Analyse comparative des méthodes de normalisation RNA Seq cible SLA

Mickael Coquerelle

2025-07-08

Contents

1 Avant-propos et objectifs

Ce rapport sert de synthèse a mes différentes investigations et scripts/essais que j'ai pu effectuer en `R`, pour générer/évaluer mes données de comptages et surtout à mes investigations sur les méthodes de normalisation pour faire de l'analyse quantitative dans le périmètre du RNA-Seq ciblé sur le panel de 56 gènes et tout ce que cela implique en terme de contraintes et de limites, comme détaillé dans mes anciennes réalisations.

Ici je traite de **TPM** (intègre la longueur des genes), **TMM** (concu au départ pour le transcriptome complet) et **RLE** (hypothèse d'une médiane stable). Je tiens à rappeler également que les approches classiques dans notre contexte de faible nombre de gènes sont relativements fragiles (cf. mon travail bibliographique sur les méthodes de normalisation).

2 Préparation/mise en forme du dataset

```
#  Charger les librairies nécessaires.
#  edgeR : pour calcNormFactors (TMM, RLE) + cpm()
#  purrr : fusion propre des dataframes
library(readr)      # Import TSV
library(dplyr)      # Manipulations de données
```

```r
library(tidyr)      # pivot_longer / separate
library(ggplot2)    # Visualisations
library(edgeR)      # Normalisation TMM & RLE
library(purrr)      # reduce pour fusion
library(broom)
library(kableExtra)
library(patchwork)  # pour juxtaposer graphiques
library(ggrepel)
library(tibble)     # rownames_to_column
```

3 Import/préparation de la matrice de comptages

```r
# Lecture du fichier de comptages généré par featureCounts (-f).
# Colonnes attendues : Geneid | gene_name | Chr | Length | échantillons...

tableau_comptages_bruts <- read_tsv("~/Final_counts_56genes.tsv", show_col_types = FALSE)

# Renommage pour uniformiser la colonne gene_name en nom_gene
tableau_comptages_bruts <- tableau_comptages_bruts %>% rename(nom_gene = gene_name, Longueur = Length)

#filtre_gene <- c("DPYSL3")
filtre_gene <- c("")
#filtre_pat <- c("202505S14-C01P092-SLA","202501b-C01P269-SLA","202312-2106212448-SLA")
filtre_pat  <- c("")
# Passage au format long pour faciliter les transformations ultérieures.

comptages_long <- tableau_comptages_bruts %>%
  pivot_longer(
    cols      = -c(Geneid, nom_gene, Chr, Longueur),
    names_to  = "ID_complet",
    values_to = "lectures_brutes"
  ) %>%
  separate(
    col     = "ID_complet",
    into    = c("run", "echantillon", "condition"),
    sep     = "-",
    extra   = "merge",
    remove  = TRUE
  )
comptages_long <- comptages_long %>% filter(!echantillon %in% filtre_pat) %>%  filter(!nom_gene %in% f

# Appliquer un seuil de comptage brut : garder les gènes exprimés à au moins 1000 dans au moins un éch
genes_conserves <- comptages_long %>%
  group_by(Geneid) %>%
  summarise(max_comptage = max(lectures_brutes, na.rm = TRUE)) %>%
  filter(max_comptage >= 100) %>%
  pull(Geneid)

# Filtrer le tableau long pour ne garder que ces gènes
comptages_long <- comptages_long %>%
  filter(Geneid %in% genes_conserves)
```

4 Récuperer l'expression théorique dans le sang pr les gènes étudiés

Avec ce chunk, je cherche a récupérer l'expression théorique sanguine sur gtex, pour voir si j'ai une cohérence globale entre les comptes empiriques (normalisés TPM) et l'expression documenté sur ensembl, mon idée générale c'est de voir si retrouve une cohérence entre mes valeurs expérimentales et les valeurs théoriques

```
genes_sla <- c("ALS2", "ANG", "ANO6", "APOE", "APP", "C21orf2","C9orf72","CAPRIN1", "CCNF","CHCHD10","
```

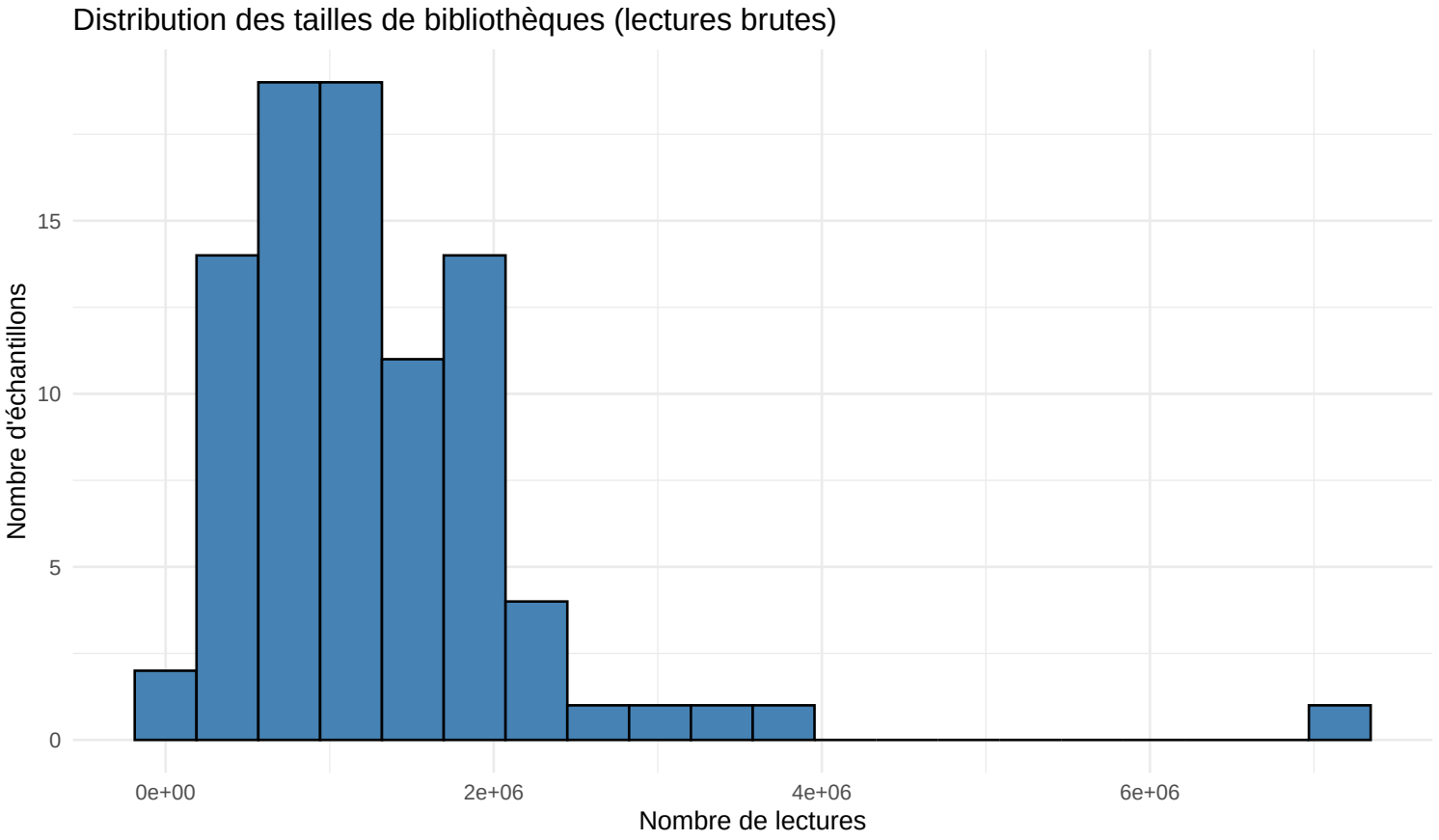
5 Analyse exploratoires des données brutes

```
# Somme des lectures par échantillon, vérification de la taille des lib
taille_bibliotheque <- comptages_long |> group_by(echantillon) |> summarise(somme_lectures =sum(lectur
print(taille_bibliotheque)
```

```
## # A tibble: 88 x 2
##    echantillon somme_lectures
##    <chr>                <dbl>
##  1 C01P092            7181393
##  2 C01P269            3646210
##  3 C01P276            3296245
##  4 C01P114            2849830
##  5 C01P035            2666206
##  6 C01P003            2377025
##  7 C01P051            2287024
##  8 C01P082            2192536
##  9 2102111799         2147570
## 10 C01P038            2054406
## # i 78 more rows
```

6 Histogramme des tailles de bibliothèques

```
ggplot(taille_bibliotheque, aes(x = somme_lectures)) +
  geom_histogram(bins = 20, fill = "steelblue", color = "black") +
  labs(title = "Distribution des tailles de bibliothèques (lectures brutes)",
       x = "Nombre de lectures",
       y = "Nombre d'échantillons") +
  theme_minimal()
```



Normalisations

```
  # On créer une matrice ge pour edgeR
comptages_larges <- comptages_long |> group_by(nom_gene, echantillon) |> summarise(total_lectures = su
```

6.1 TPM

```
tpm_long <- comptages_long |> group_by(echantillon) |> mutate(
  taux_expression = lectures_brutes / Longueur,
  TPM_transcrits_par_M = taux_expression / sum(taux_expression) * 1e6) |> ungroup ()
```

```
# La somme des TPM est bien de 1M pour chaque patient:
verif_tpm <- tpm_long |> group_by(echantillon) |>
  summarise(somme_tpm = sum(TPM_transcrits_par_M)) |> arrange(somme_tpm)
#Sortie
verif_tpm |> knitr::kable(digits = 1, col.names = c("Échantillon", "Somme des TPM")) |>
  kableExtra::kable_styling(full_width = FALSE, position = "center")
```

Échantillon	Somme des TPM
2306221774	1e+06
2307121892	1e+06
C01P016	1e+06
C01P027	1e+06
C01P034	1e+06
C01P039	1e+06
C01P040	1e+06
C01P070	1e+06
C01P092	1e+06
C01P094	1e+06

C01P195	1e+06
C01P216	1e+06
C01P253	1e+06
1409241167	1e+06
1810111440	1e+06
1912031641	1e+06
2006161710	1e+06
2102111799	1e+06
2103051881	1e+06
2105181882	1e+06
2105201539	1e+06
2106212448	1e+06
2108021909	1e+06
2306221661	1e+06
2306221684	1e+06
2306221703	1e+06
2309191962	1e+06
2309211582	1e+06
2401121737	1e+06
2406121482	1e+06
2411071506	1e+06
C01P003	1e+06
C01P006	1e+06
C01P012	1e+06
C01P014	1e+06
C01P017	1e+06
C01P022	1e+06
C01P023	1e+06
C01P024	1e+06
C01P026	1e+06
C01P030	1e+06
C01P035	1e+06
C01P038	1e+06
C01P041	1e+06
C01P047	1e+06
C01P051	1e+06
C01P053	1e+06
C01P056	1e+06
C01P057	1e+06
C01P059	1e+06
C01P060	1e+06
C01P068	1e+06
C01P069	1e+06
C01P073	1e+06
C01P075	1e+06
C01P082	1e+06
C01P083	1e+06
C01P086	1e+06
C01P089	1e+06
C01P090	1e+06

C01P095	1e+06
C01P114	1e+06
C01P116	1e+06
C01P120	1e+06
C01P128	1e+06
C01P135	1e+06
C01P138	1e+06
C01P145	1e+06
C01P148	1e+06
C01P154	1e+06
C01P159	1e+06
C01P163	1e+06
C01P167	1e+06
C01P179	1e+06
C01P181	1e+06
C01P196	1e+06
C01P222	1e+06
C01P255	1e+06
C01P260	1e+06
C01P263	1e+06
C01P269	1e+06
C01P276	1e+06
C01P278	1e+06
C01P013	1e+06
C01P061	1e+06
C01P088	1e+06
C01P188	1e+06
C01P238	1e+06

6.2 TMM (Trimmed Mean of M-values)

```
objet_tmm_edge <- DGEList(counts = as.matrix(comptages_larges))
objet_tmm_edge <- calcNormFactors(objet_tmm_edge, method ="TMM")

matrice_cpm_tmm <- cpm(objet_tmm_edge, normalized.lib.sizes = TRUE)
```

6.3 RLE (Relative Log expression)

```
objet_rle_edge <- DGEList(counts = as.matrix(comptages_larges))
objet_rle_edge <- calcNormFactors(objet_rle_edge, method ="RLE")

matrice_cpm_rle <- cpm(objet_rle_edge, normalized.lib.sizes = TRUE)
```

6.4 Création d'un tibble global des comptes normalisés

```
tpm_tb <- tpm_long |> select(echantillon, nom_gene, TPM_transcrits_par_M)

tmm_tb <- matrice_cpm_tmm |> as.data.frame() |> rownames_to_column("nom_gene") |> pivot_longer(-nom_ge

rle_tb <- matrice_cpm_rle |> as.data.frame() |> rownames_to_column("nom_gene") |> pivot_longer(-nom_ge
```

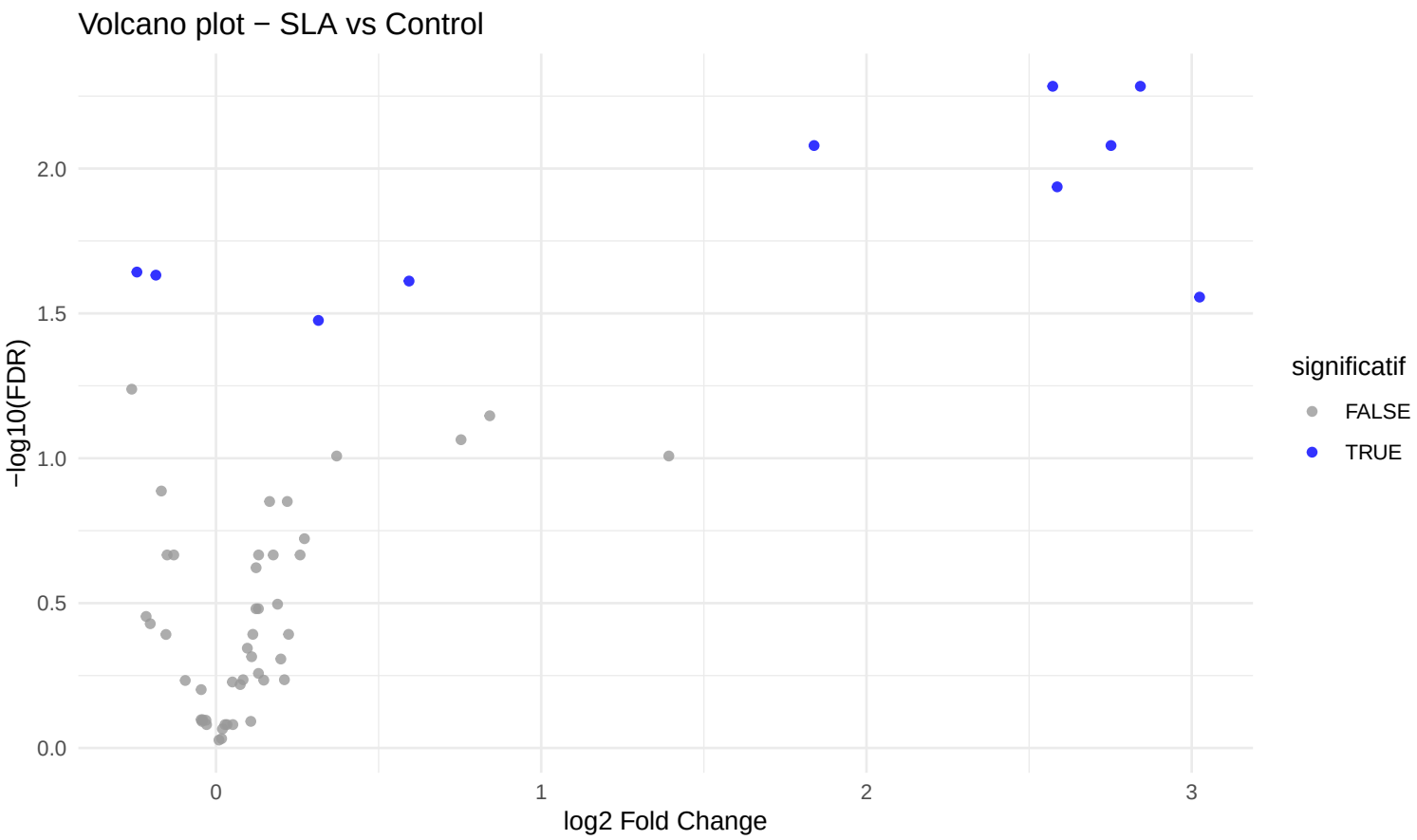
```
names_to = "echantillon", values_to = "RLE_cpm")

bruts_tb <- comptages_long %>%
  group_by(echantillon, nom_gene) %>%
  summarise(total_lectures = sum(lectures_brutes), .groups="drop")

donnees_normalisees <- reduce(list(tpm_tb, tmm_tb, rle_tb,bruts_tb), full_join, by = c("echantillon","
  left_join (comptages_long |> distinct(echantillon,run,condition), by ="echantillon")
```

```
# Création d'une table de design
condition <- donnees_normalisees %>%
  distinct(echantillon, condition) %>%
  arrange(match(echantillon, colnames(comptages_larges))) %>% pull(condition)

# Vérification des niveaux
condition <- factor(condition, levels = c("Control", "SLA"))

# Création d'un objet DGEList + normalisation
dge <- DGEList(counts = as.matrix(comptages_larges))
dge <- calcNormFactors(dge, method = "TMM")

# Design matrix
design <- model.matrix(~ condition)

# Estimation dispersion + modélisation
dge <- estimateDisp(dge, design)
fit <- glmFit(dge, design)

# Test du coefficient lié à la condition SLA
lrt <- glmLRT(fit, coef = 2)

# Extraction des résultats
resultats_diff_expr <- topTags(lrt, n = Inf)$table

# Ajout du statut significatif
resultats_diff_expr$significatif <- resultats_diff_expr$FDR < 0.05

# Affichage
resultats_diff_expr |> filter(significatif) |> arrange(FDR) |> head()
```

```
##             logFC    logCPM       LR       PValue         FDR significatif
## UNC13A  2.8421031  9.317547 14.89134 0.0001138847 0.005202006         TRUE
## ERBB4   2.5726048  7.608751 13.96965 0.0001857859 0.005202006         TRUE
## DPYSL3  1.8388445  7.582423 11.85283 0.0005757388 0.008331061         TRUE
## GC      2.7517465  5.764210 11.79132 0.0005950758 0.008331061         TRUE
## DAO     2.5862989  6.610109 10.76757 0.0010329420 0.011568950         TRUE
## TAF15  -0.2431073 14.538618  9.18536 0.0024395856 0.022769466         TRUE
```

```
ggplot(resultats_diff_expr, aes(x = logFC, y = -log10(FDR), color = significatif)) +
  geom_point(alpha = 0.8) +
  scale_color_manual(values = c("gray60", "blue")) +
  labs(title = "Volcano plot - SLA vs Control",
       x = "log2 Fold Change",
       y = "-log10(FDR)") +
```

```
  theme_minimal()
```

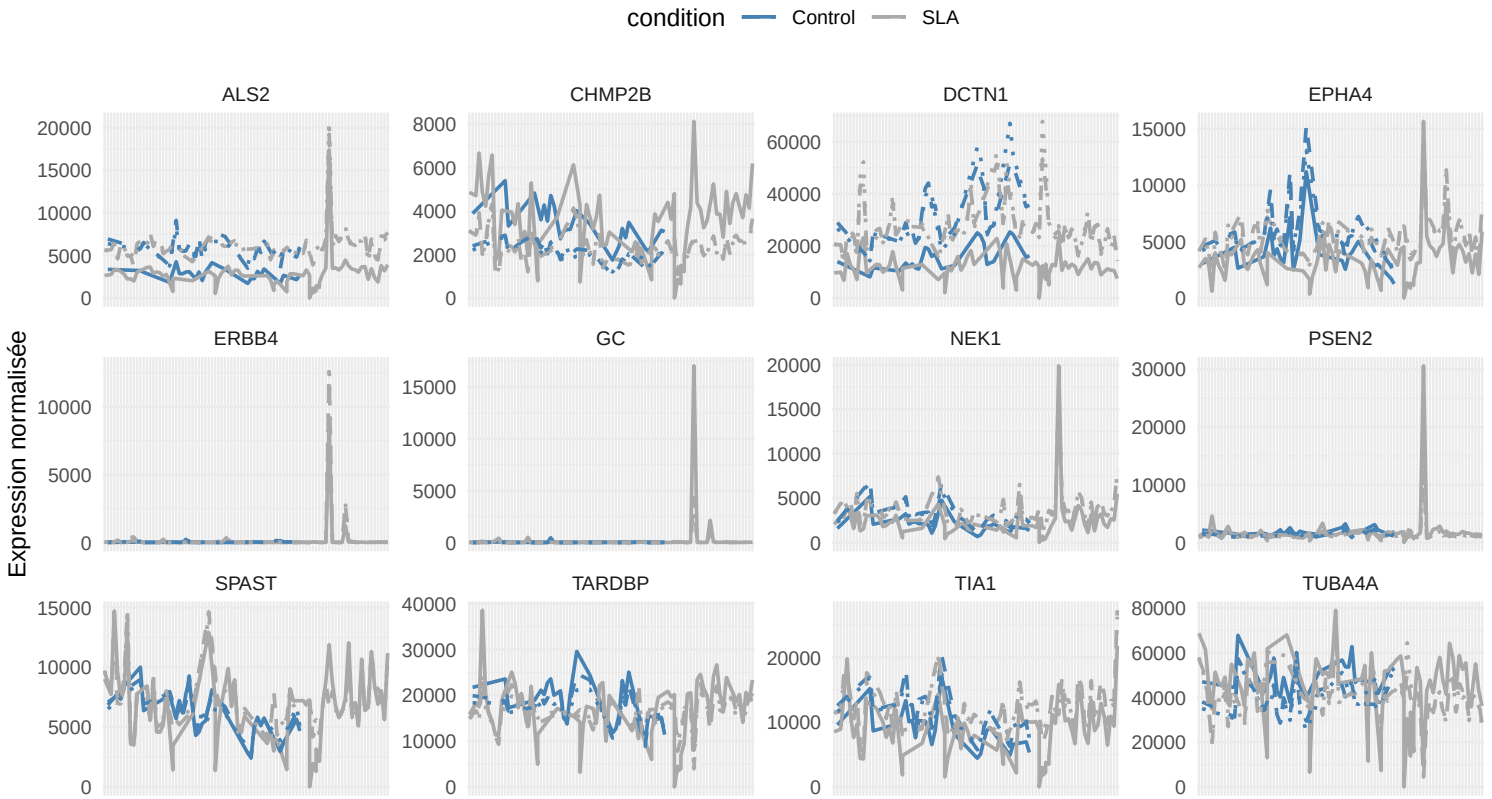


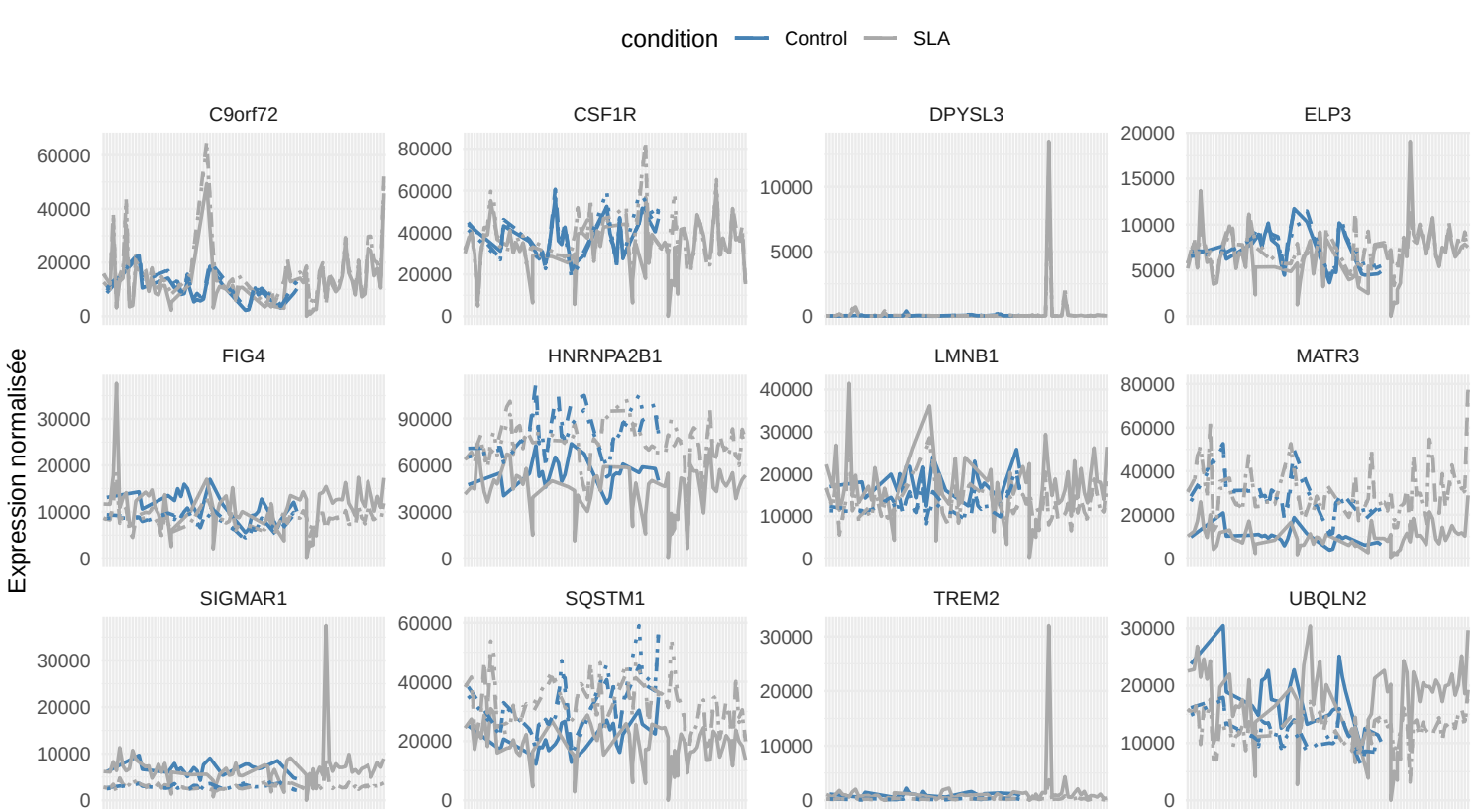
6.5 Visualisation des comptes normalisés

On regarde a premiere vue comment les comptes normalisés ce répartissent par gènes, avec une facet par gène pour les trois approches. L'idée étant surtout ici d'observer si il y'a des différences excessives entre les approches et comprendre dérrière à quoi elles peuvent êtres dues ...

