

Analyse RNA Seq SLA - Visualisation des comptages normaliser TPM, Condition contrôle et SLA

12 juin 2025

1 Introduction

Ce rapport présente une analyse détaillée de l'expression des gènes dans le cadre de la Sclérose Latérale Amyotrophique (SLA) en utilisant les données RNA-seq. L'objectif est de visualiser l'expression normalisée des 56 gènes d'intérêt à travers différents échantillons et conditions. Nous détaillons chaque étape du processus, des choix méthodologiques aux résultats finaux.

2 Méthodologie

Pour commencer, j'utilise les librairies `readr` et `tidyverse` (qui inclut `ggplot2`, `dplyr`, et `tidyr`) pour la manipulation des données et la visualisation/transformation.

```
library(readr)      # Pour la lecture de fichiers TSV
library(tidyverse)  # Pour la manipulation de données et la visualisation
```

2.1 Etape 1 : lecture des données

Il s'agit ici de transformer en format long la matrice de comptage des 56 gènes chargés pour faciliter la manipulation et la visualisation :

```
# Chargement de la matrice de comptage des 56 gènes
counts <- read_tsv("~/Final_counts_56genes.tsv")

# Transformation en format long
counts_long <- counts %>%
  pivot_longer(
    cols = -c(Geneid, gene_name, Chr, Length),
    names_to = "full_id",
    values_to = "Counts"
  ) %>% separate(full_id, into = c("Run", "Sample", "Condition"), sep = "-", remove = TRUE)
```

2.2 Etape 2 : Calcul des TPM

Le TPM (Transcripts Per Million) normalise les données par la longueur des gènes et la profondeur de lecture.

```
counts_tpm <- counts_long %>%
  group_by(Sample) %>% mutate(rate = Counts / Length, tpm = rate / sum(rate) * 1e6) %>% ungroup()
```

Pour une visualisation lisible, on divise les 56 gènes en groupes ici par 8 pour l'exemple :

