Plots

Box plots allow us to look at the spread of the data. This helps to identify outliers that may skew any later results.


Count
500000
400000
300000
200000
100000
0
Control_155
Control_159
Control_170
Control_2303
Patient_192
Patient_193
Patient_194
Patient_48
Patient_51
Patient_55
Control_155
Control_159
Control_170
Control_2303
Patient_192
Patient_193
Patient_194
Patient_48
Patient_51
Patient_55
Control_155
Control_159
Control_170
Control_2303
Patient_192
Patient_193
Patient_194
Patient_48
Patient_51
Patient_55


Log Count
1e+05
1e+03
1e+01
1e−01
Control_155
Control_159
Control_170
Control_2303
Patient_192
Patient_193
Patient_194
Patient_48
Patient_51
Patient_55
Control_155
Control_159
Control_170
Control_2303
Patient_192
Patient_193
Patient_194
Patient_48
Patient_51
Patient_55