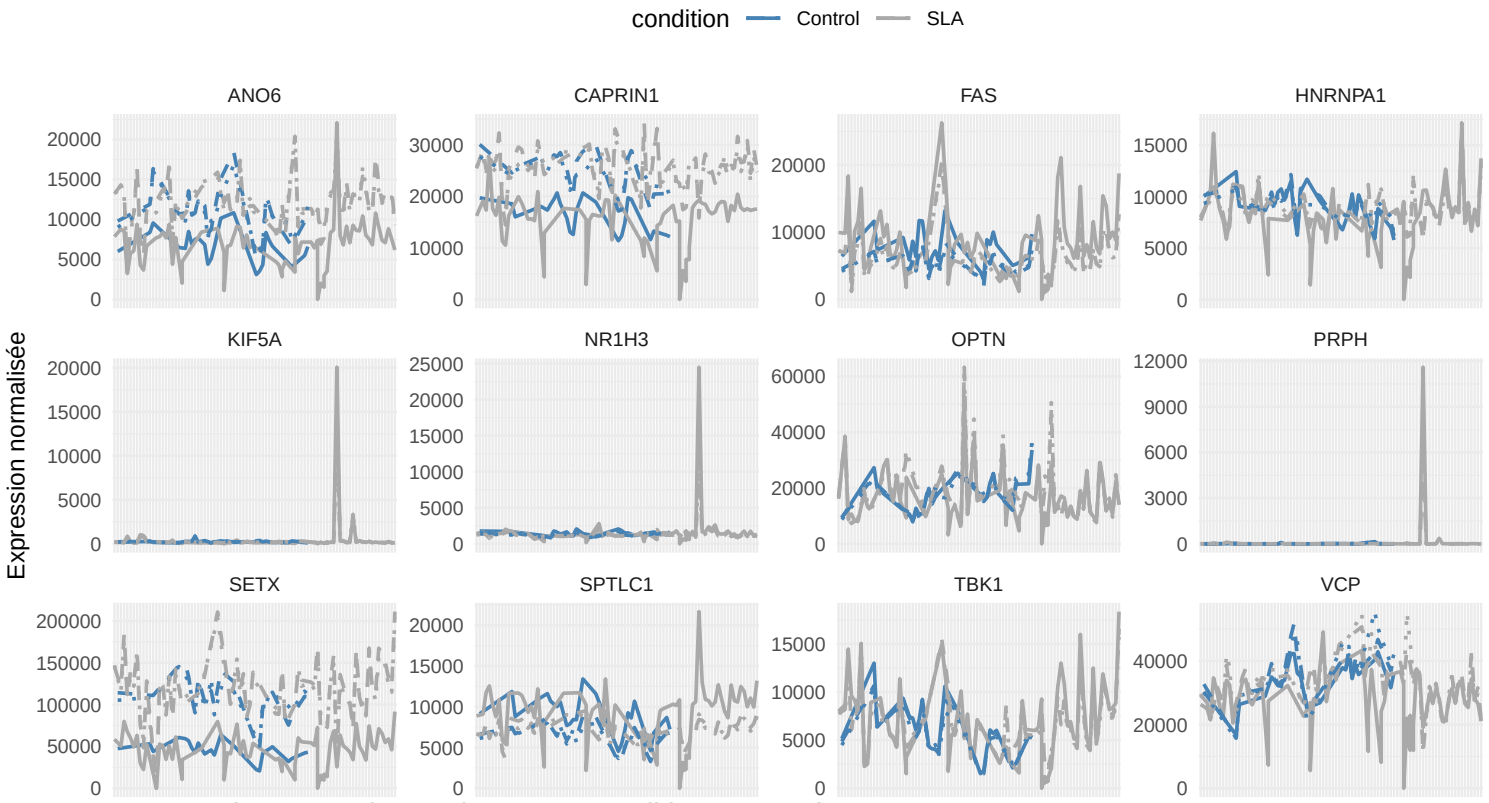
```
liste_genes <- unique(donnees_normalisees$nom_gene)
groupes_genes <- split(liste_genes, ceiling(seq_along(liste_genes) / 12))

donnees_normalisees$condition <- factor(donnees_normalisees$condition, levels = c("Control", "SLA"))

for (indice_pan in seq_along(groupes_genes)) {
  sous_ensemble <- donnees_normalisees %>% filter(nom_gene %in% groupes_genes[[indice_pan]])

  p <- ggplot(sous_ensemble, aes(x = echantillon)) +
    geom_line(aes(y = TPM_transcrits_par_M, color = condition,
                  group = interaction(condition, "TPM")), linewidth = 0.7) +
    geom_line(aes(y = TMM_cpm, color = condition,
                  group = interaction(condition, "TMM")), linewidth = 0.7, linetype = "dashed") +
    geom_line(aes(y = RLE_cpm, color = condition,
                  group = interaction(condition, "RLE")), linewidth = 0.7, linetype = "dotted") +
    facet_wrap(~ nom_gene, scales = "free_y") +
    theme_minimal(base_size = 10) +
    theme(axis.text.x = element_blank(), legend.position = "top") +
    scale_color_manual(values = c("Control" = "steelblue", "SLA" = "darkgray")) +
    labs(title = sprintf("Comparaison TPM / TMM / RLE par condition - Panel %d", indice_pan),
         y = "Expression normalisée", x = NULL)
```

```
  print(p)
}
```

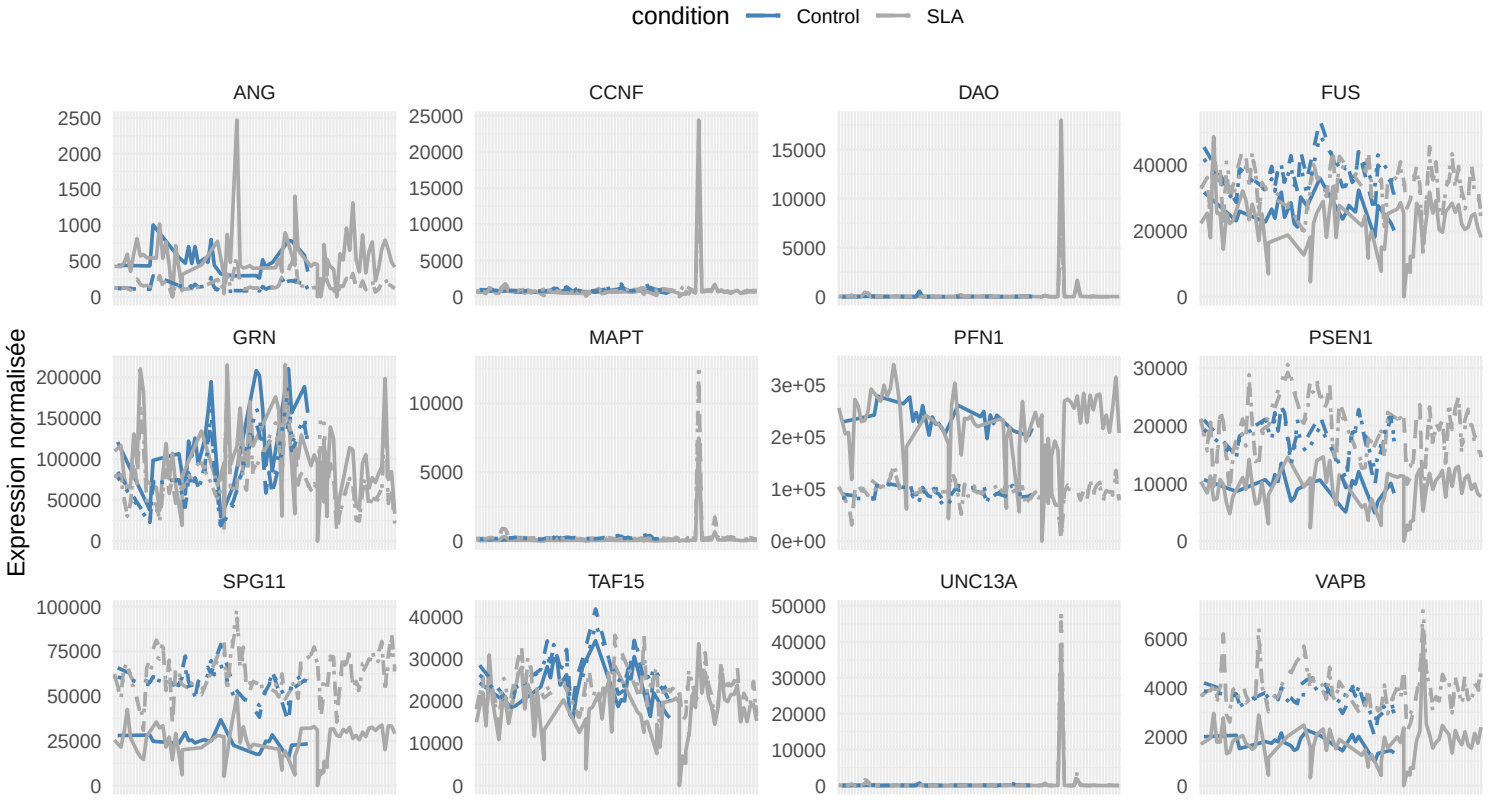

Comparaison TPM / TMM / RLE par condition ... Panel 1
condition
Control
SLA
ALS2
CHMP2B
DCTN1
EPHA4
ERBB4
GC
NEK1
PSEN2
SPAST
TARDBP
TIA1
TUBA4A
Expression normalisée

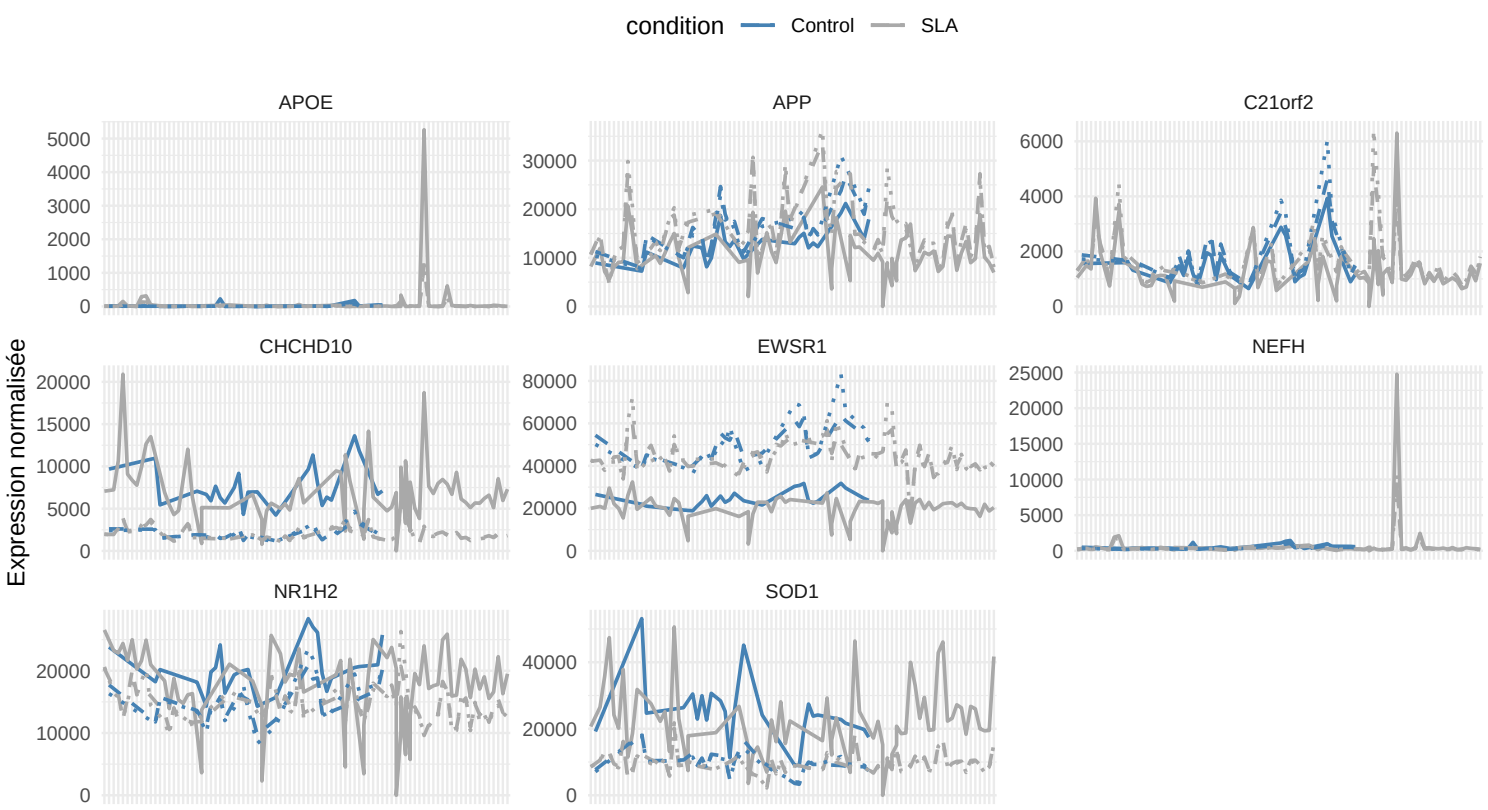

Comparaison TPM / TMM / RLE par condition ... Panel 2
condition
Control
SLA
C9orf72
CSF1R
DPYSL3
ELP3
FIG4
HNRNPA2B1
LMNB1
MATR3
SIGMAR1
SQSTM1
TREM2
UBQLN2
Expression normalisée

Comparaison TPM / TMM / RLE par condition ... Panel 3



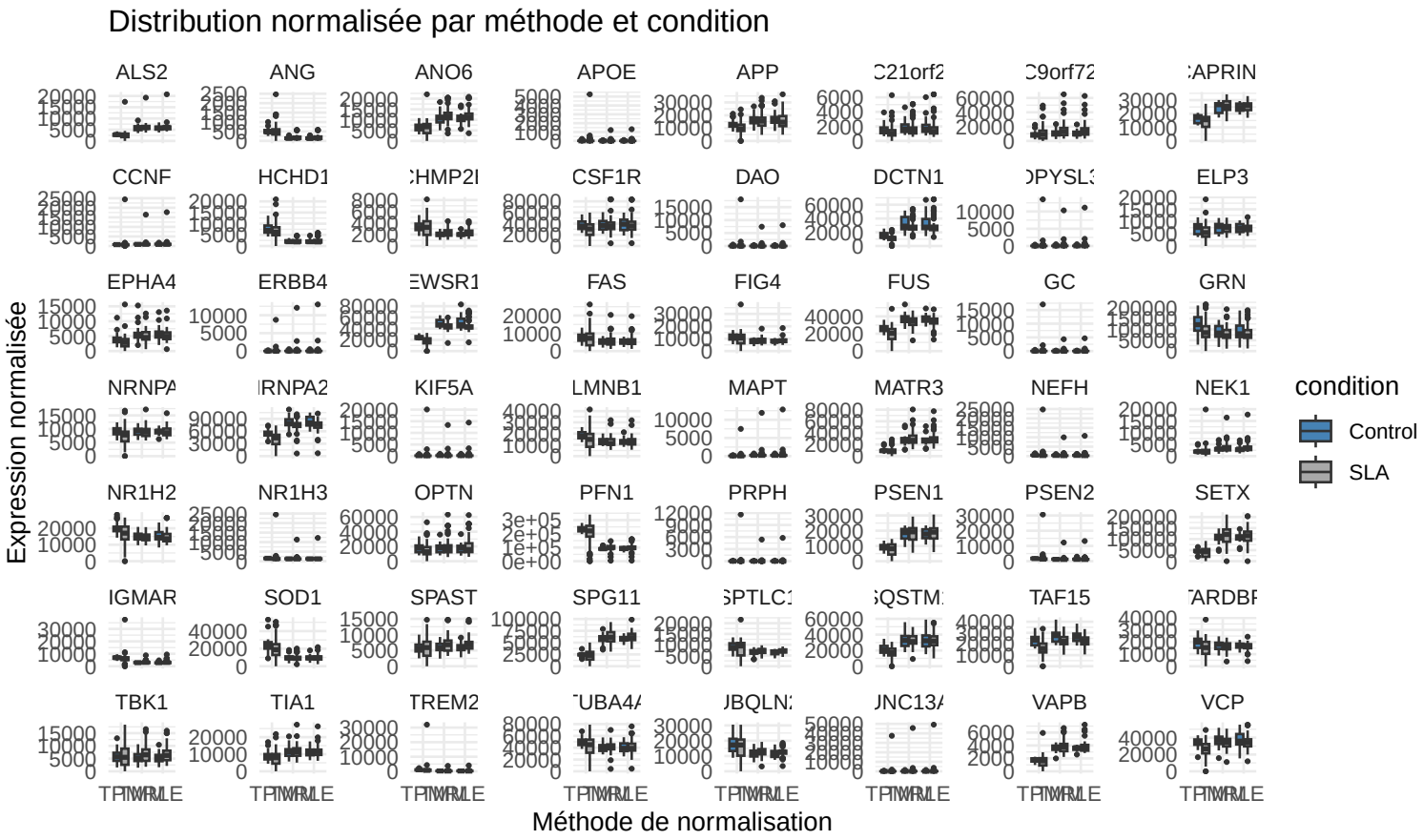
Comparaison TPM / TMM / RLE par condition ... Panel 4





6.6 Comparaison visuelle des normalisations

```
ggplot(donnees_long, aes(x = methode, y = expression_norm, fill = condition)) +
  geom_boxplot(outlier.size = 0.5) +
  facet_wrap(~ nom_gene, scales = "free_y") +
  theme_minimal() +
  labs(title = "Distribution normalisée par méthode et condition",
       x = "Méthode de normalisation",
       y = "Expression normalisée") +
  scale_fill_manual(values = c("Control" = "steelblue", "SLA" = "darkgray"))
```



6.6.1 1. MA-plot TPM

```
tpm_ma <- tpm_long %>%
  group_by(nom_gene, condition) %>%
  summarise(expr = mean(TPM_transcrits_par_M), .groups="drop") %>%
  pivot_wider(names_from = condition, values_from = expr) %>%
  mutate(
    aveExpr = log2((Control + SLA)/2 + 1),
    logFC   = log2(SLA + 1) - log2(Control + 1),
    significatif = abs(logFC) > 1
  )
```

6.6.2 2. MA-plot TMM

7 LRT déjà calculé → lrt

```
tmm_res <- as_tibble(resultats_diff_expr, rownames = "nom_gene") %>%
  select(nom_gene, logFC, logCPM, FDR) %>%
  mutate(
    aveExpr = logCPM,            # déjà en log2 CPM moyens
    significatif = FDR < 0.05
  )
```

7.0.1 3. MA-plot RLE

```
dge_rle <- DGEList(counts = as.matrix(comptages_larges))
dge_rle <- calcNormFactors(dge_rle, method = "RLE")
dge_rle <- estimateDisp(dge_rle, design)
```

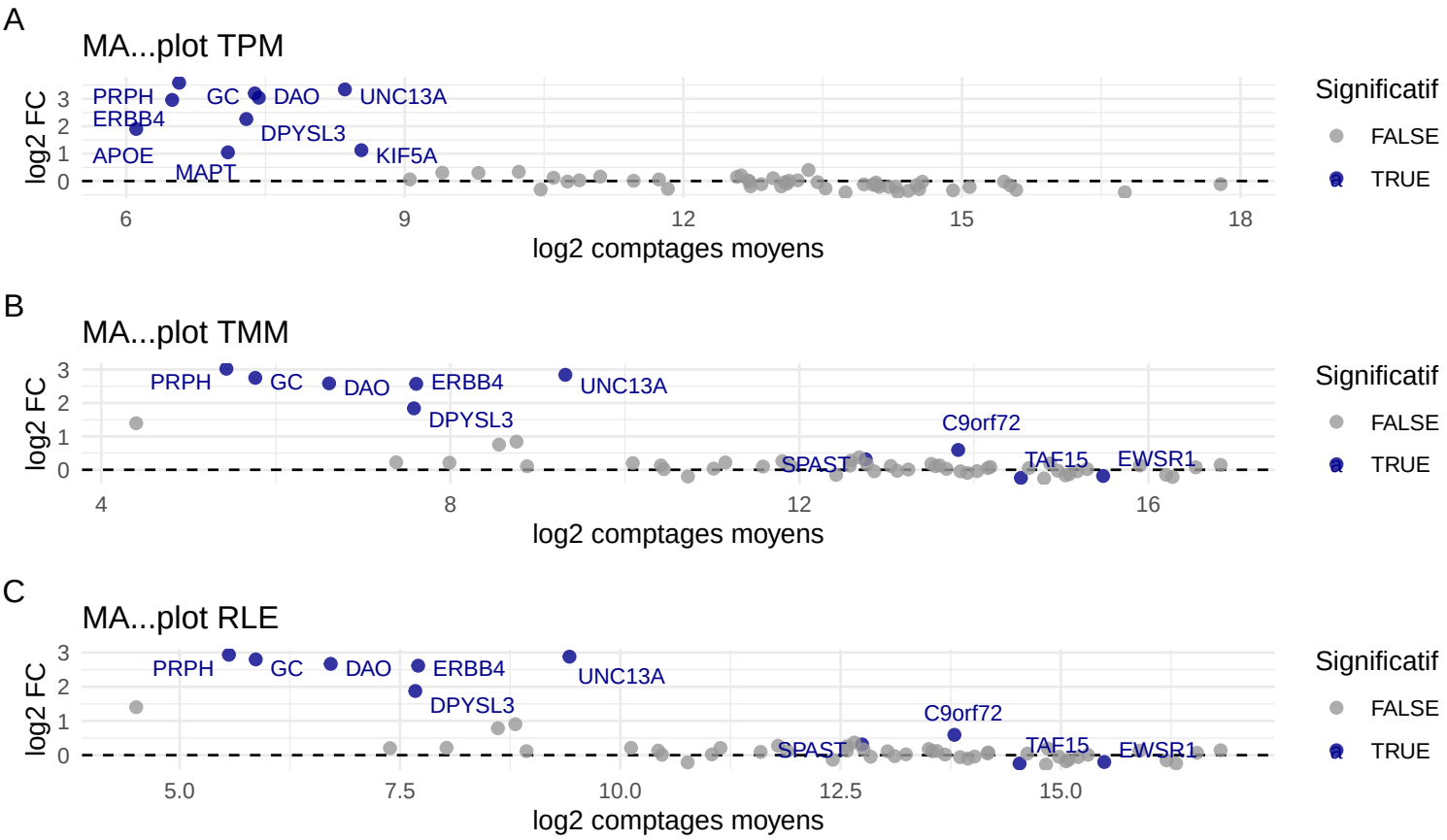
```r
fit_rle <- glmFit(dge_rle, design)
lrt_rle <- glmLRT(fit_rle, coef = 2)
rle_res <- topTags(lrt_rle, n = Inf)$table %>%
  as_tibble(rownames = "nom_gene") %>%
  mutate(
    aveExpr = logCPM,
    significatif = FDR < 0.05
  )
```

7.0.2 4. Fonction générique de tracé

Dans les MA-plot ci dessous je cherche à regarde l'expression moyenne des gènes par la moyenne des comptages en face de la variation relative d'expression du dit gène entre ces deux conditions.

x : Représente le niveau moyen d'expression normalisée (log2) d'un gène dans les deux conditions comparées (sla et controle).