2.3 Etape 3: Résultat et visualisation

Chaque groupe est visualisé par un boxplot montrant l'expression TPM par gène selon le Run et la Condition (SLA ou controle)

```

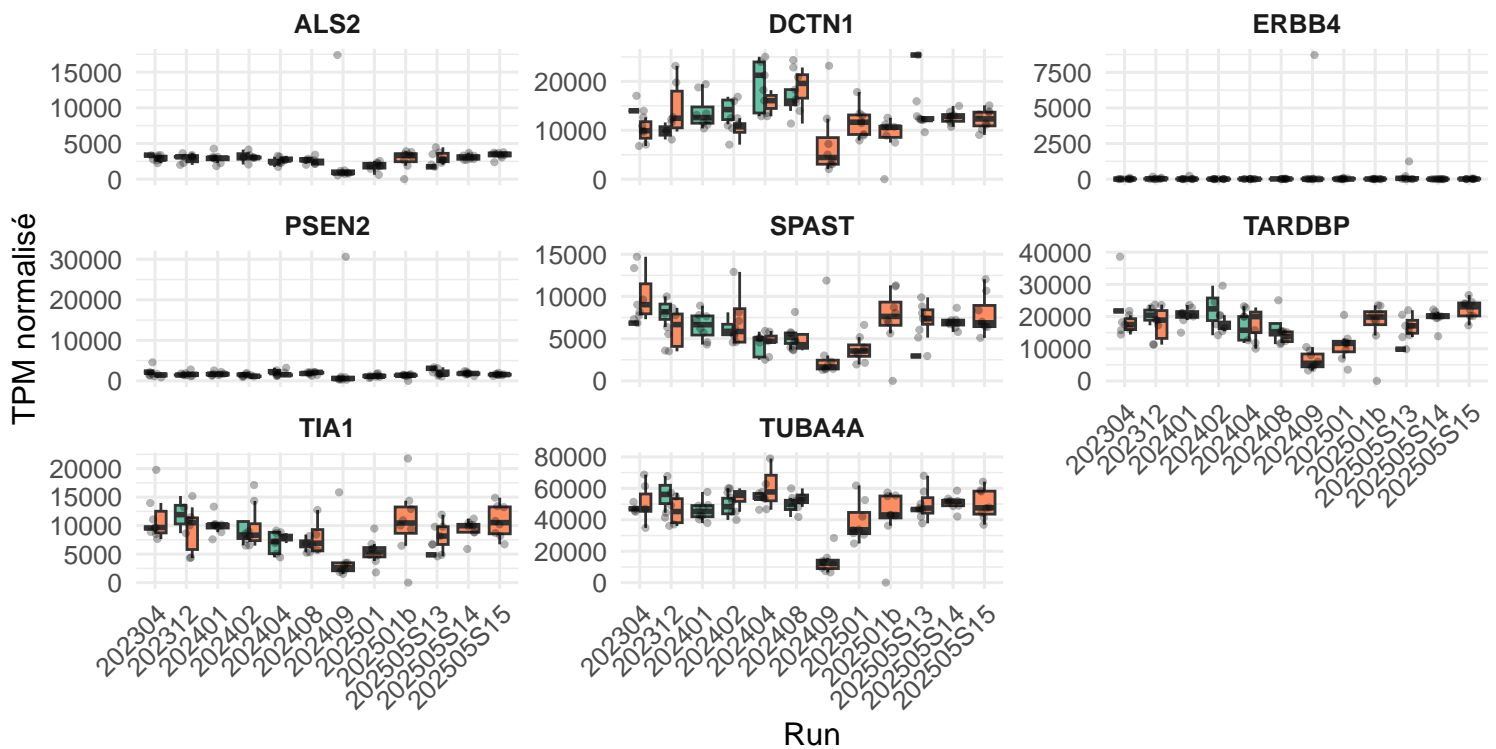
for(i in seq_along(gene_groups)) {
  gsub <- gene_groups[[i]]
  df_sub <- counts_tpm %>% filter(gene_name %in% gsub)

  p <- ggplot(df_sub, aes(x = Run, y = tpm, fill = Condition)) +
    geom_boxplot(outlier.shape = NA) +
    geom_jitter(width = 0.2, alpha = 0.3, size = 0.8) +
    facet_wrap(~ gene_name, scales = "free_y") +
    scale_fill_brewer(palette = "Set2") +
    theme_minimal(base_size = 12) +
    labs(
      title = paste("Expression TPM des gènes - Groupe", i),
      x = "Run",
      y = "TPM normalisé"
    ) +
    theme(
      axis.text.x = element_text(angle = 45, hjust = 1),
      strip.text = element_text(face = "bold"),
      legend.position = "top"
    )
  print(p)
}

```

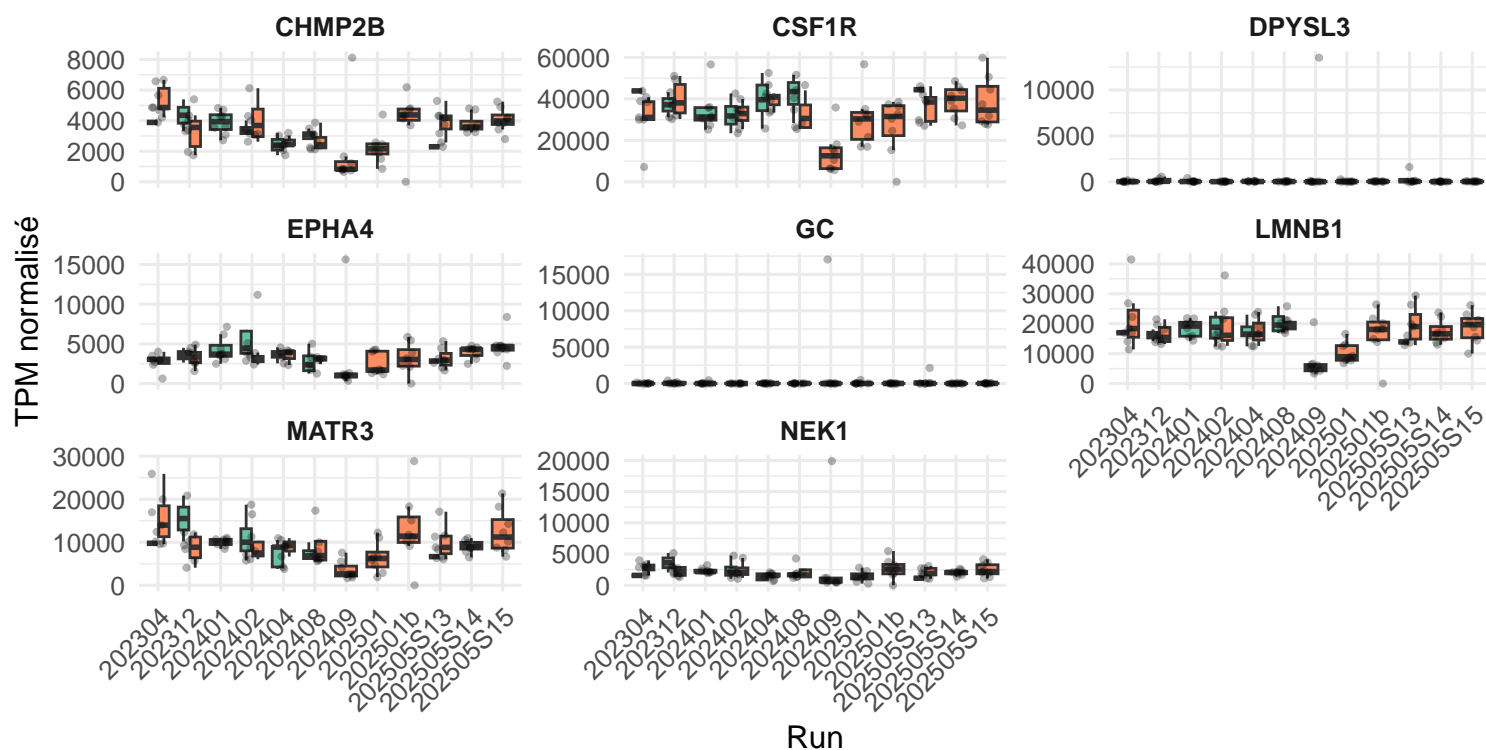
Expression TPM des gènes – Groupe 1

Condition  Control  SLA





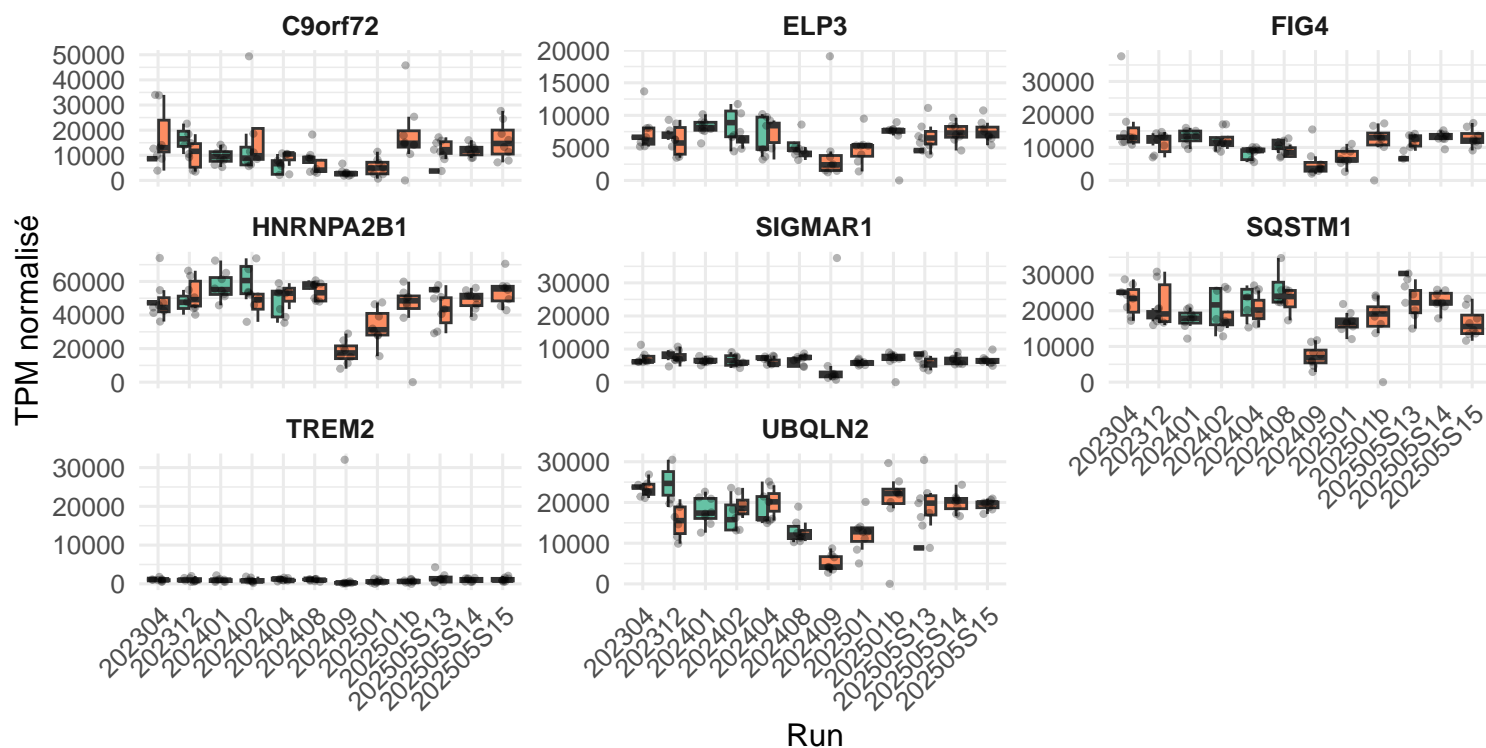