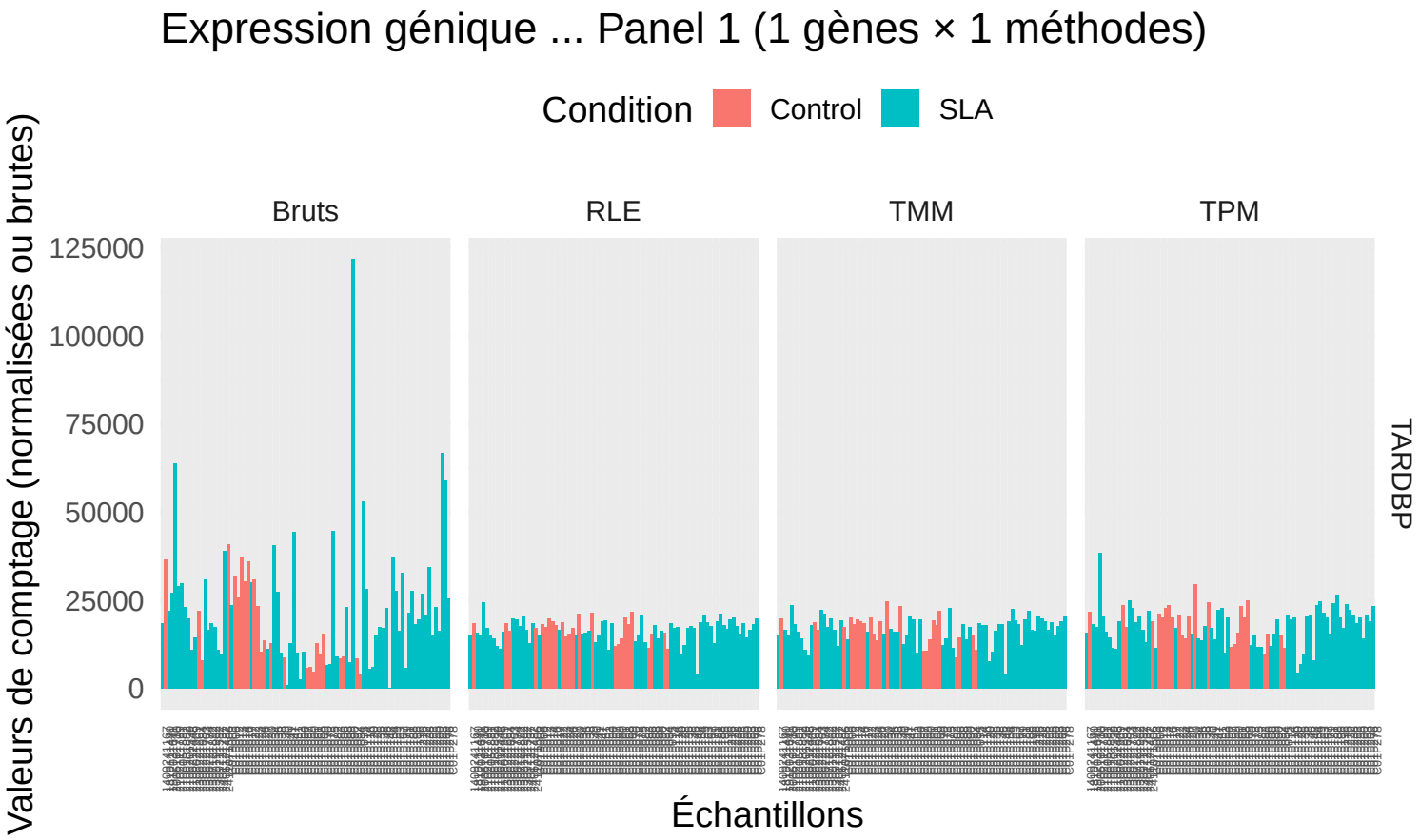
```r
trace_ma <- function(df, titre) {
  ggplot(df, aes(x = aveExpr, y = logFC, colour = significatif)) +
    geom_hline(yintercept = 0, linetype = "dashed") +
    geom_point(size = 2, alpha = 0.8) +
    scale_colour_manual(values = c(`TRUE` = "darkblue", `FALSE` = "grey60")) +
    coord_cartesian(ylim = c(min(df$logFC), max(df$logFC))) +

    geom_text_repel(
      data = df %>% filter(significatif),       # que les points significatifs
      aes(label = nom_gene),
      size = 3,
      box.padding = 0.3,
      point.padding = 0.5,
      max.overlaps = Inf                         # pour afficher tous les labels
    ) +
    labs(title = titre,
         x = "log2 comptages moyens",
         y = "log2 FC",
         colour = "Significatif") +
    theme_minimal(base_size = 11)
}

p_tpm <- trace_ma(tpm_ma, "MA-plot TPM")
p_tmm <- trace_ma(tmm_res, "MA-plot TMM")
p_rle <- trace_ma(rle_res, "MA-plot RLE")

(p_tpm / p_tmm / p_rle) +
  plot_annotation(tag_levels = "A")                  # tags A/B/C optionnels
```

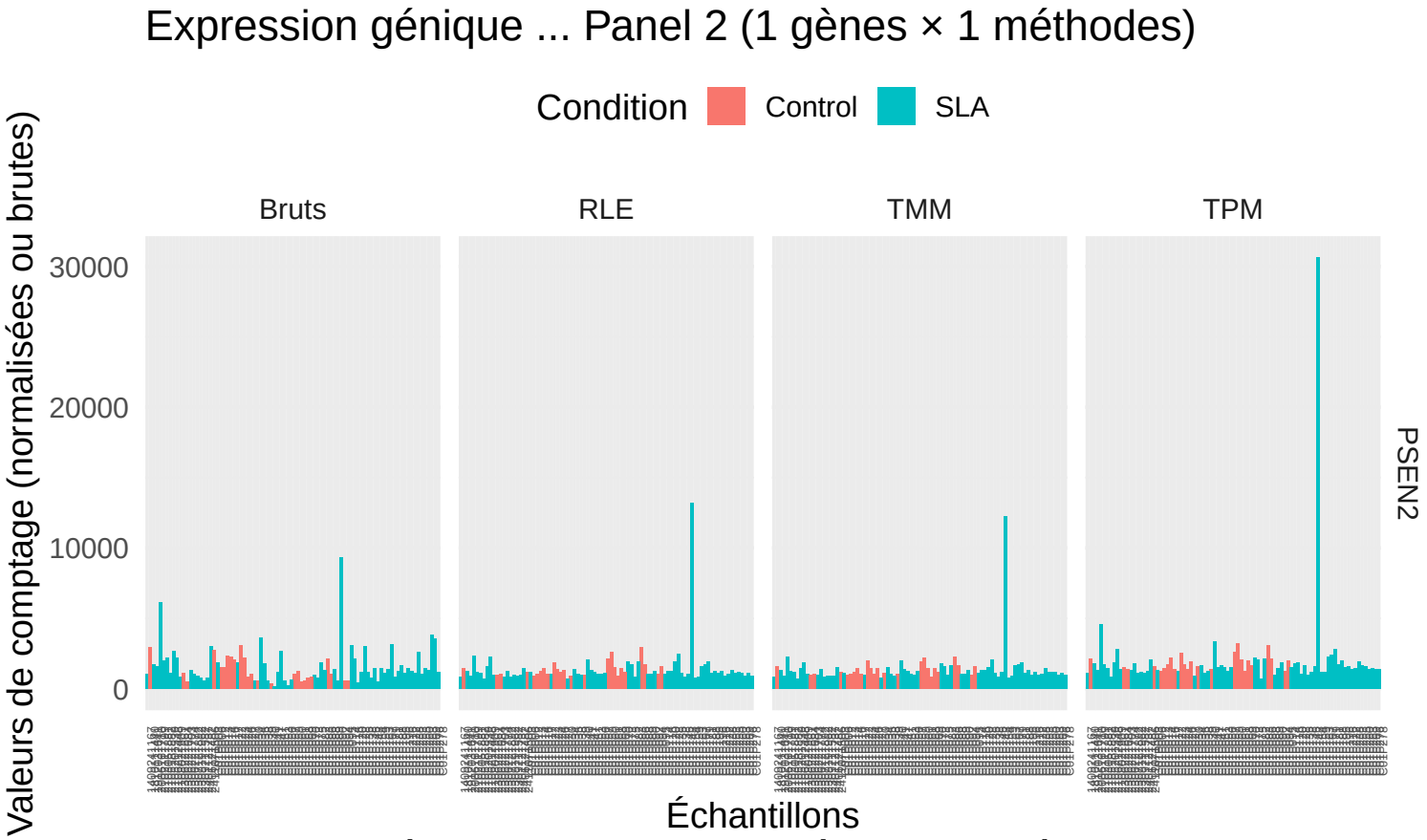


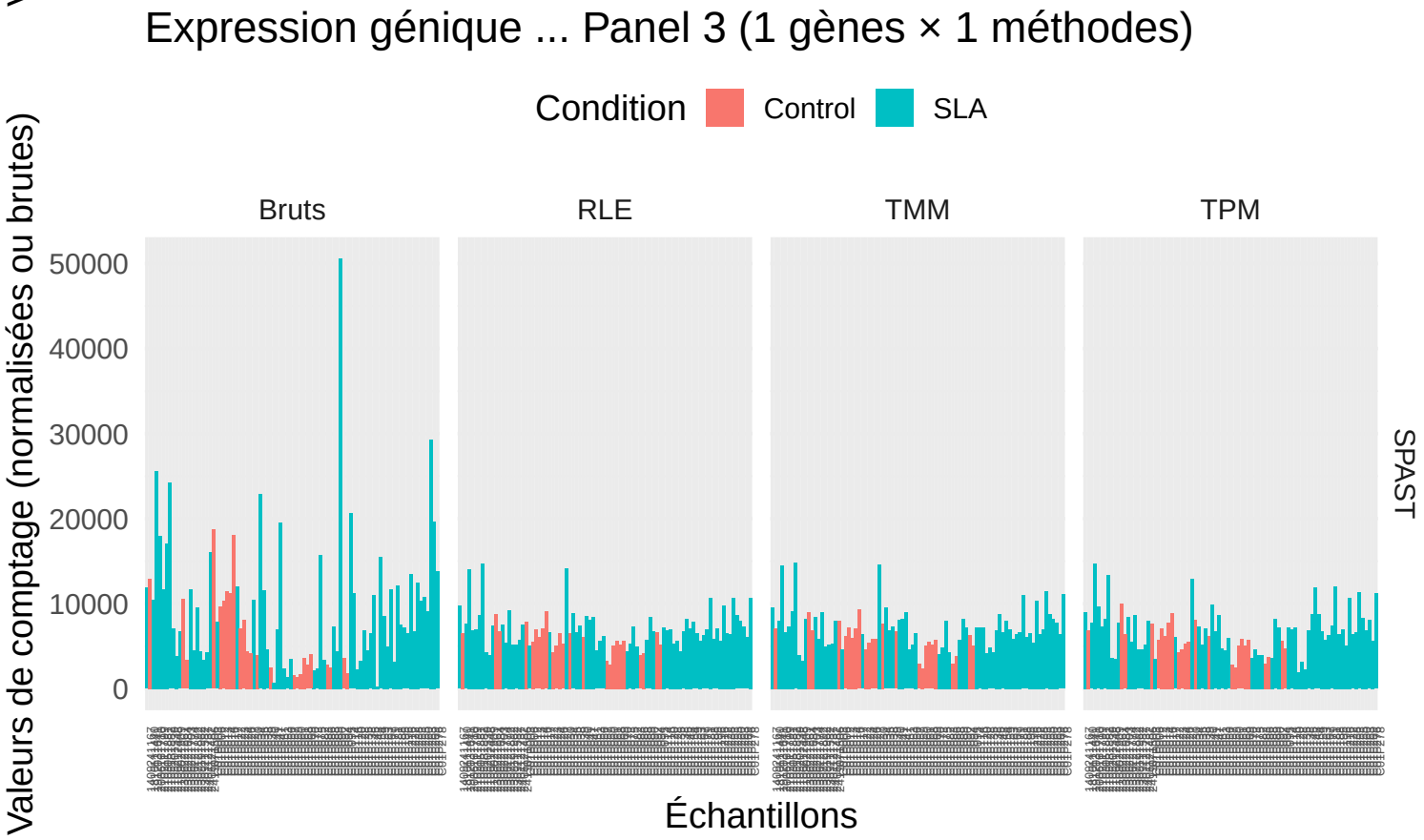
8 Comptes normalisés et bruts par gène, méthode et patient

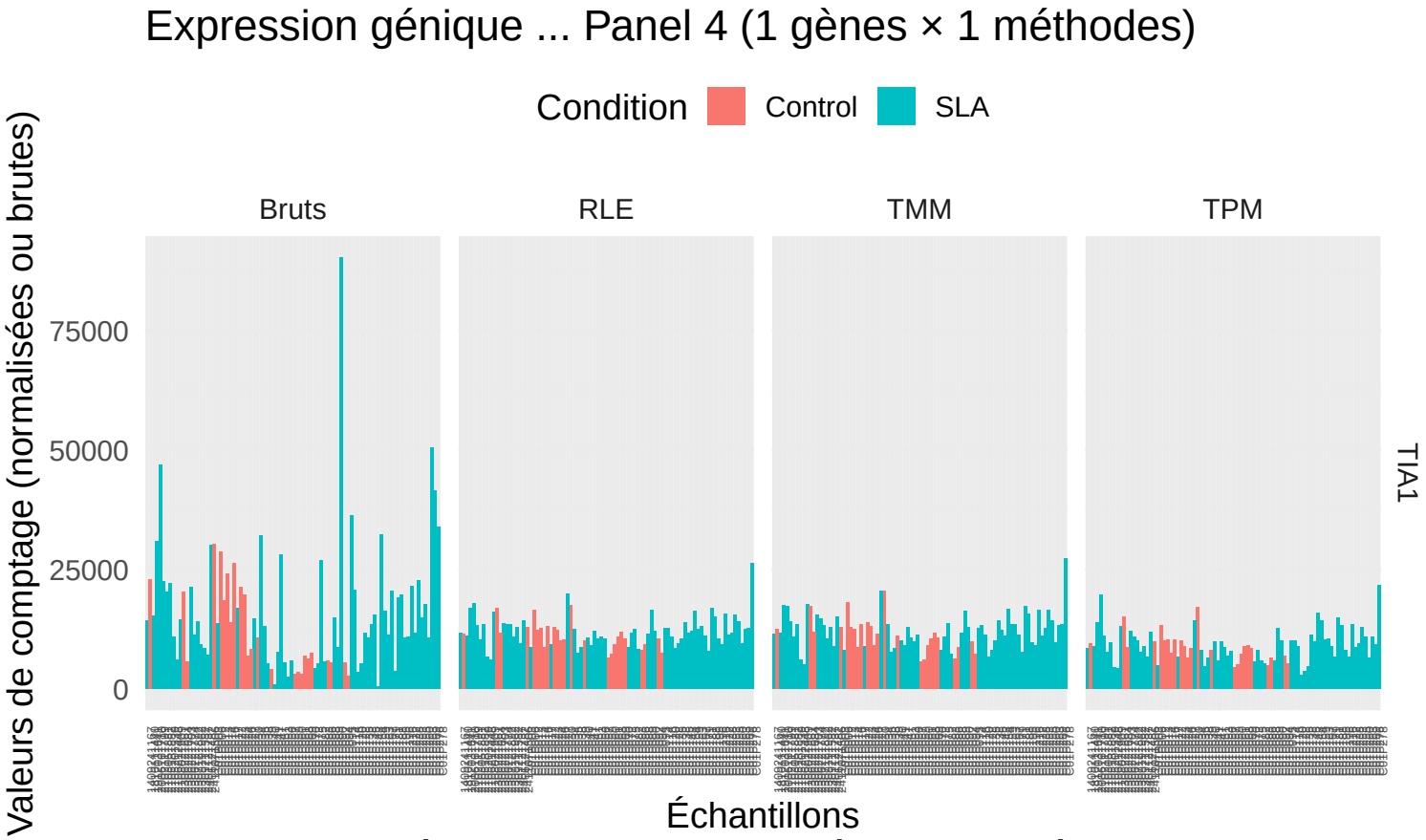
```
vars_y        <- c("TPM_transcrits_par_M", "TMM_cpm", "RLE_cpm", "total_lectures")
ncol_facets   <- 1
genes_par_fig <- 1
genes_par_row <- 1

donnees_longues <- donnees_normalisees %>%
  select(echantillon, nom_gene, condition, all_of(vars_y)) %>%
  pivot_longer(cols = all_of(vars_y), names_to = "methode", values_to = "expr") %>%
  mutate(methode = recode(methode,
                          "TPM_transcrits_par_M" = "TPM",
                          "TMM_cpm"              = "TMM",
                          "RLE_cpm"              = "RLE",
                          "total_lectures"       = "Bruts"))

# 3) Découpage en groupes de gènes pour paginer ------------------------------
cut_genes <- split(unique(donnees_longues$nom_gene),
                   ceiling(seq_along(unique(donnees_longues$nom_gene)) / genes_par_fig))

# 4) Boucle de génération des graphiques -------------------------------------
for (i in seq_along(cut_genes)) {

  p <- donnees_longues %>%
    filter(nom_gene %in% cut_genes[[i]]) %>%
    ggplot(aes(x = echantillon, y = expr, fill = condition)) +
    geom_col(position = "dodge") +
    facet_grid(rows = vars(nom_gene), cols = vars(methode), scales = "free_y") +
```

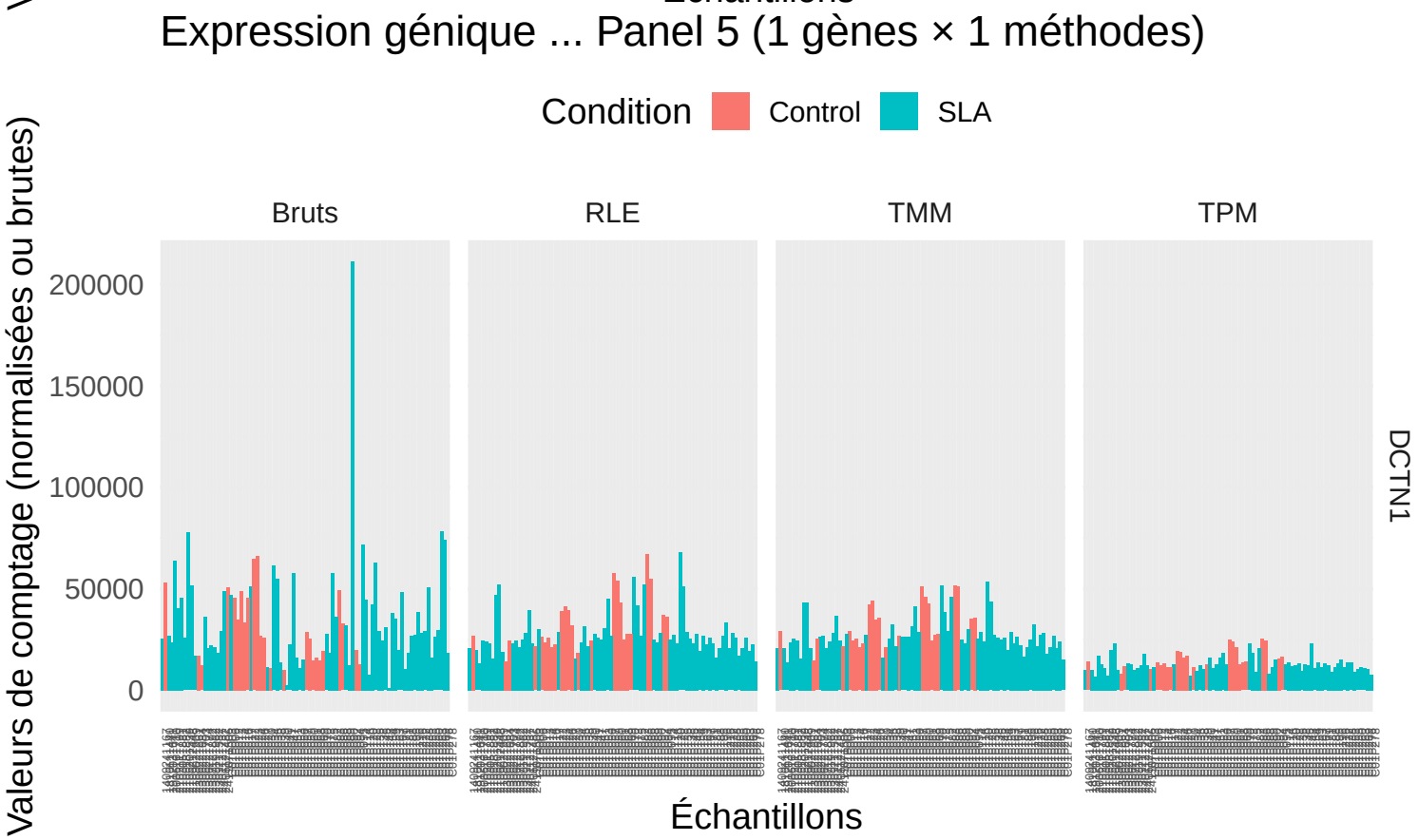
```
    theme_minimal(base_size = 15) +
    theme(axis.text.x = element_text(angle = 90, hjust = 1, size = 5),
          strip.text = element_text(size = 12),
          legend.position = "top") +
    labs(title = sprintf("Expression génique - Panel %d (%d gènes × %d méthodes)",
                         i, genes_par_fig, ncol_facets),
         x = "Échantillons",
         y = "Valeurs de comptage (normalisées ou brutes)",
         fill = "Condition")
  print(p)
}
```