Expression TPM des gènes – Groupe 2

Condition  Control  SLA



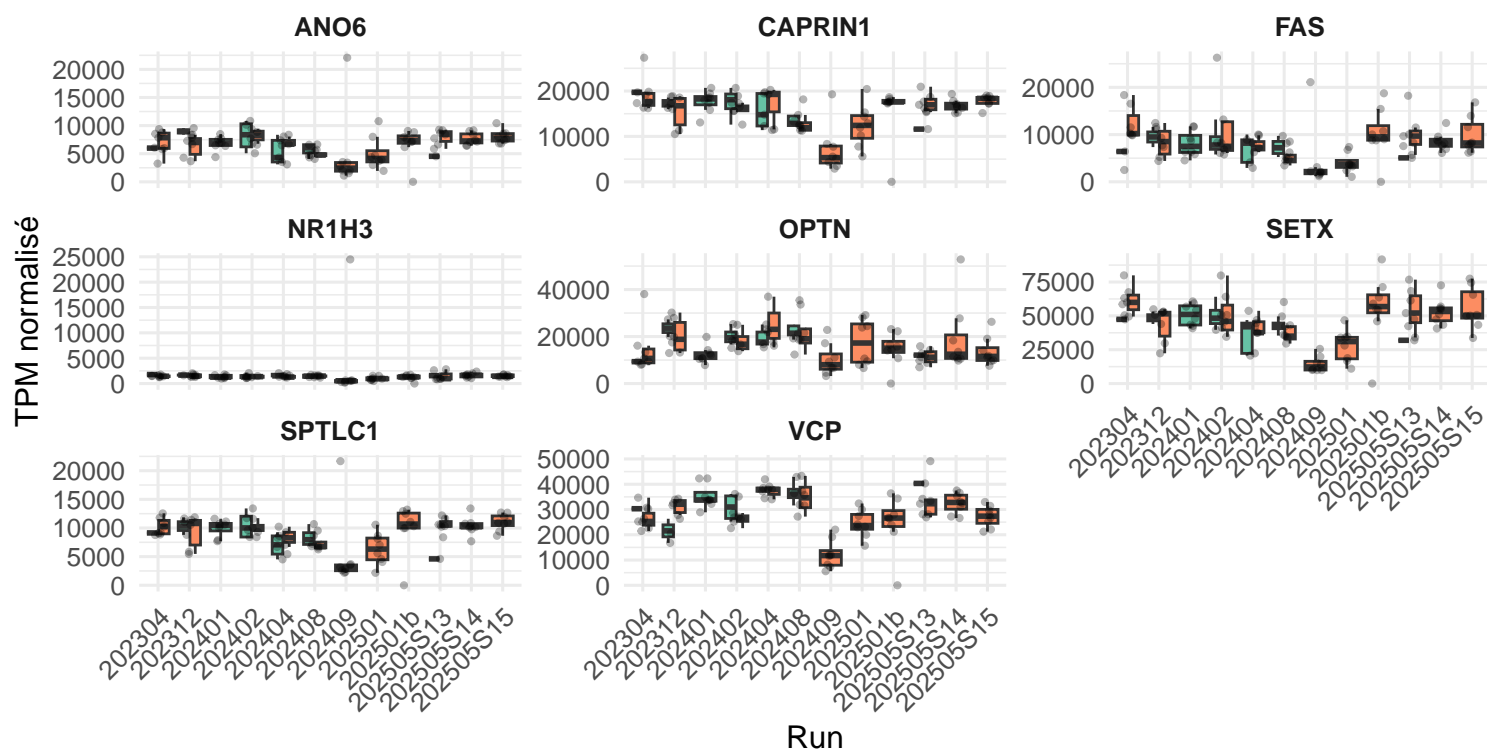
Expression TPM des gènes – Groupe 3

Condition  Control  SLA





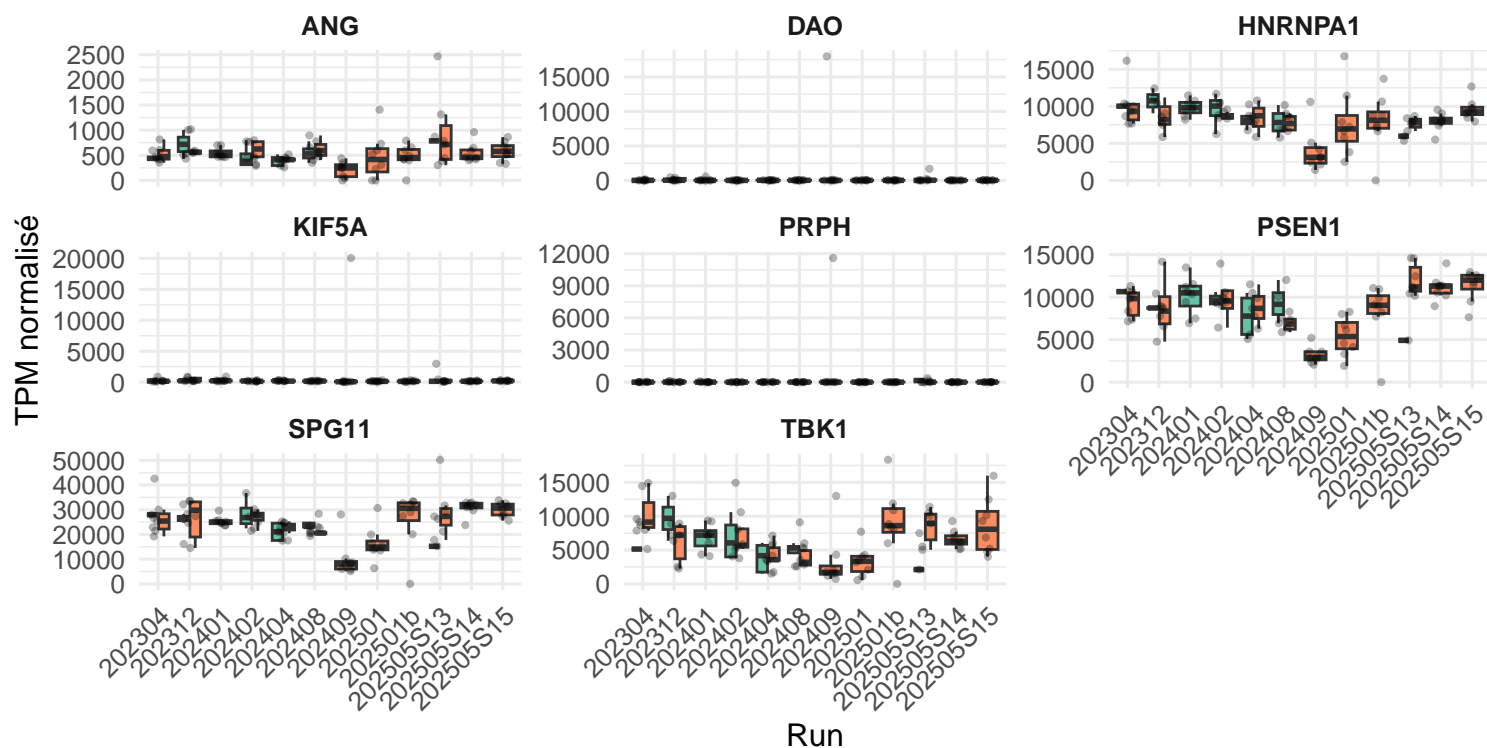
Expression TPM des gènes – Groupe 4

Condition  Control  SLA



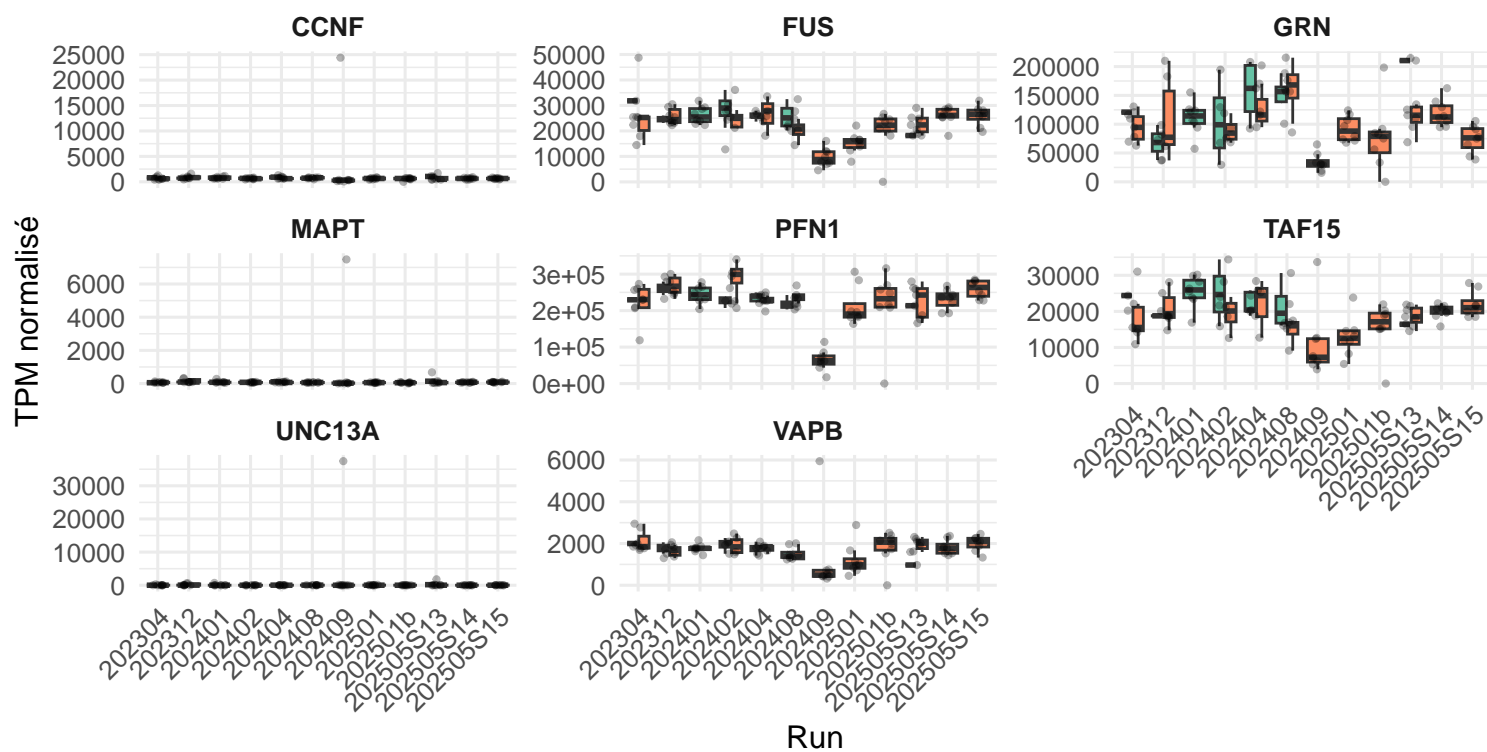
Expression TPM des gènes – Groupe 5

Condition  Control  SLA





Expression TPM des gènes – Groupe 6

Condition  Control  SLA



Expression TPM des gènes – Groupe 7

Condition  Control  SLA