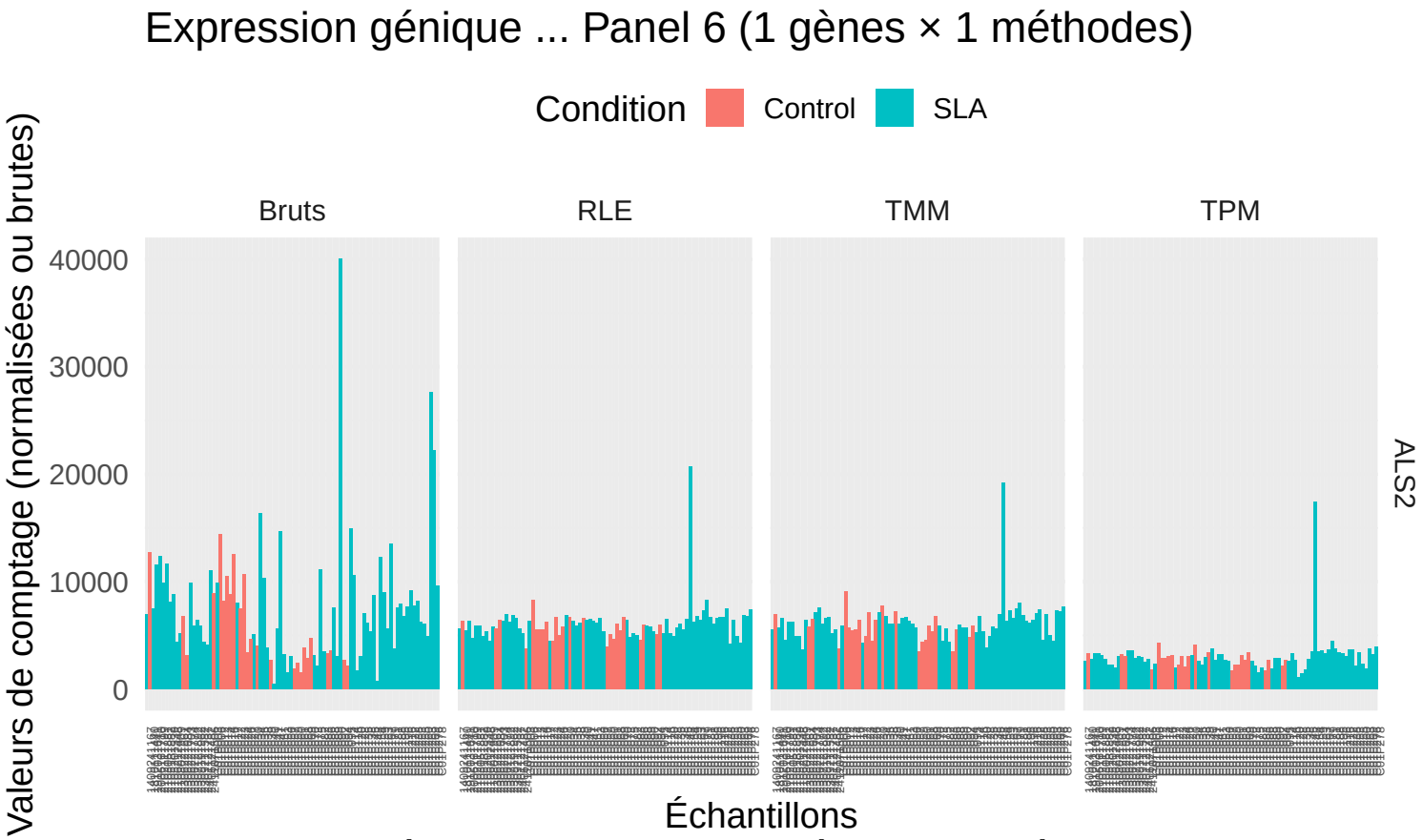


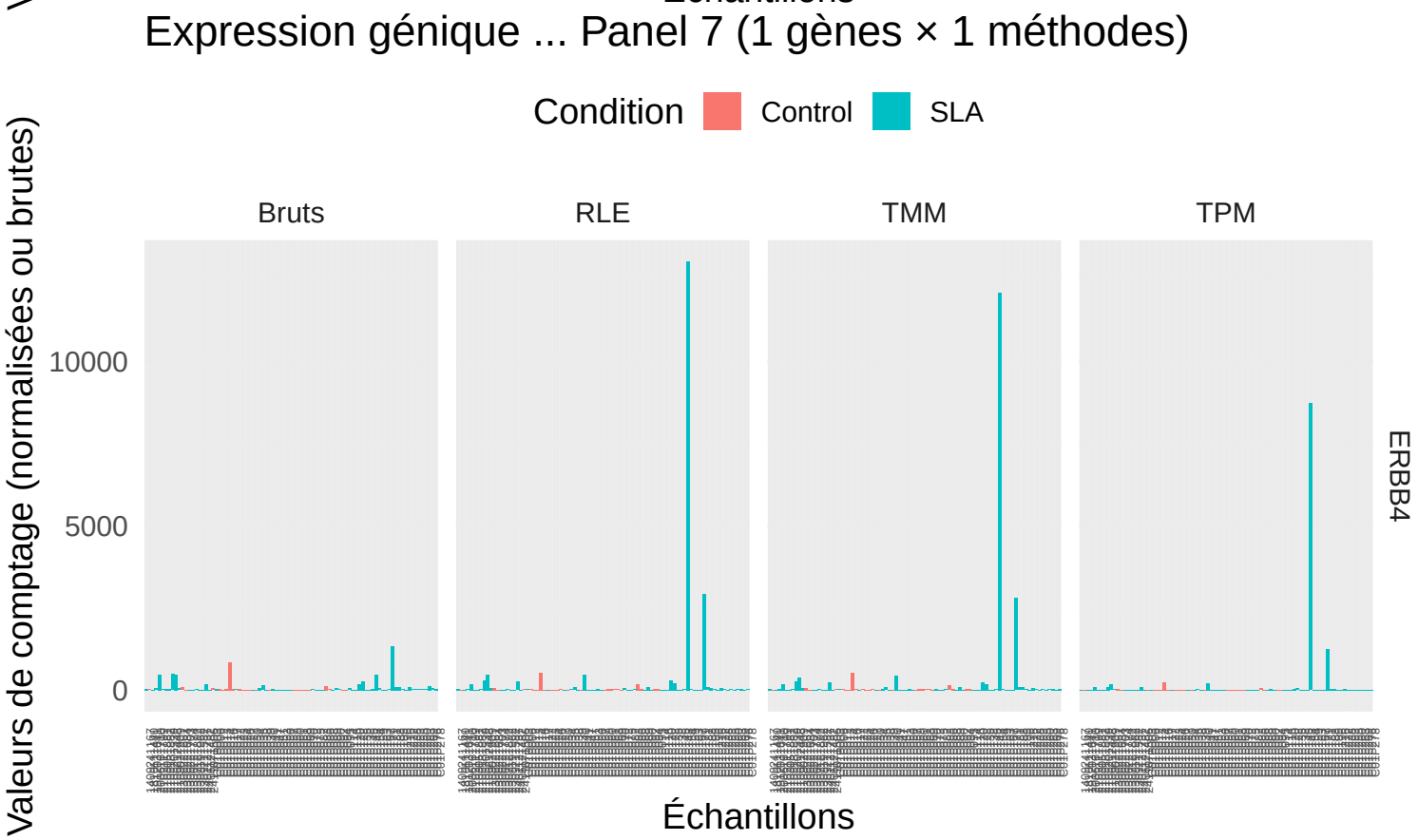

Expression génique ... Panel 2 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
30000
20000
10000
0
PSEN2
Valeurs de comptage (normalisées ou brutes)
Échantillons

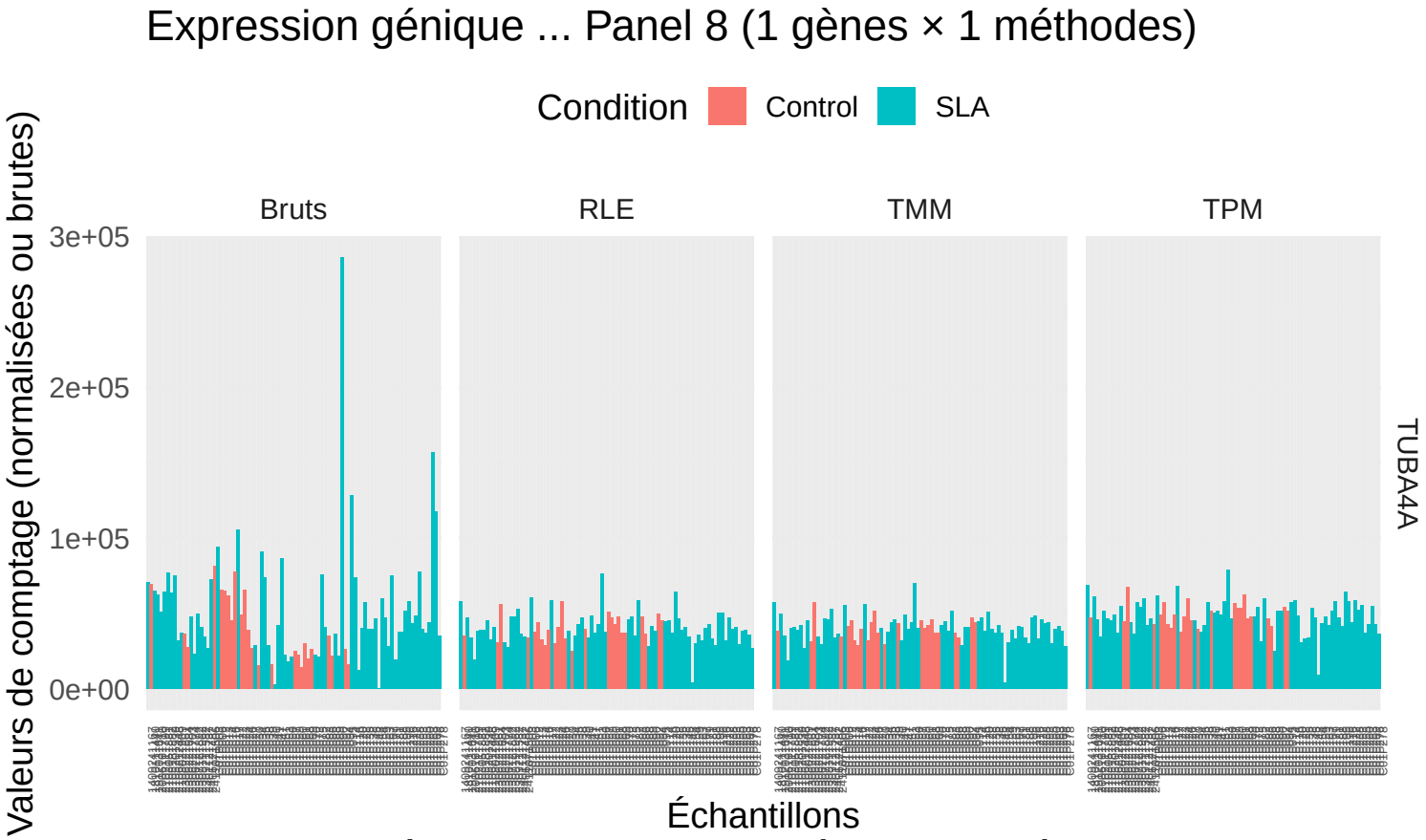

Expression génique ... Panel 3 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
50000
40000
30000
20000
10000
0
SPAST
Valeurs de comptage (normalisées ou brutes)
Échantillons

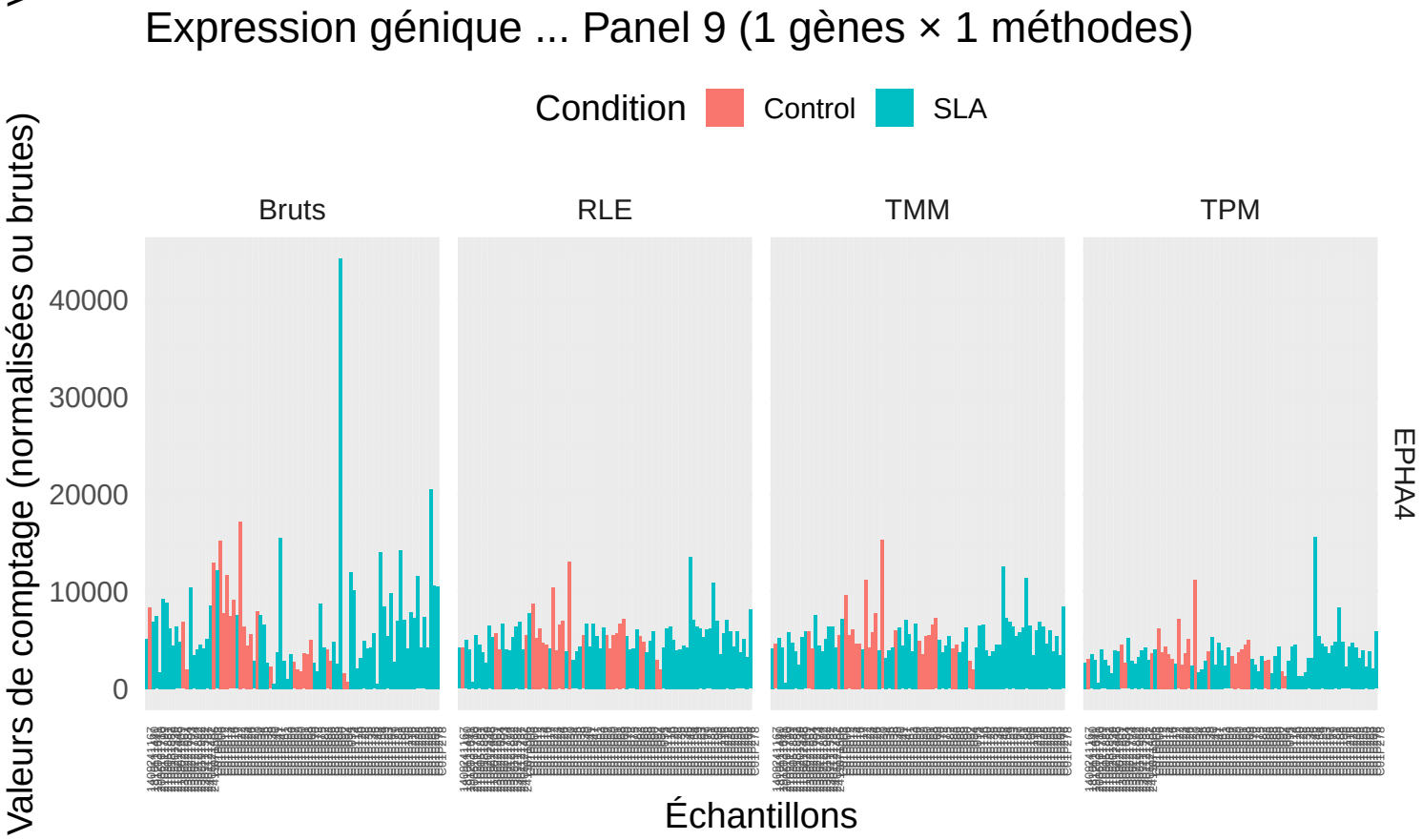

Expression génique ... Panel 4 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
75000
50000
25000
0
TIA1
Échantillons

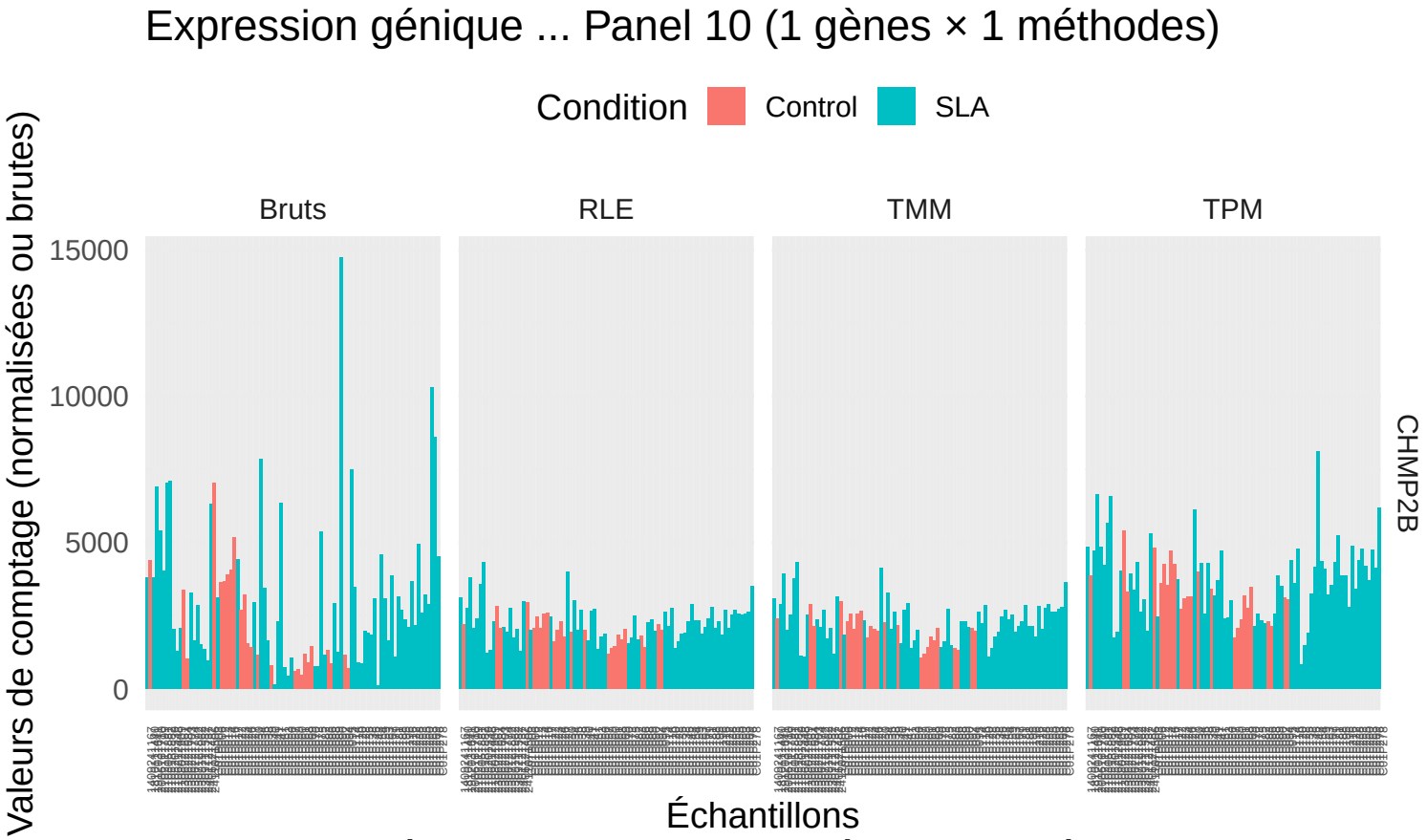

Expression génique ... Panel 5 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
200000
150000
100000
50000
0
DCTN1
Échantillons

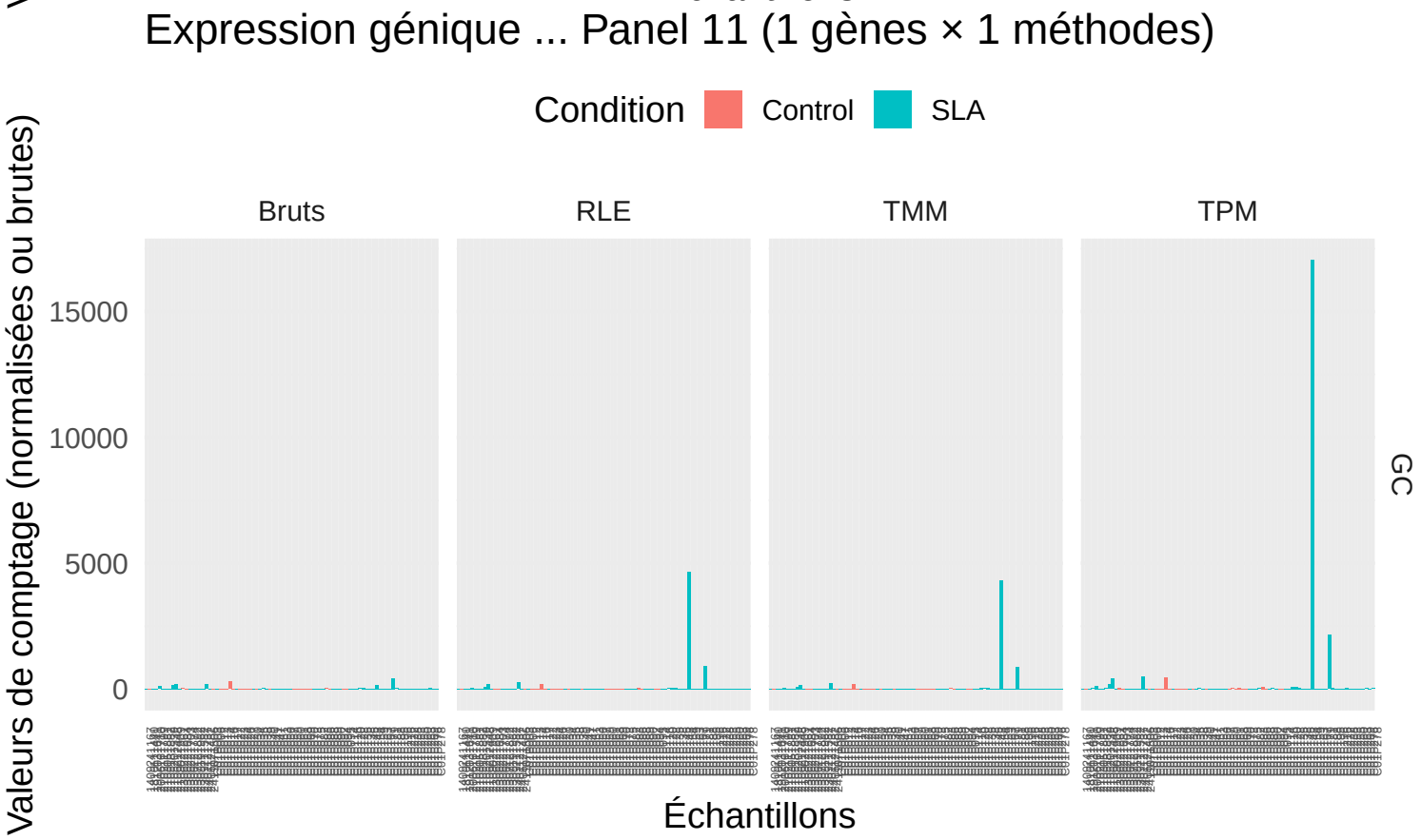

Expression génique ... Panel 6 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
40000
30000
20000
10000
0
Valeurs de comptage (normalisées ou brutes)
ALS2
Échantillons

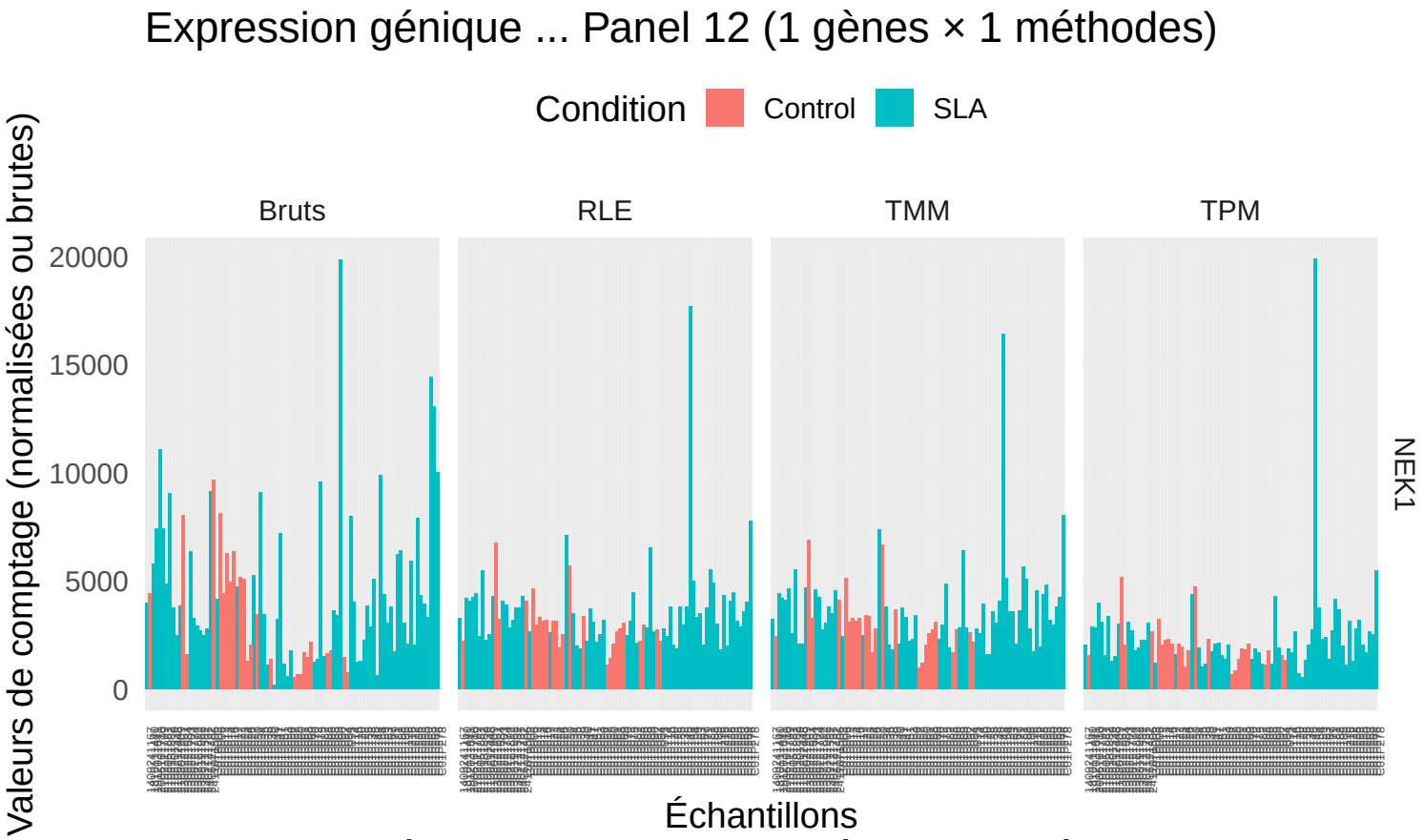

Expression génique ... Panel 7 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
10000
5000
0
Valeurs de comptage (normalisées ou brutes)
ERBB4
Échantillons

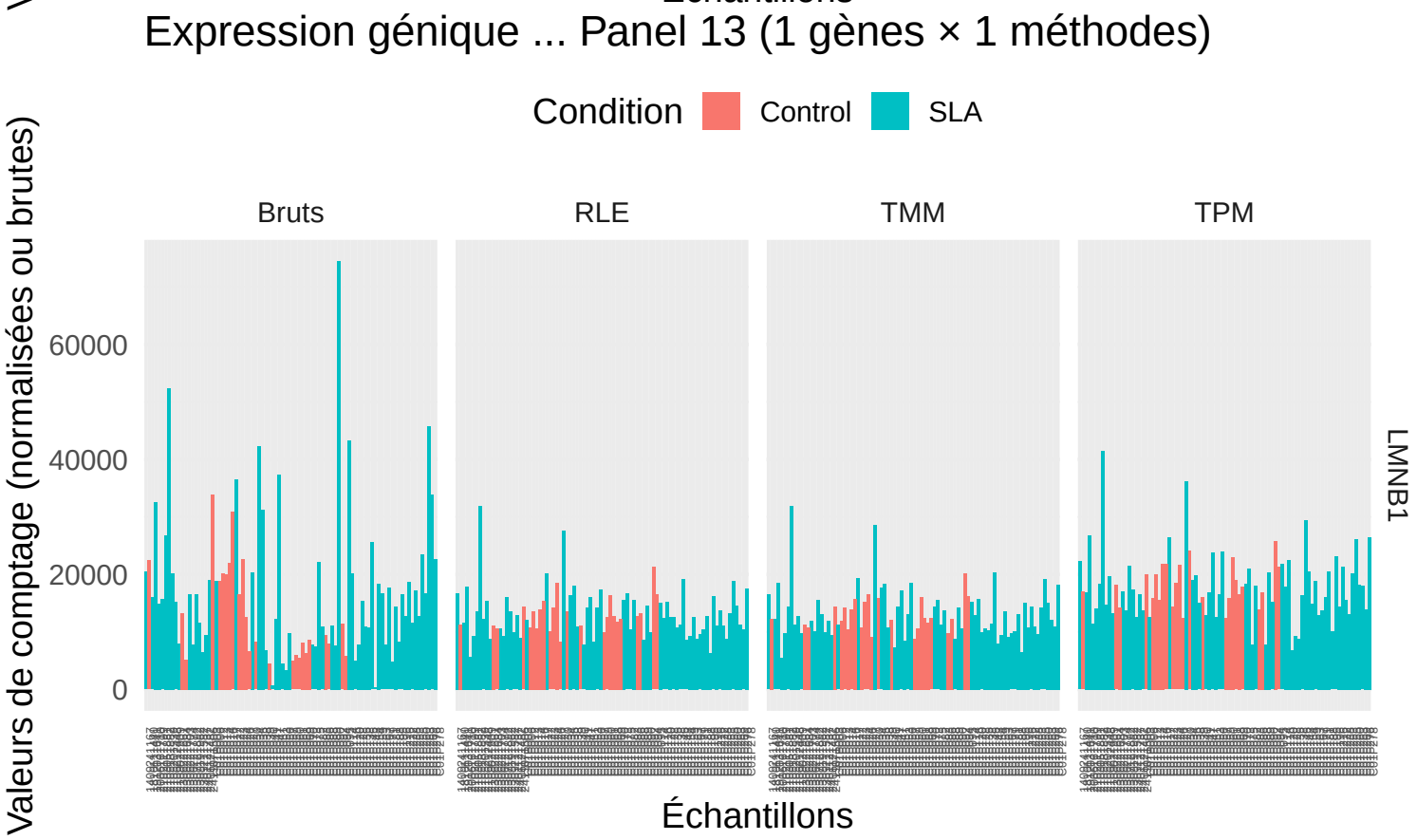

Expression génique ... Panel 8 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
3e+05
2e+05
1e+05
0e+00
Valeurs de comptage (normalisées ou brutes)
TUBA4A
Échantillons

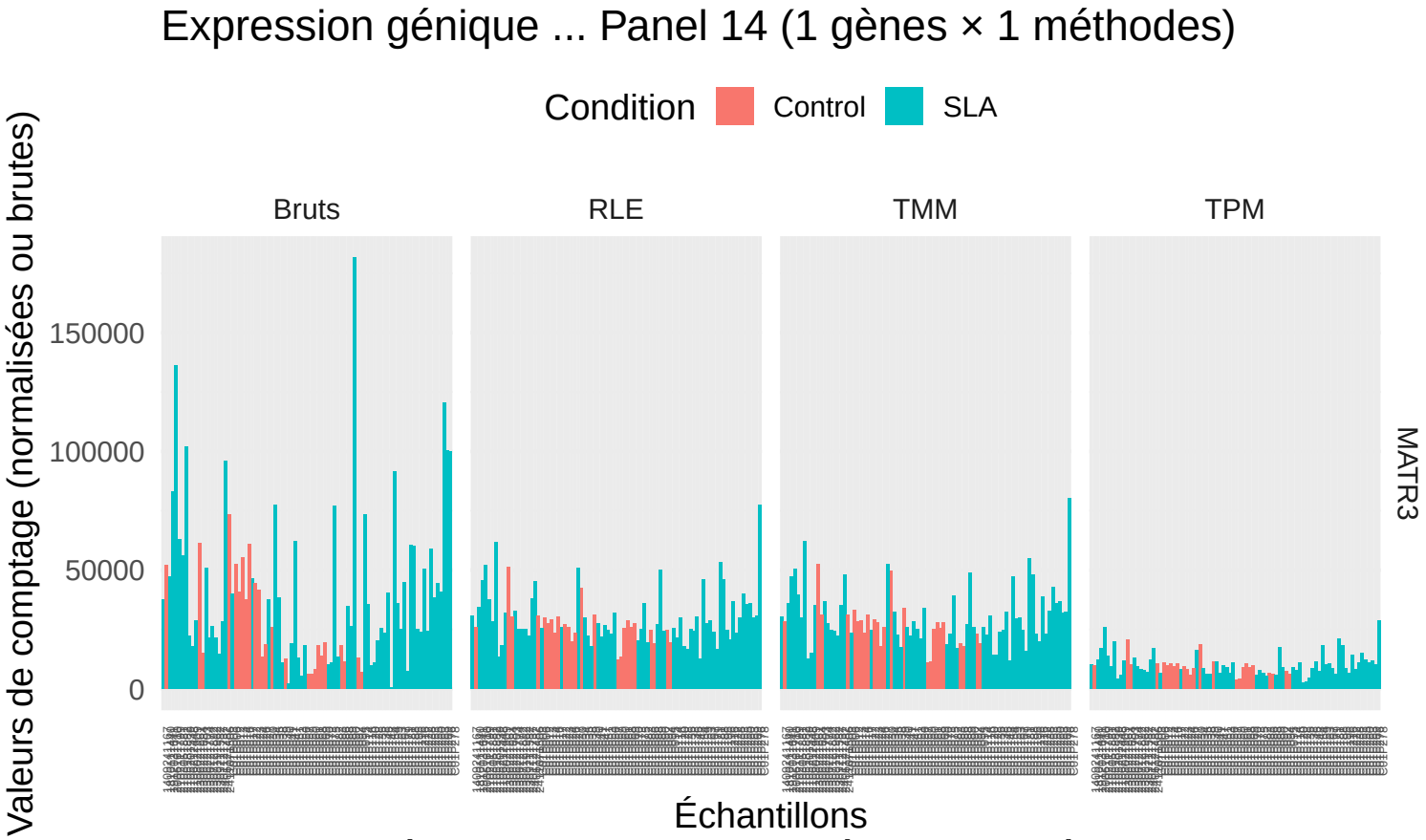

Expression génique ... Panel 9 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
40000
30000
20000
10000
0
Valeurs de comptage (normalisées ou brutes)
EPHA4
Échantillons

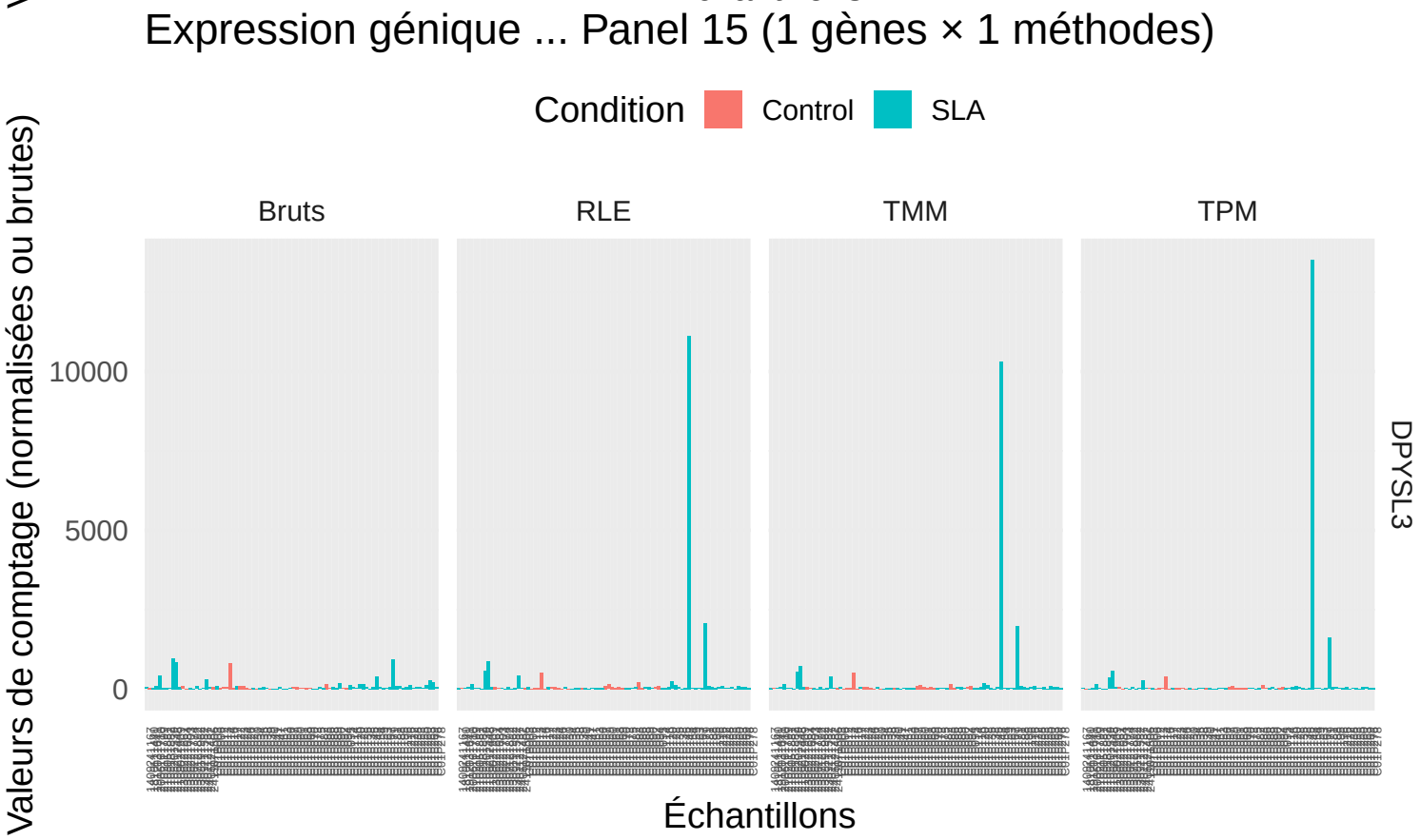

Expression génique ... Panel 10 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
15000
10000
5000
0
Valeurs de comptage (normalisées ou brutes)
CHMP2B
Échantillons

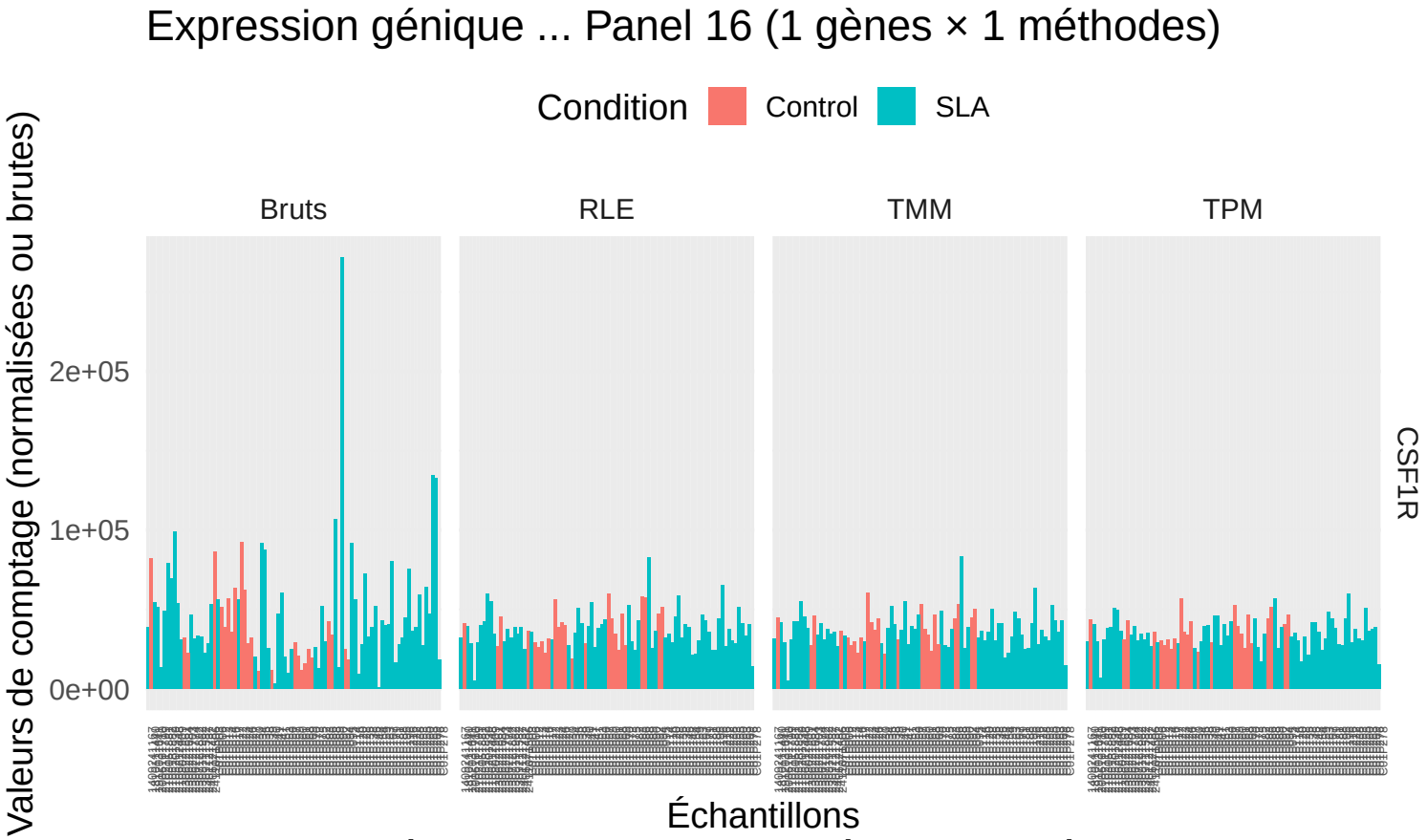

Expression génique ... Panel 11 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
15000
10000
5000
0
Valeurs de comptage (normalisées ou brutes)
GC
Échantillons

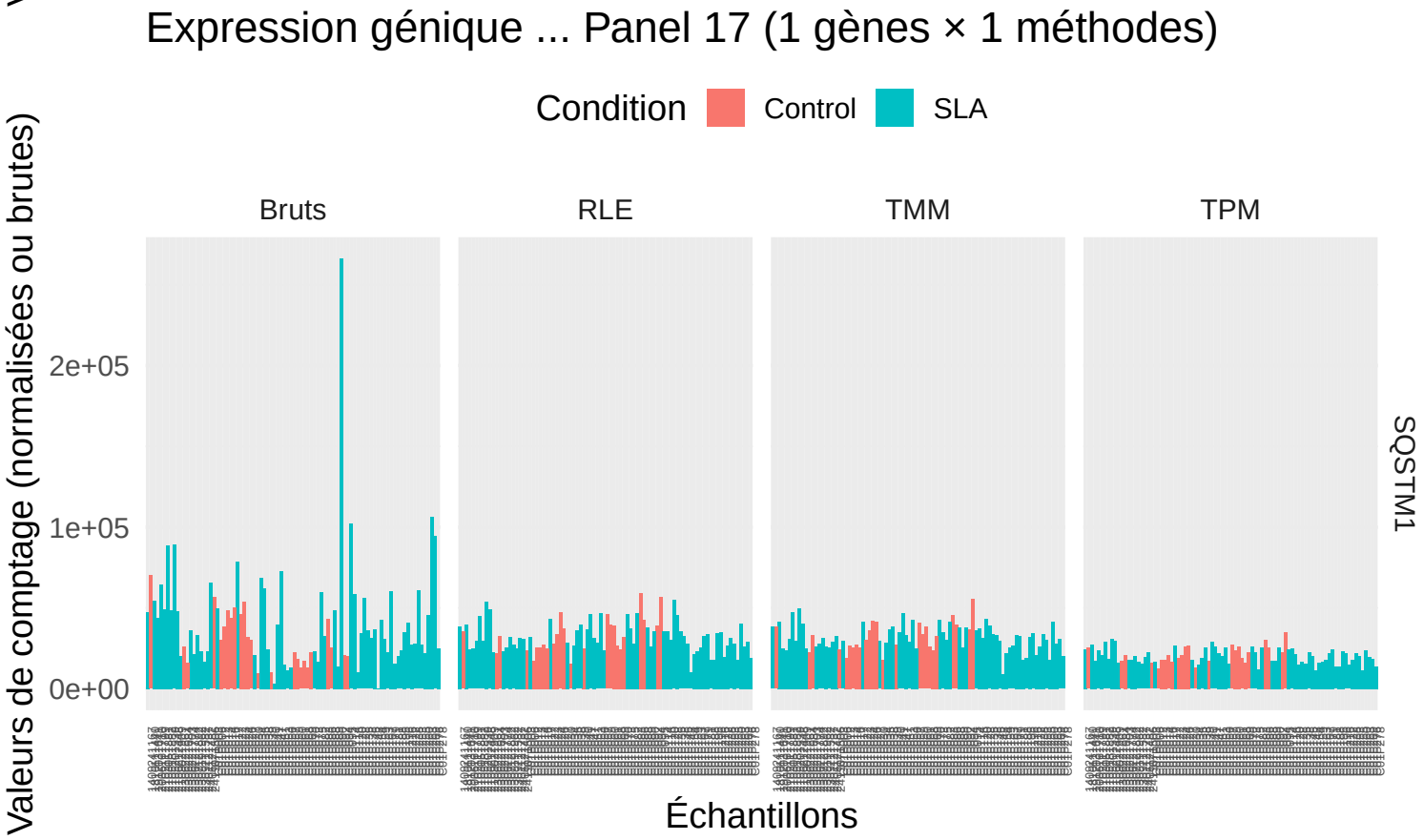

Expression génique ... Panel 12 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
20000
15000
10000
5000
0
NEK1
Échantillons

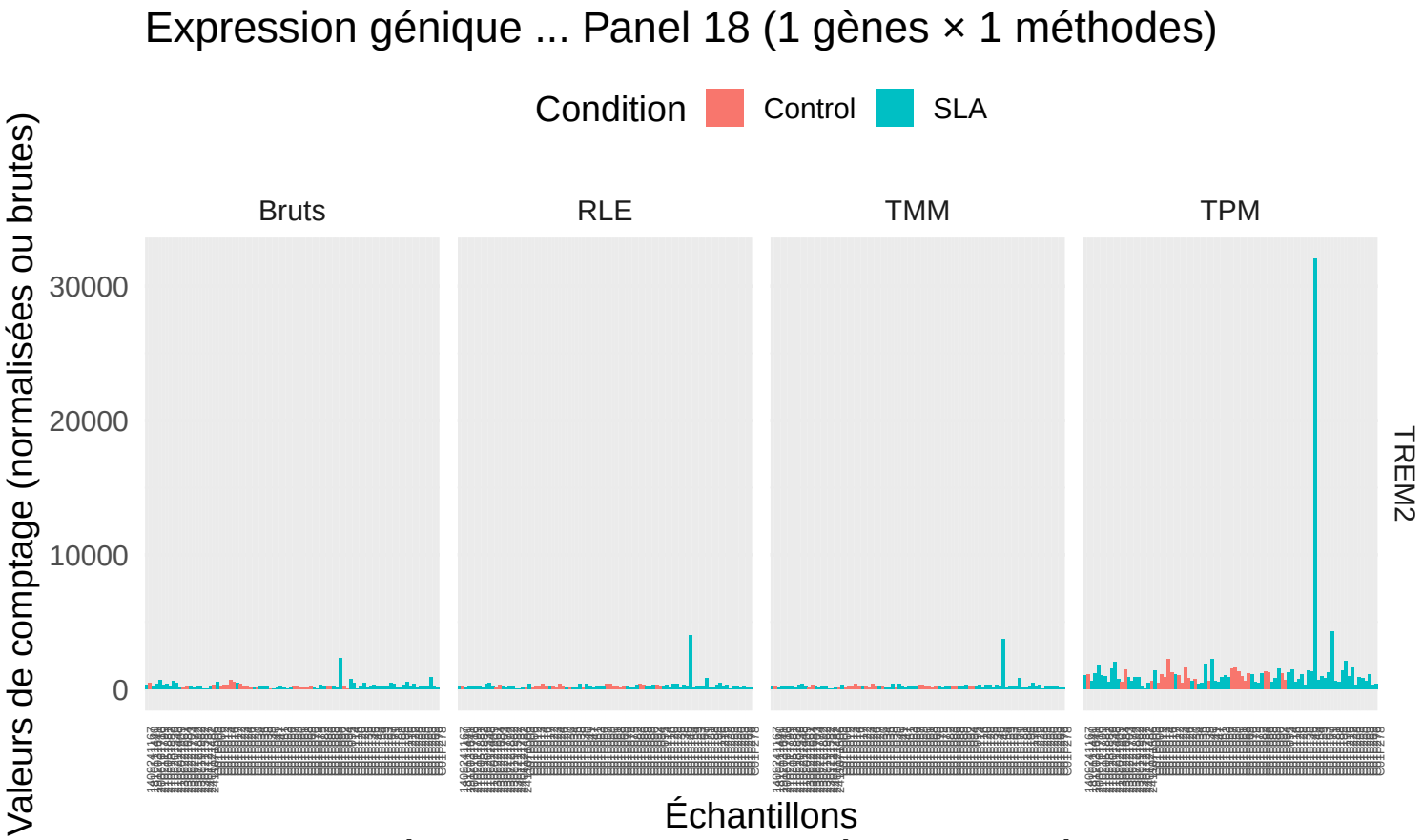

Expression génique ... Panel 13 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
60000
40000
20000
0
LMNB1
Échantillons

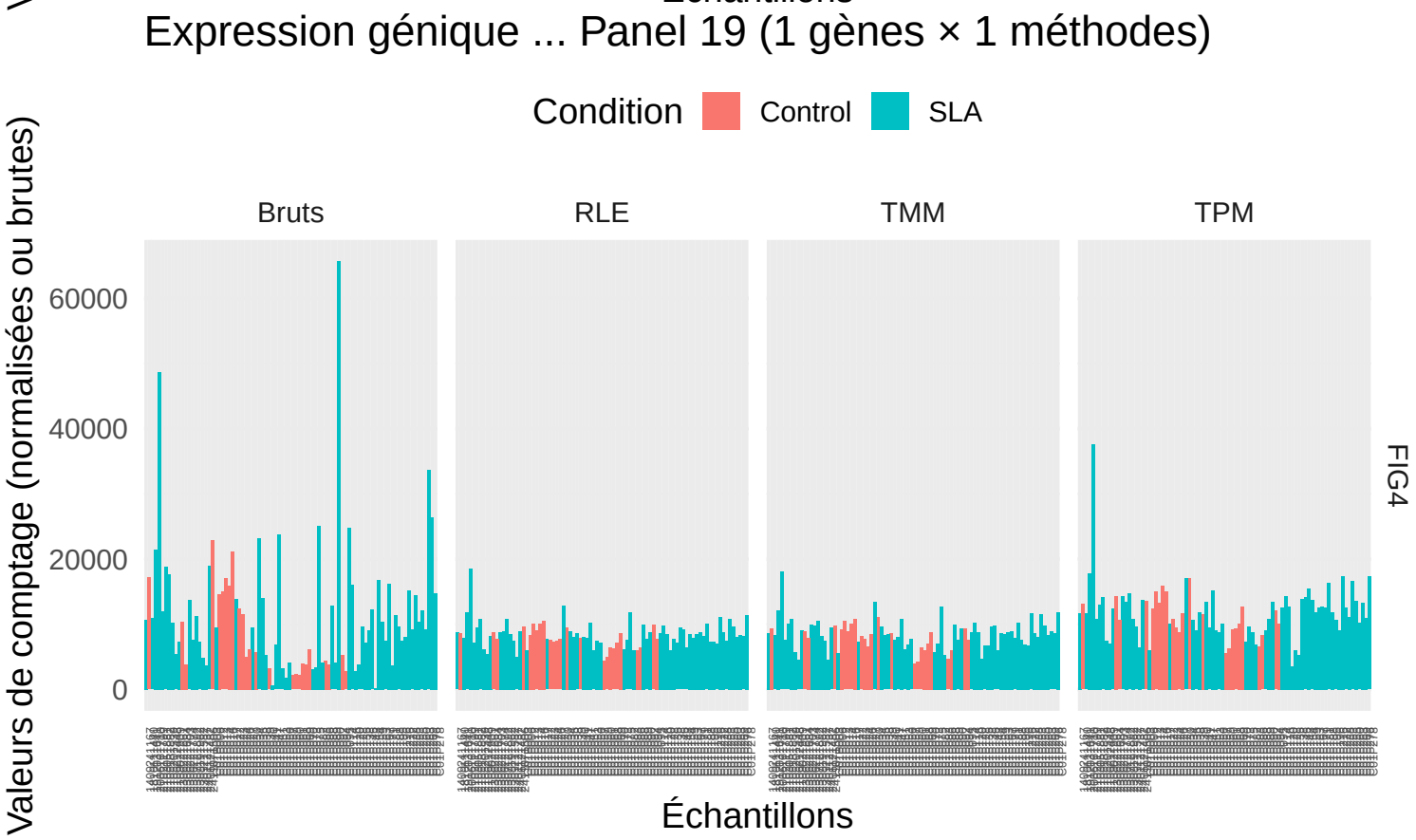

Expression génique ... Panel 14 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
150000
100000
50000
0
MATR3
Valeurs de comptage (normalisées ou brutes)
Échantillons

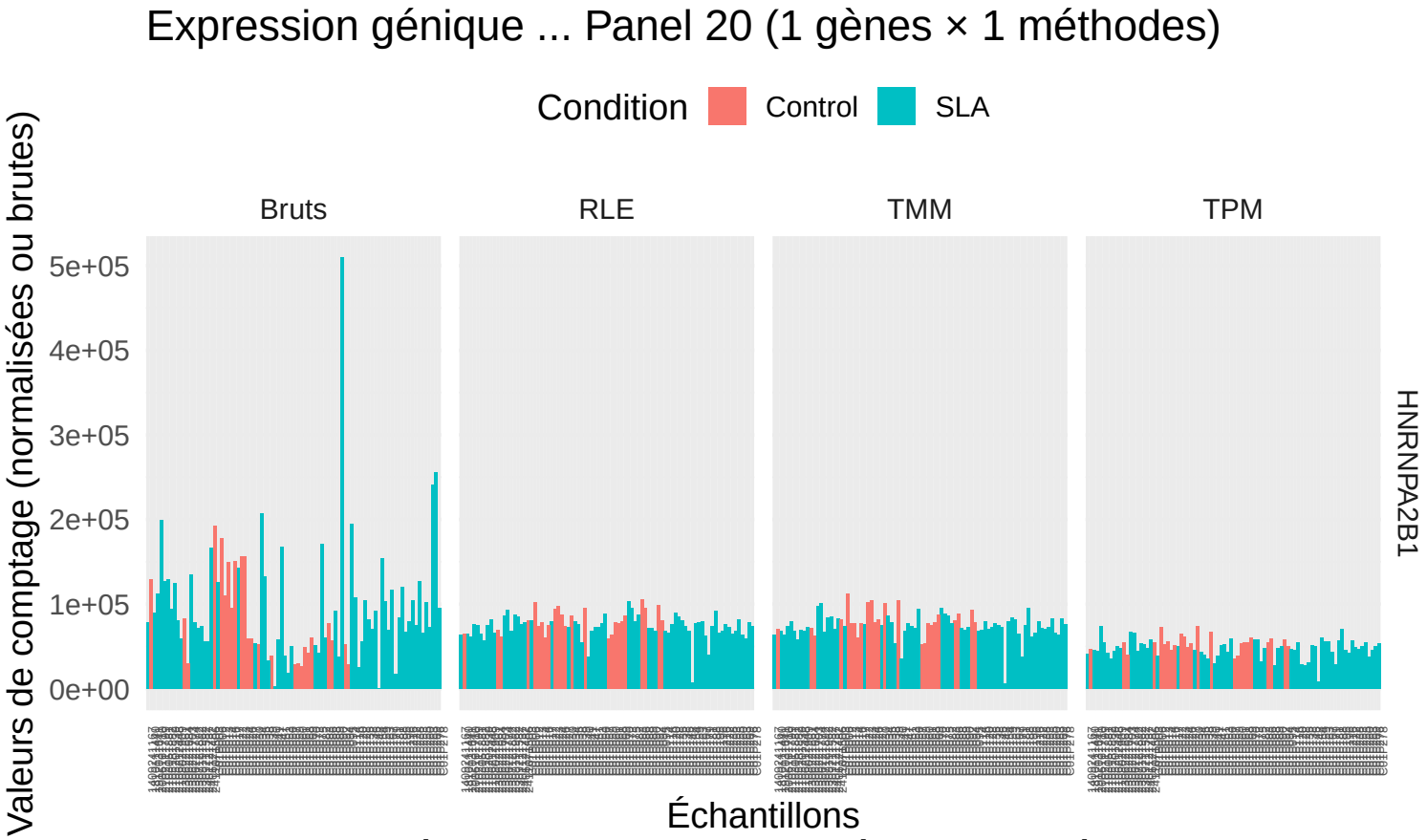

Expression génique ... Panel 15 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
10000
5000
0
DPYSL3
Valeurs de comptage (normalisées ou brutes)
Échantillons

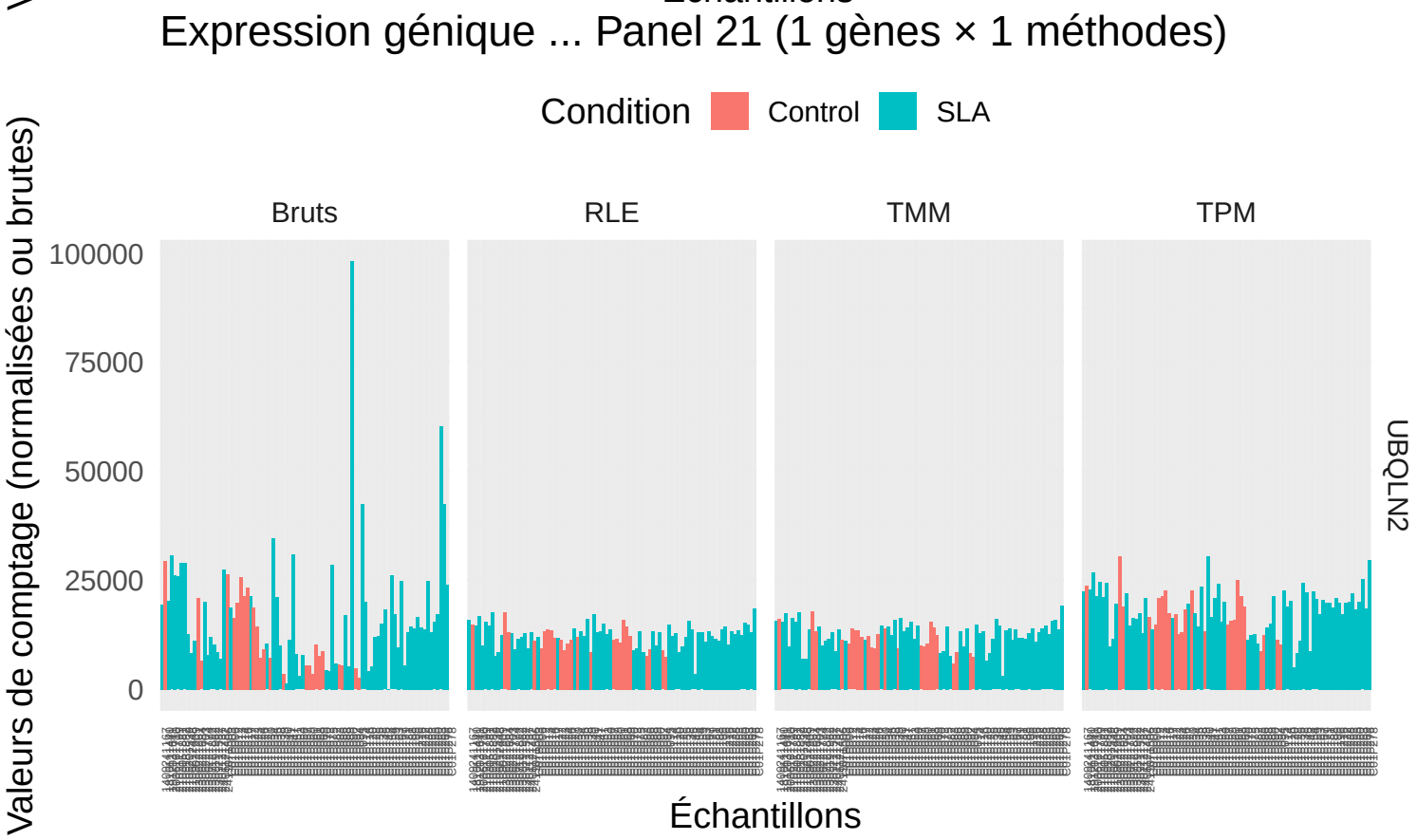

Expression génique ... Panel 16 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
2e+05
1e+05
0e+00
CSF1R
Échantillons

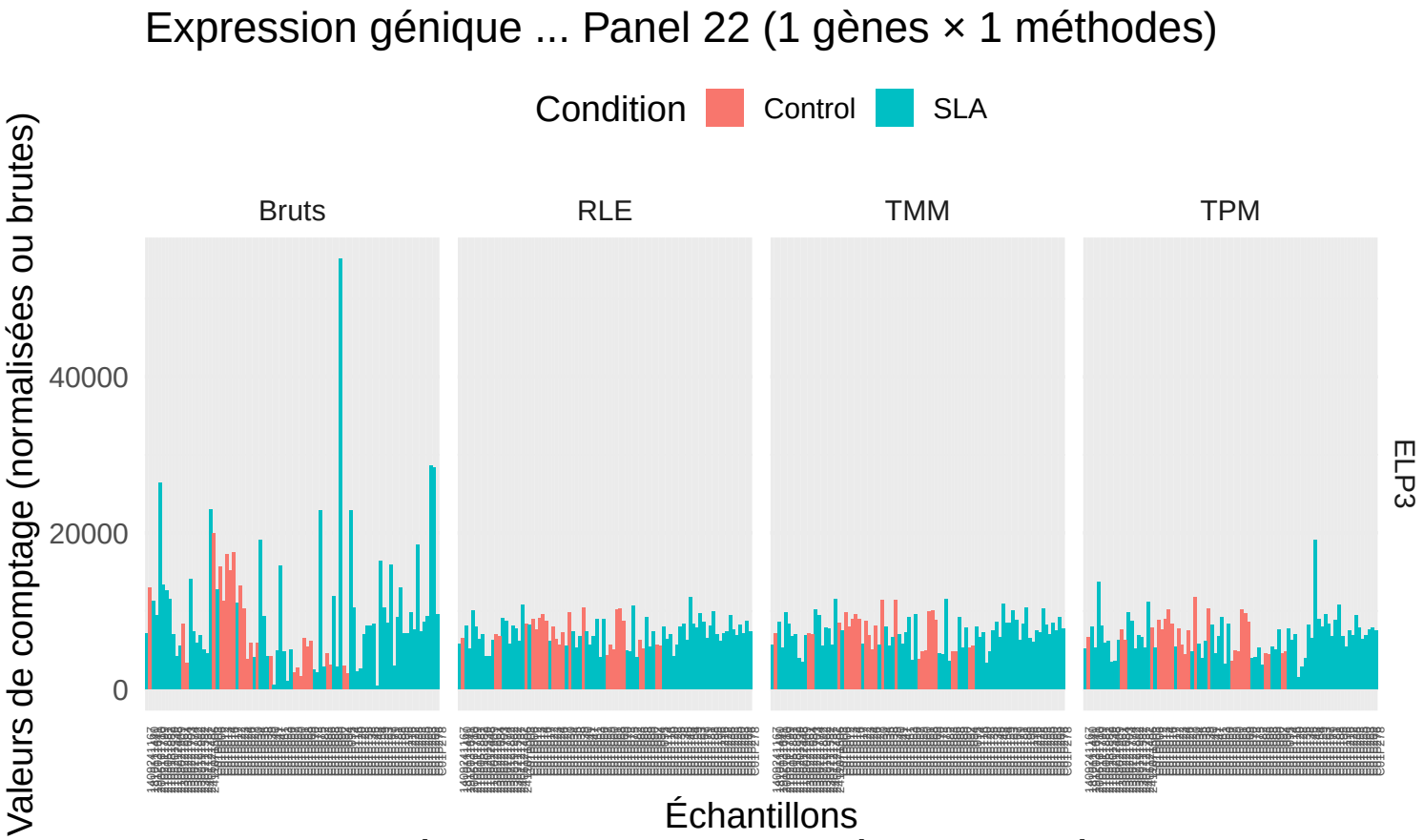

Expression génique ... Panel 17 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
2e+05
1e+05
0e+00
SQSTM1
Échantillons

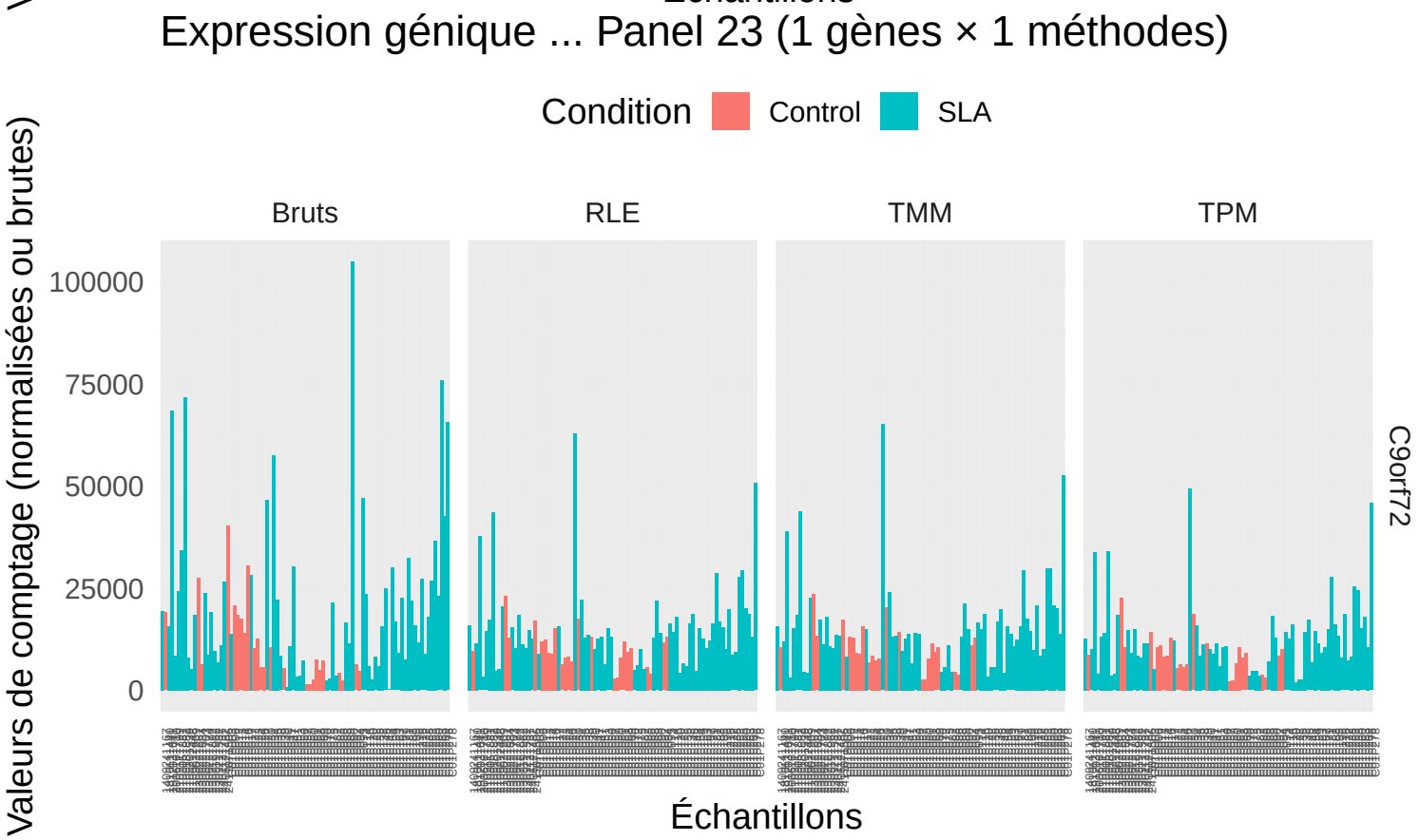

Expression génique ... Panel 18 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
30000
20000
10000
0
Valeurs de comptage (normalisées ou brutes)
TREM2
Échantillons

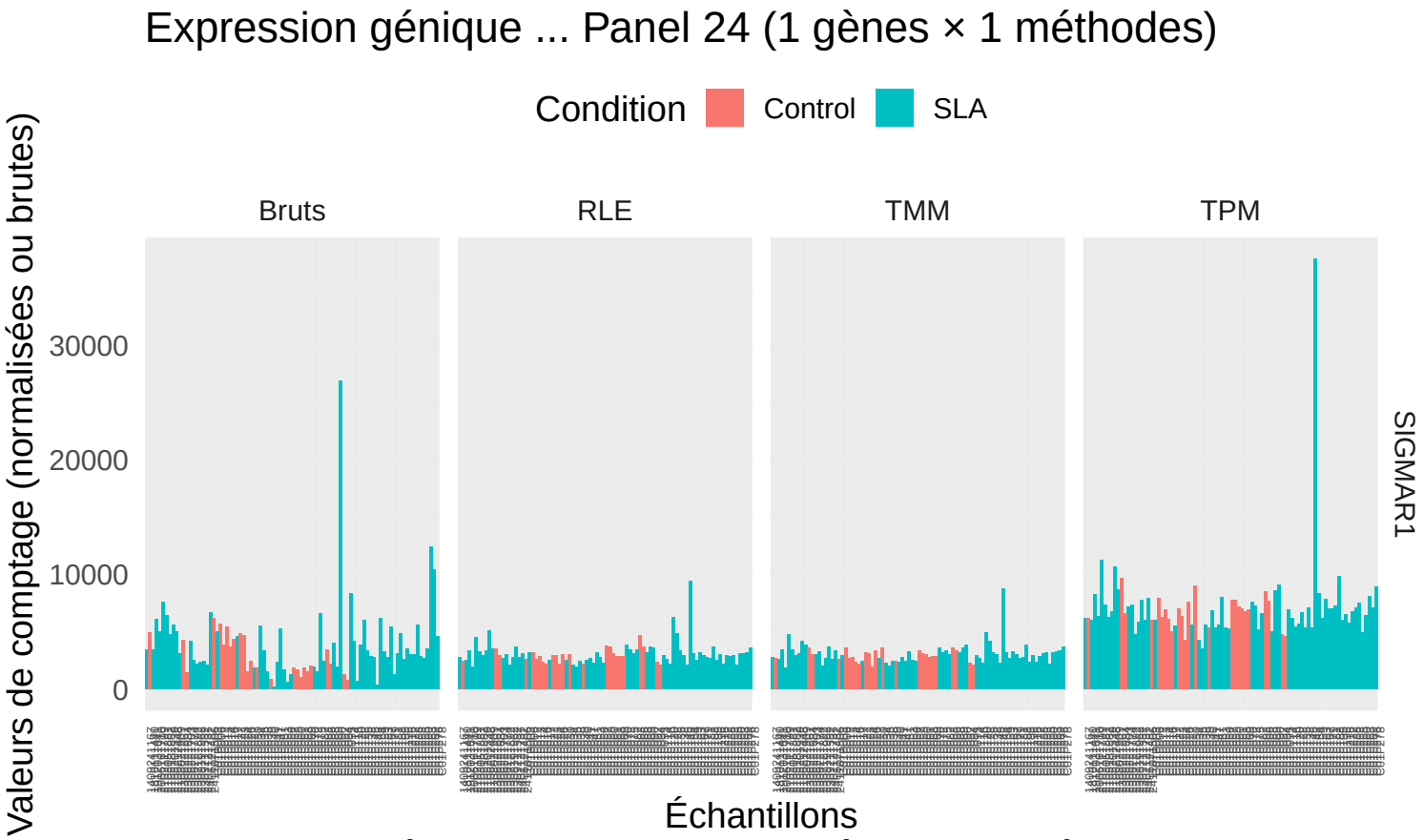

Expression génique ... Panel 19 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
60000
40000
20000
0
Valeurs de comptage (normalisées ou brutes)
FIG4
Échantillons

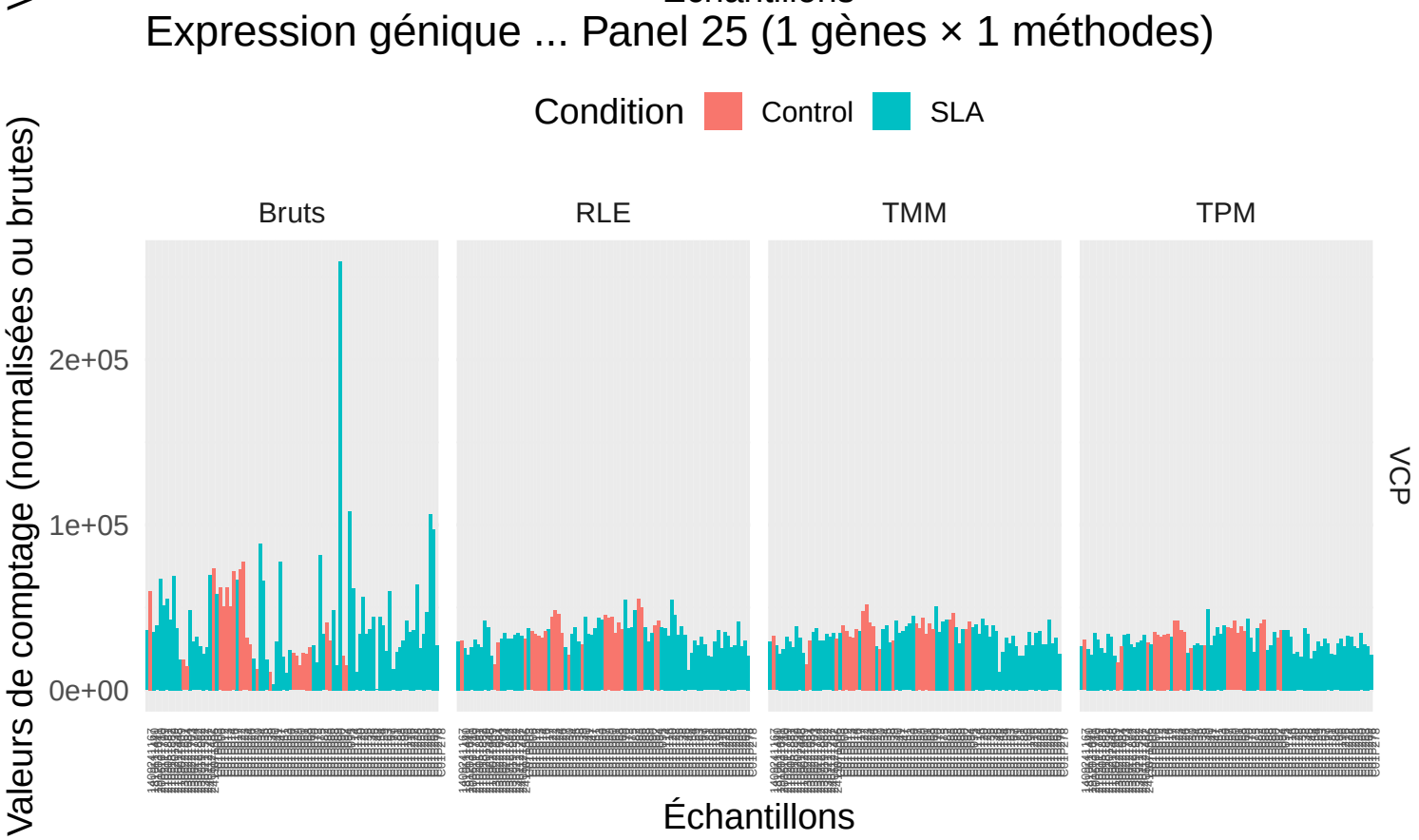

Expression génique ... Panel 20 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
5e+05
4e+05
3e+05
2e+05
1e+05
0e+00
Valeurs de comptage (normalisées ou brutes)
HNRNPA2B1
Échantillons

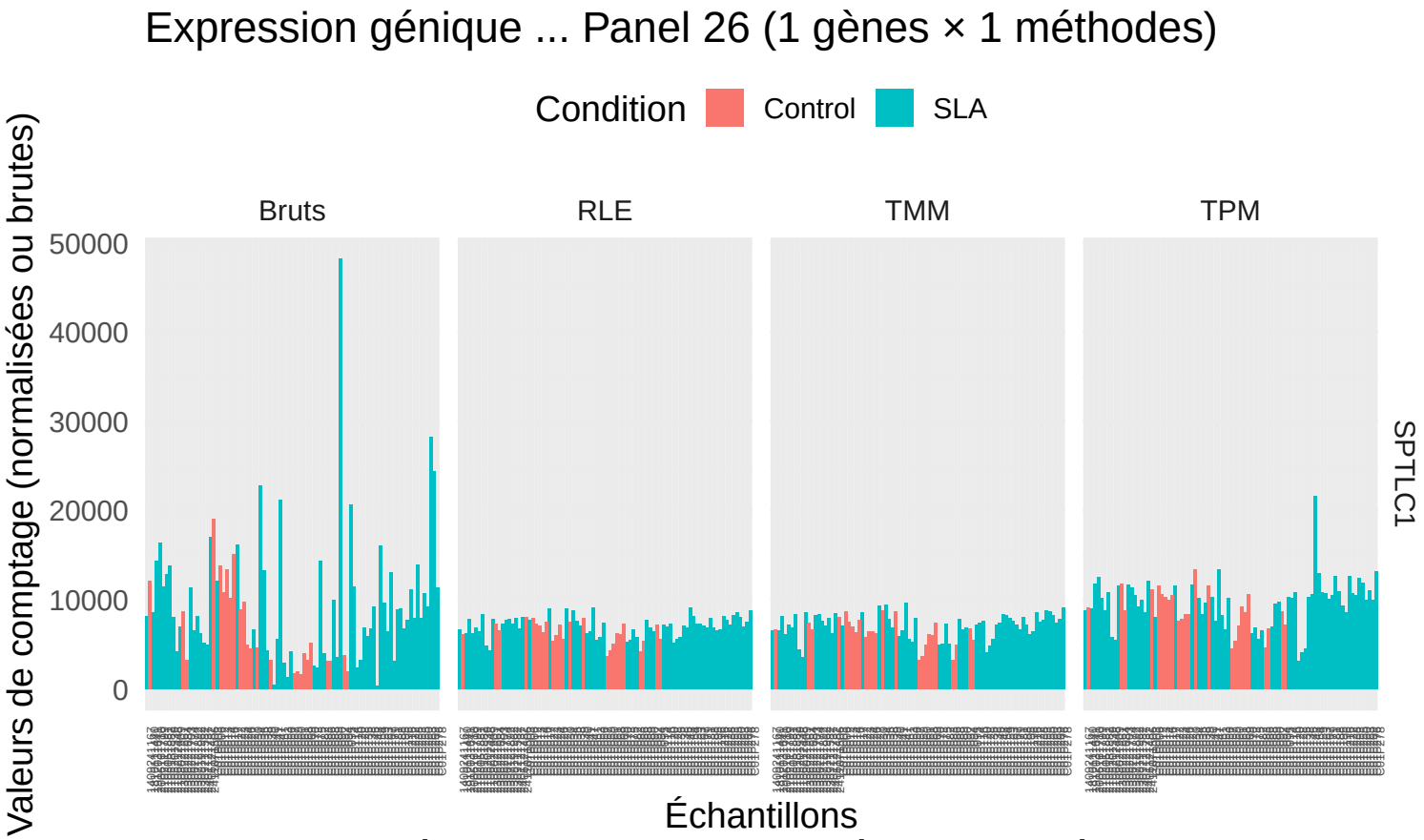

Expression génique ... Panel 21 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
100000
75000
50000
25000
0
Valeurs de comptage (normalisées ou brutes)
UBQLN2
Échantillons

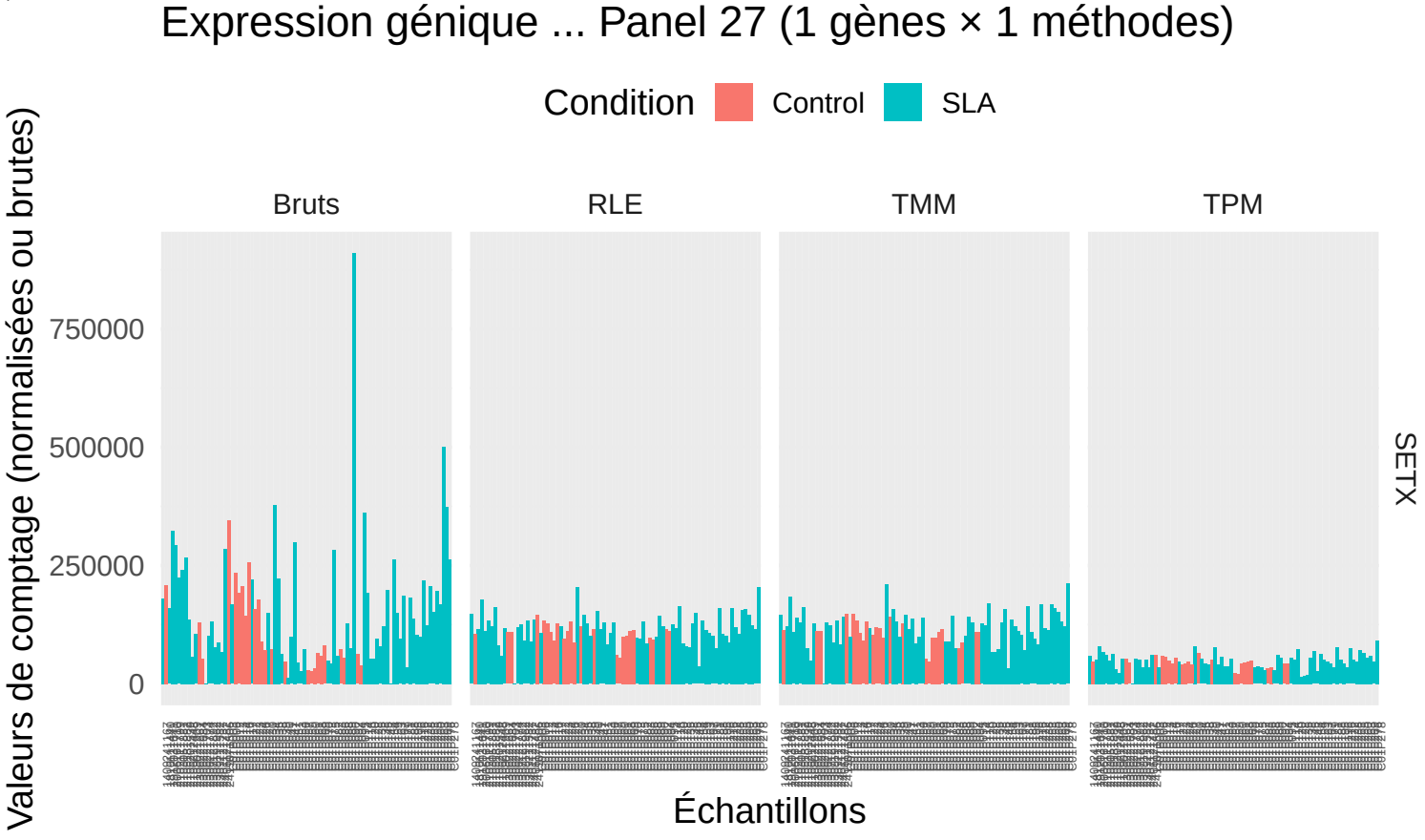

Expression génique ... Panel 22 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
40000
20000
0
ELP3
Valeurs de comptage (normalisées ou brutes)
Échantillons

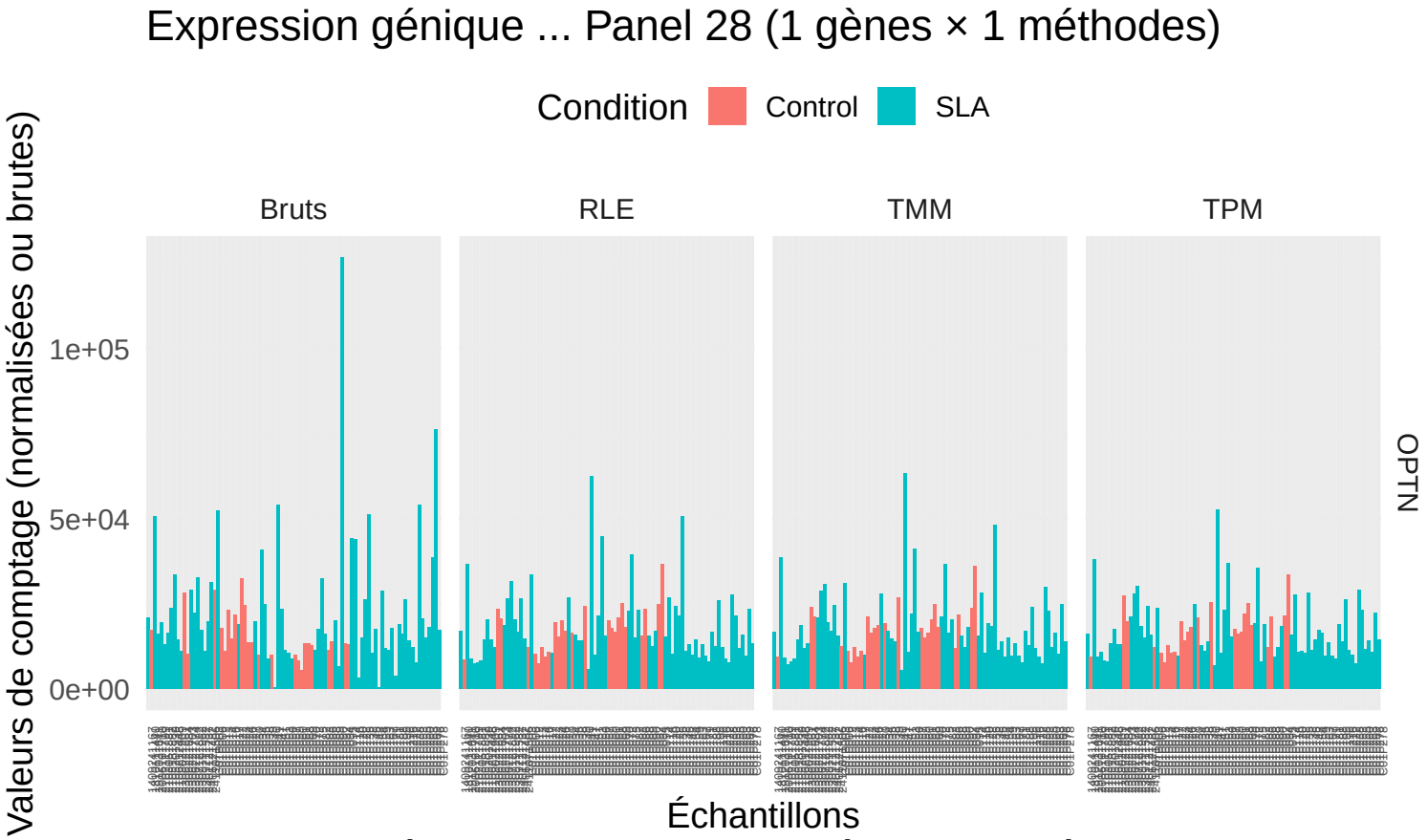

Expression génique ... Panel 23 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
100000
75000
50000
25000
0
C9orf72
Valeurs de comptage (normalisées ou brutes)
Échantillons

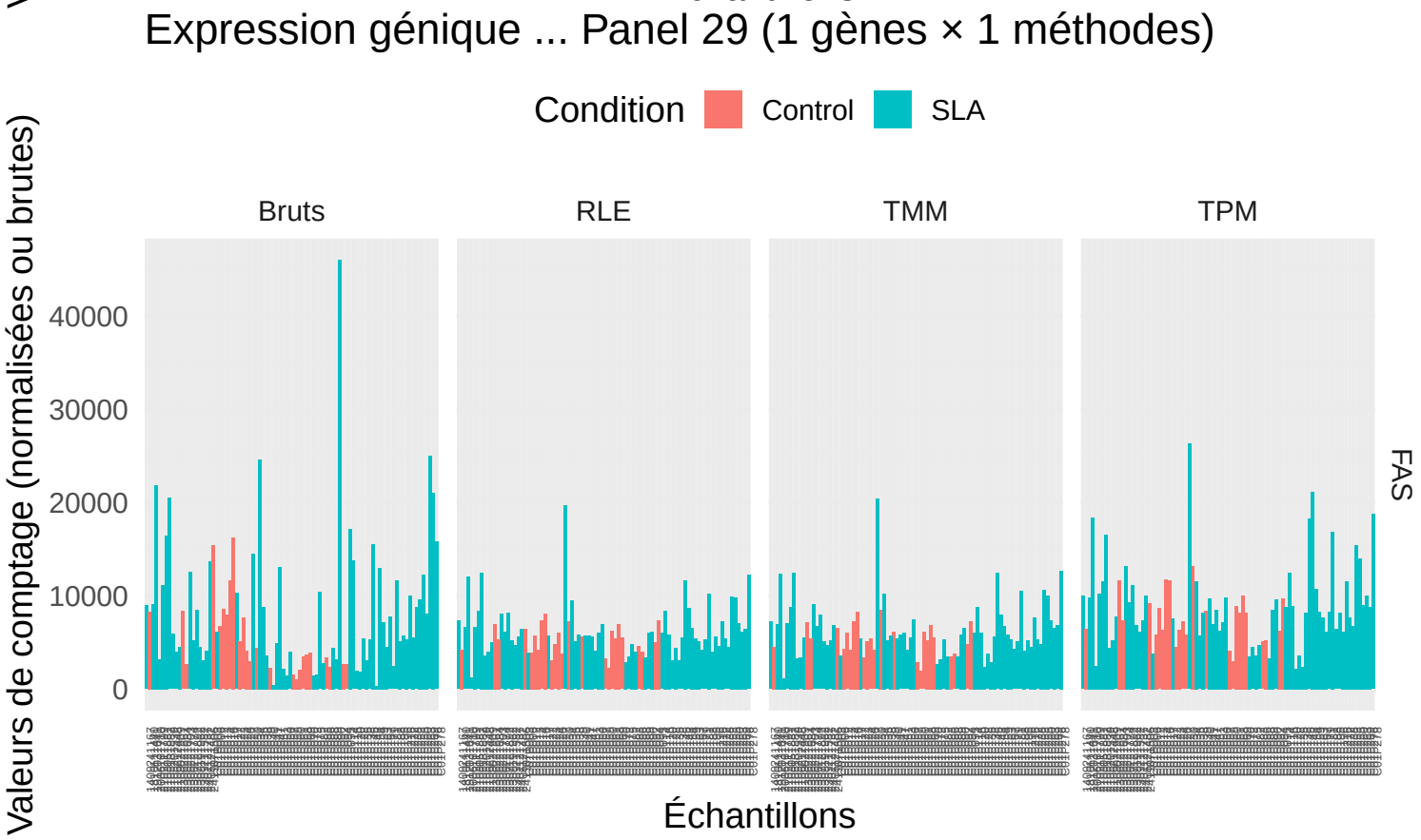

Expression génique ... Panel 24 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
30000
20000
10000
0
Valeurs de comptage (normalisées ou brutes)
SIGMAR1
Échantillons

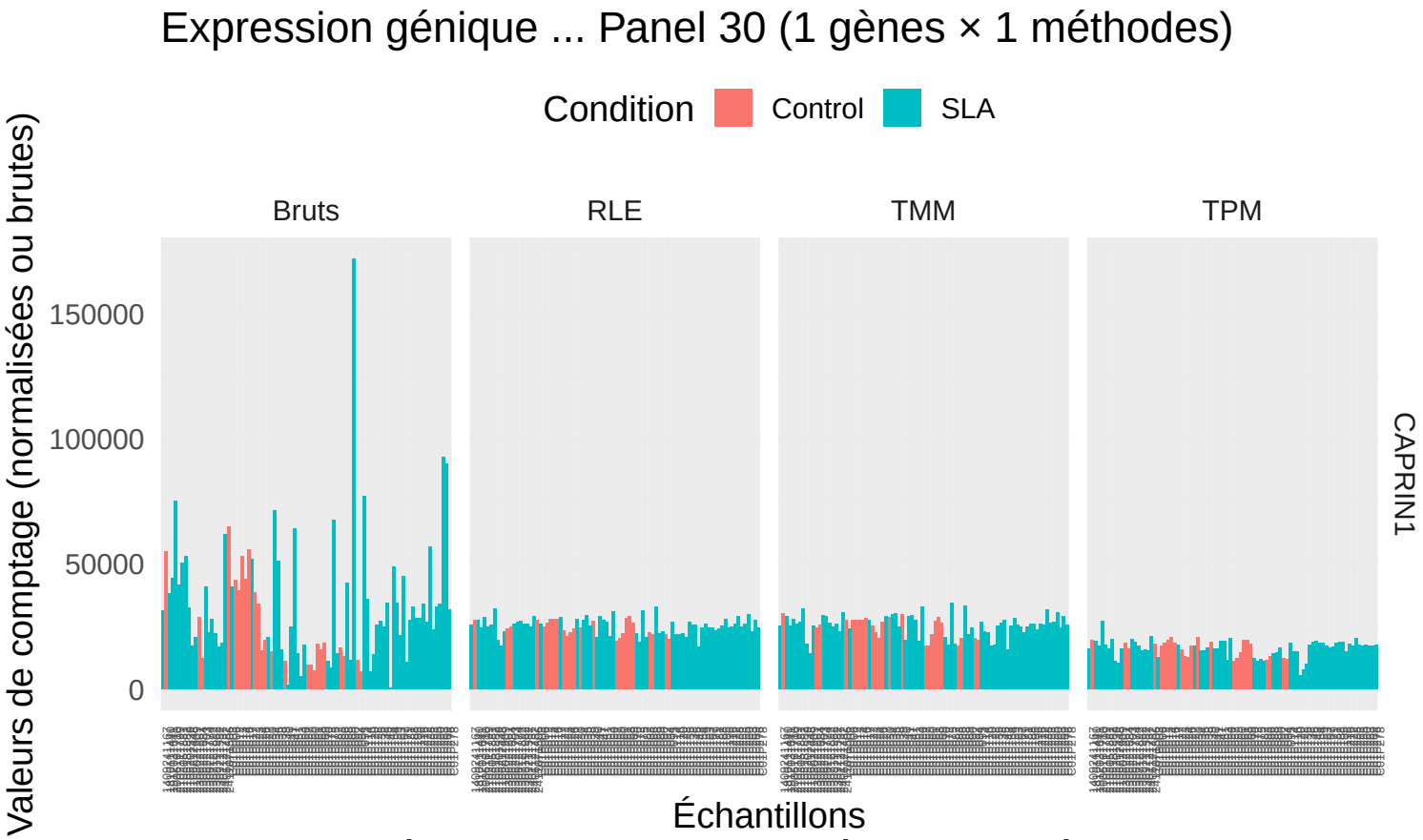

Expression génique ... Panel 25 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
2e+05
1e+05
0e+00
Valeurs de comptage (normalisées ou brutes)
VCP
Échantillons

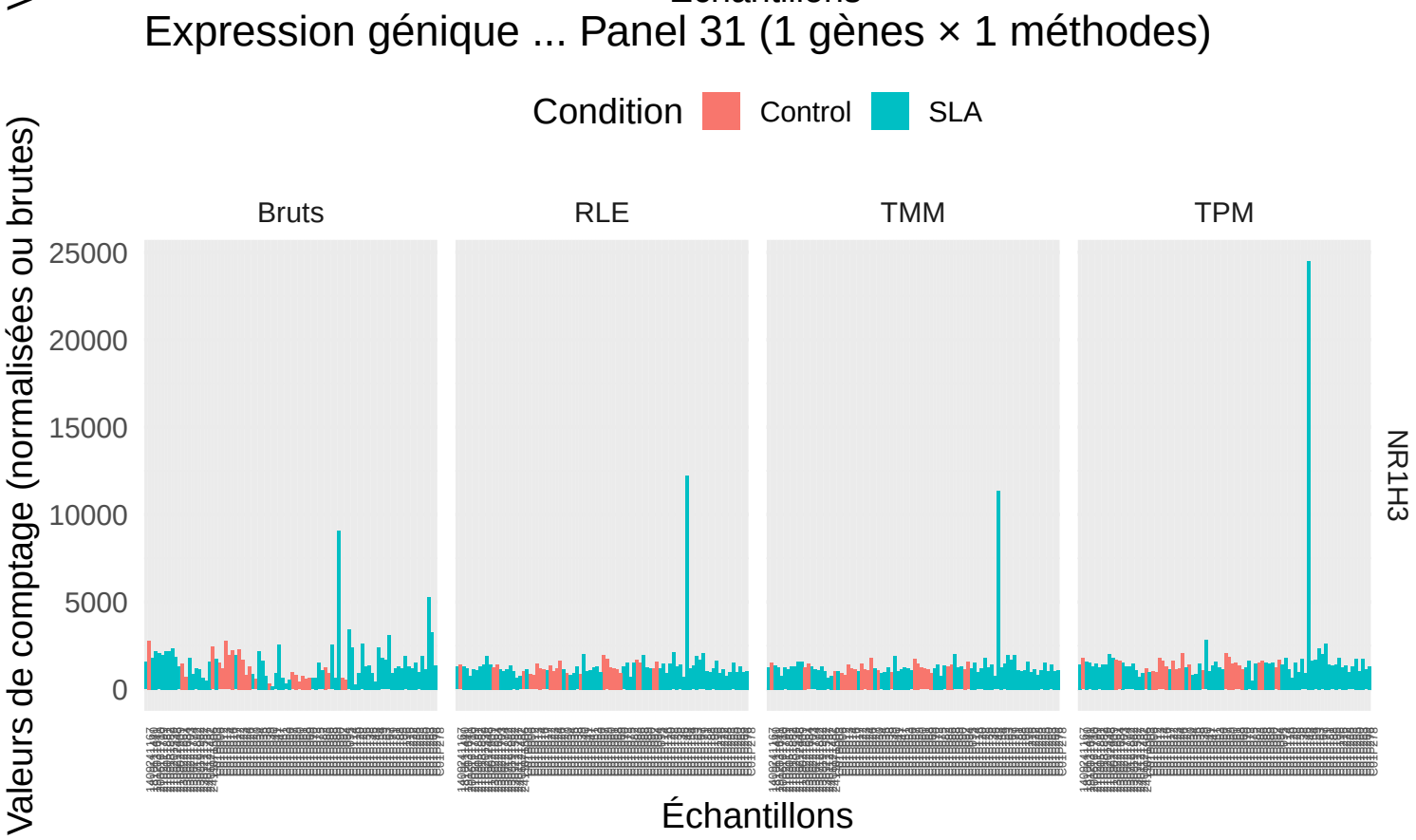

Expression génique ... Panel 26 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
50000
40000
30000
20000
10000
0
Valeurs de comptage (normalisées ou brutes)
SPTLC1
Échantillons

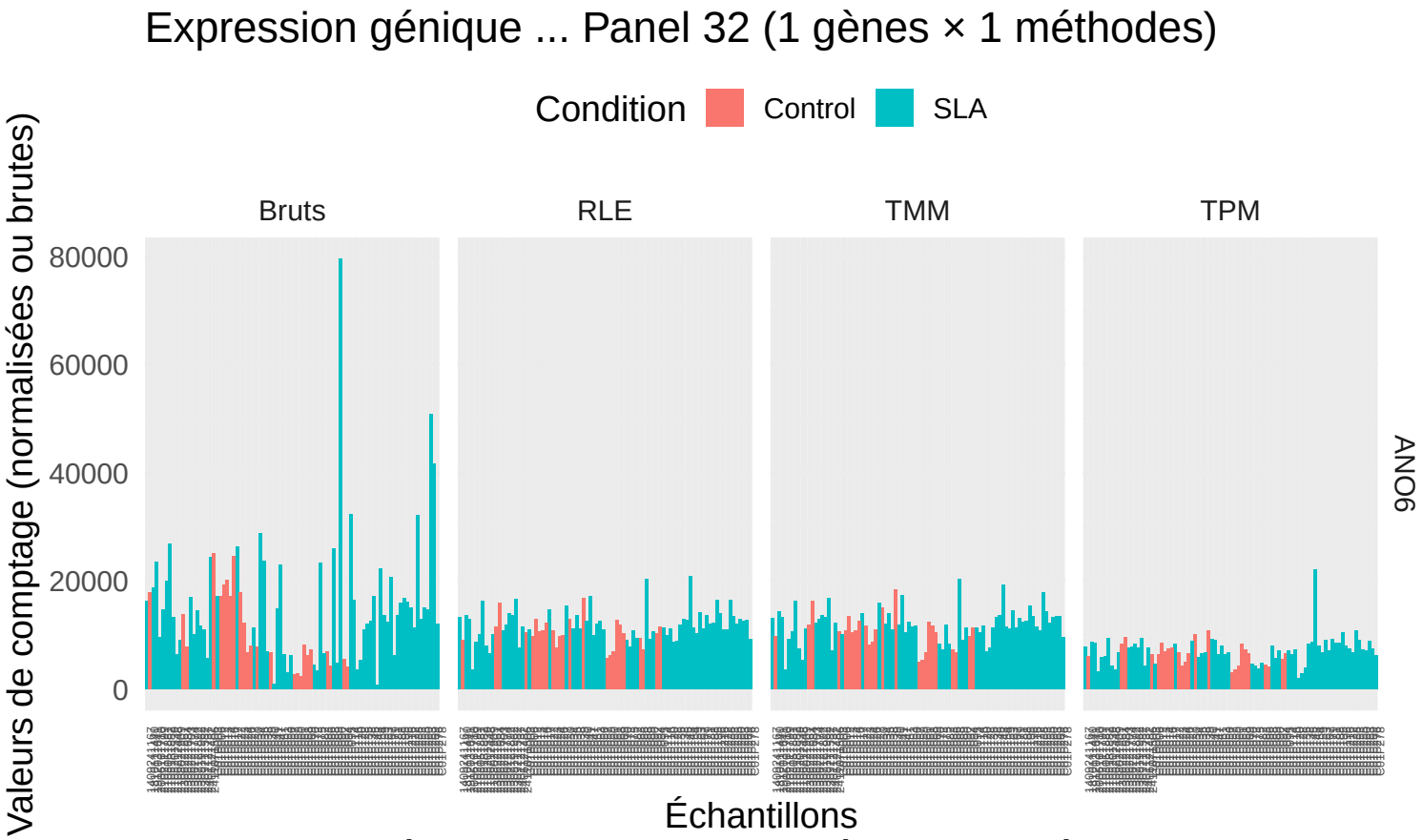

Expression génique ... Panel 27 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
750000
500000
250000
0
Valeurs de comptage (normalisées ou brutes)
SETX
Échantillons

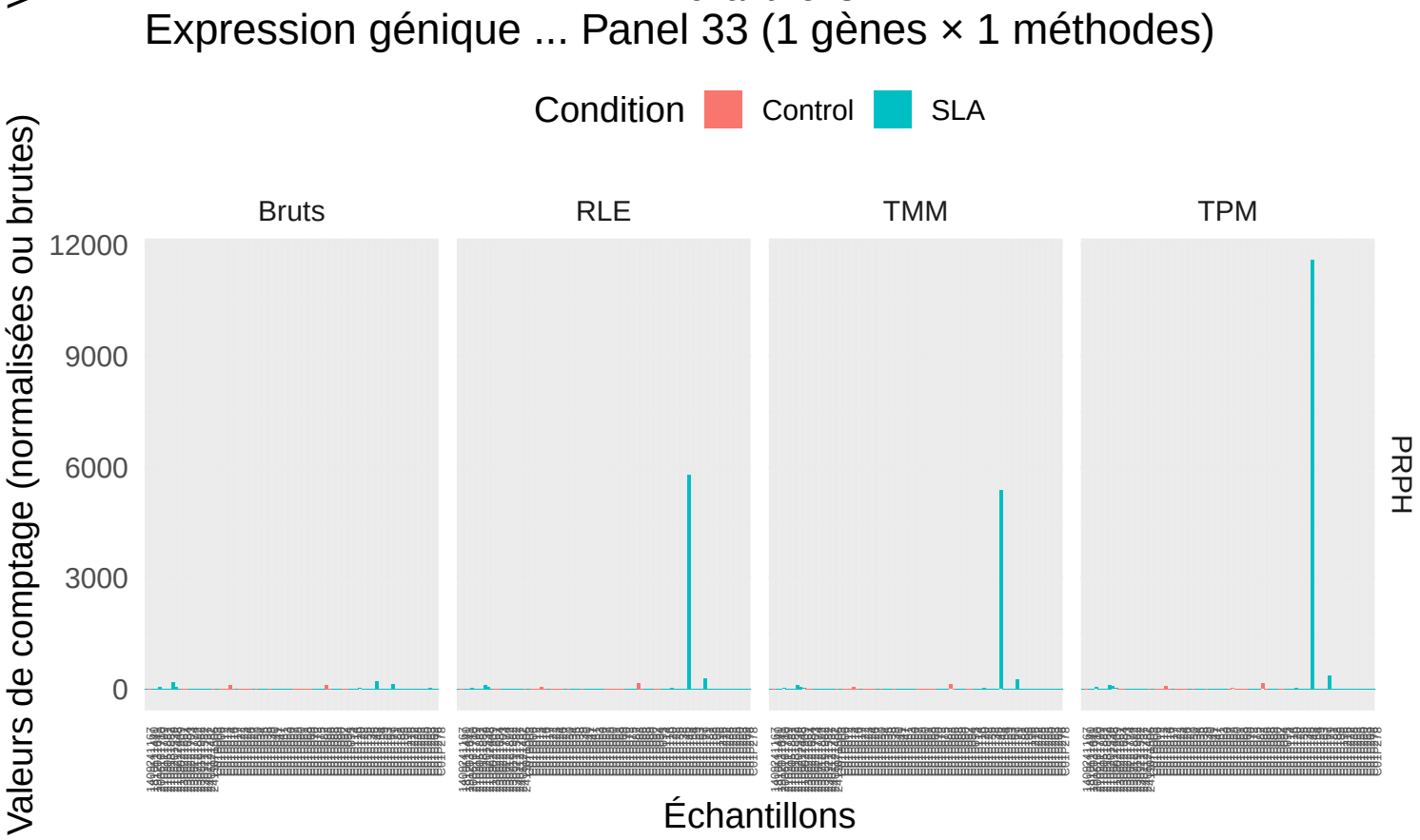

Expression génique ... Panel 28 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
1e+05
5e+04
0e+00
OPTN
Échantillons

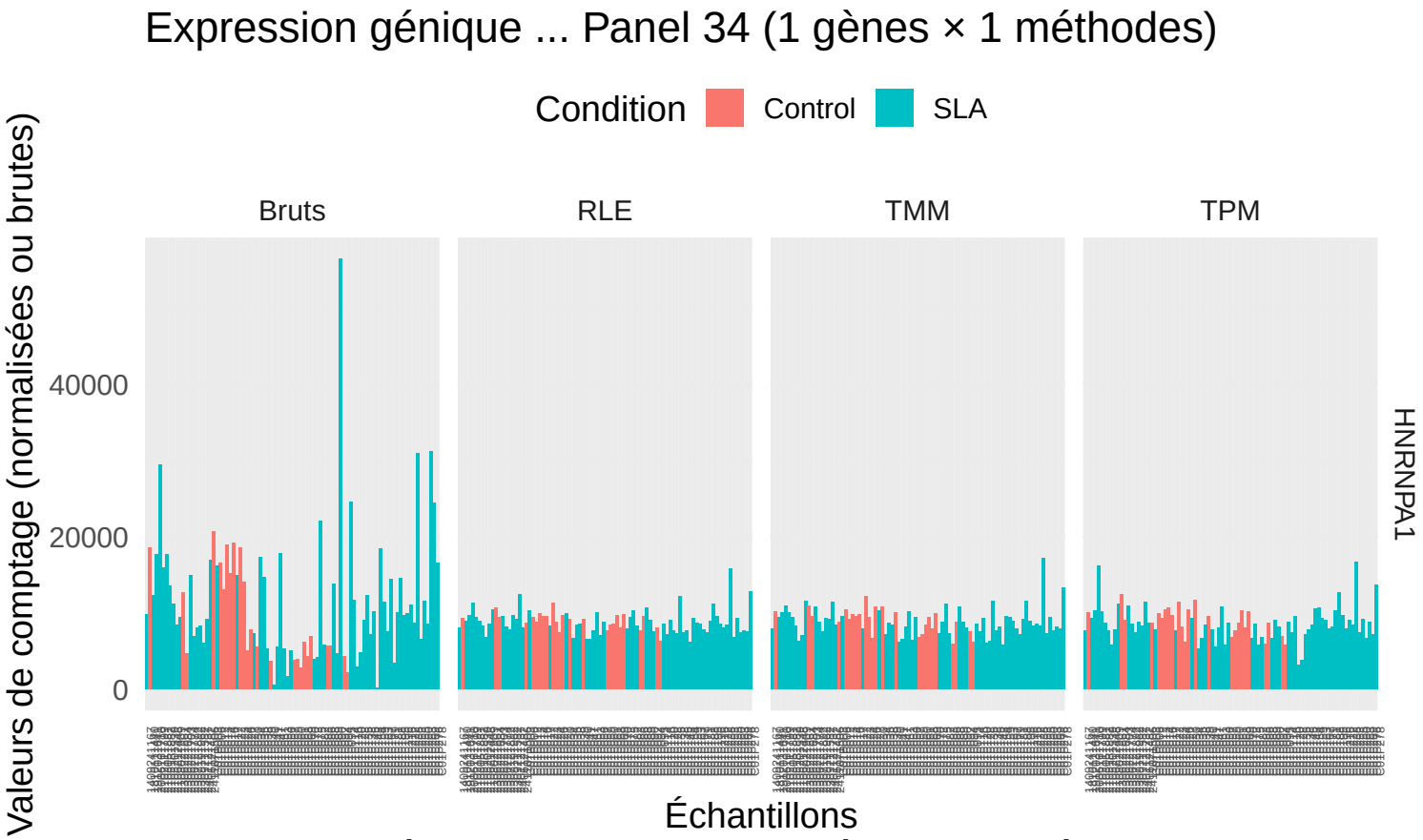

Expression génique ... Panel 29 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
40000
30000
20000
10000
0
FAS
Échantillons

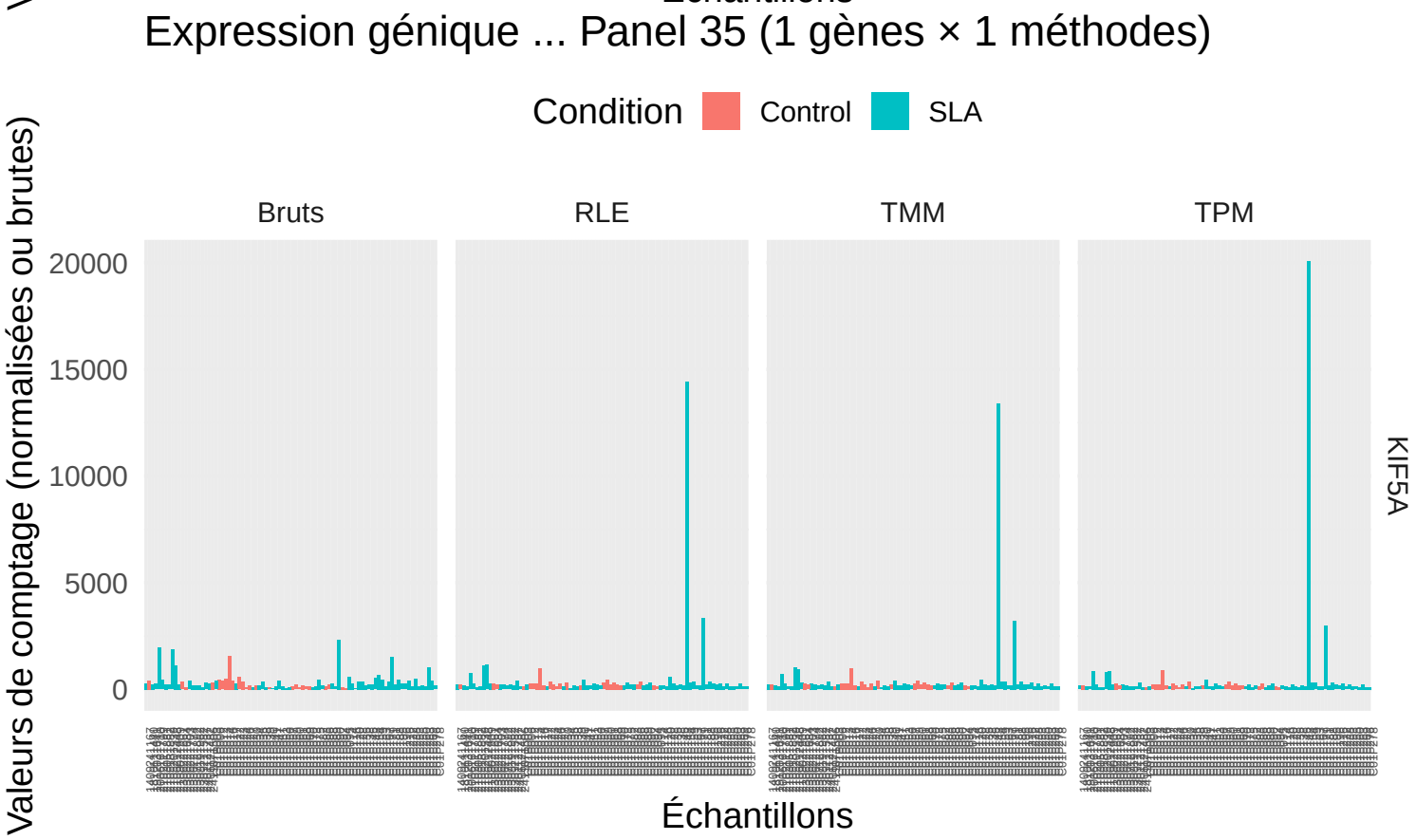

Expression génique ... Panel 30 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
150000
100000
50000
0
Valeurs de comptage (normalisées ou brutes)
CAPRIN1
Échantillons

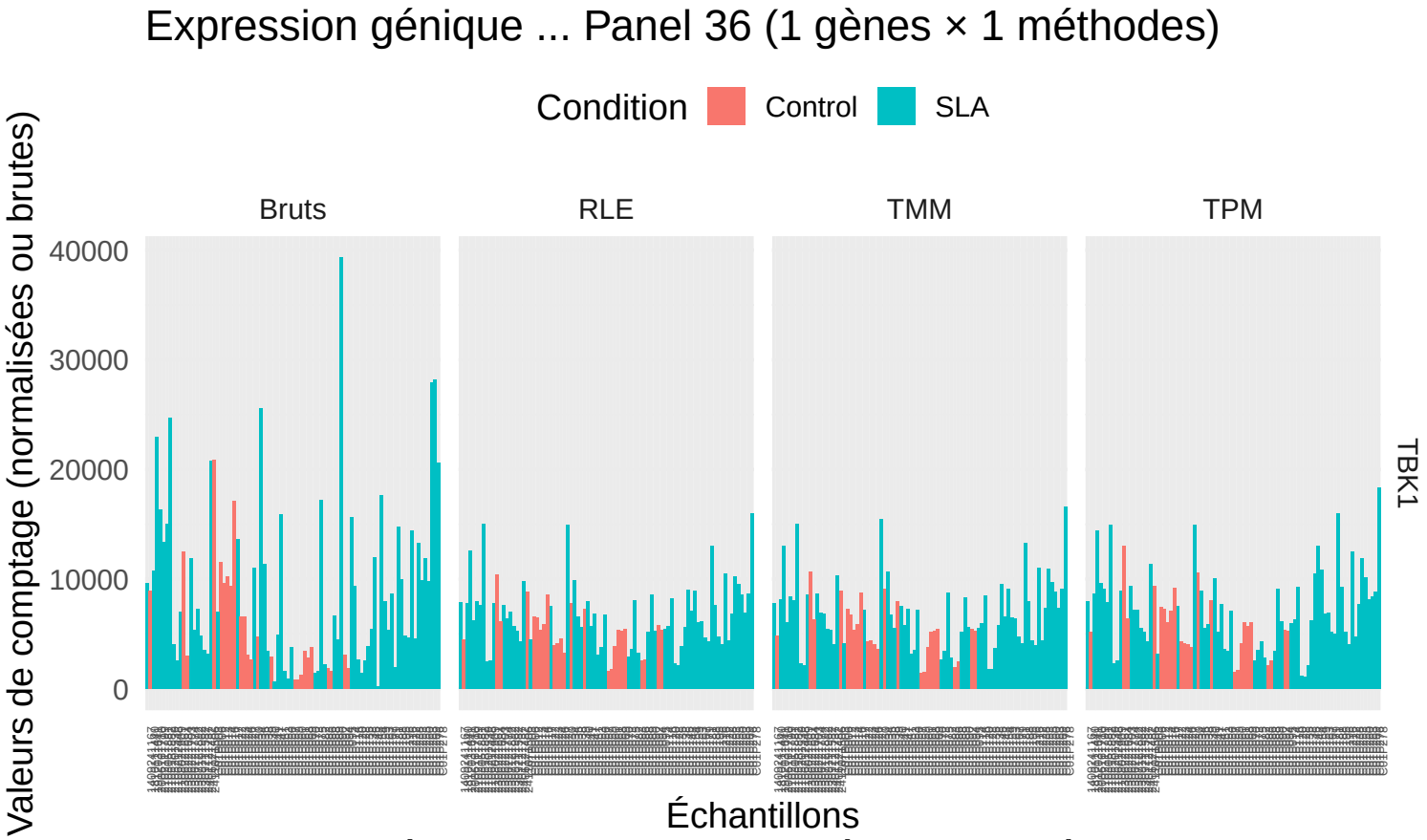

Expression génique ... Panel 31 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
25000
20000
15000
10000
5000
0
Valeurs de comptage (normalisées ou brutes)
NR1H3
Échantillons

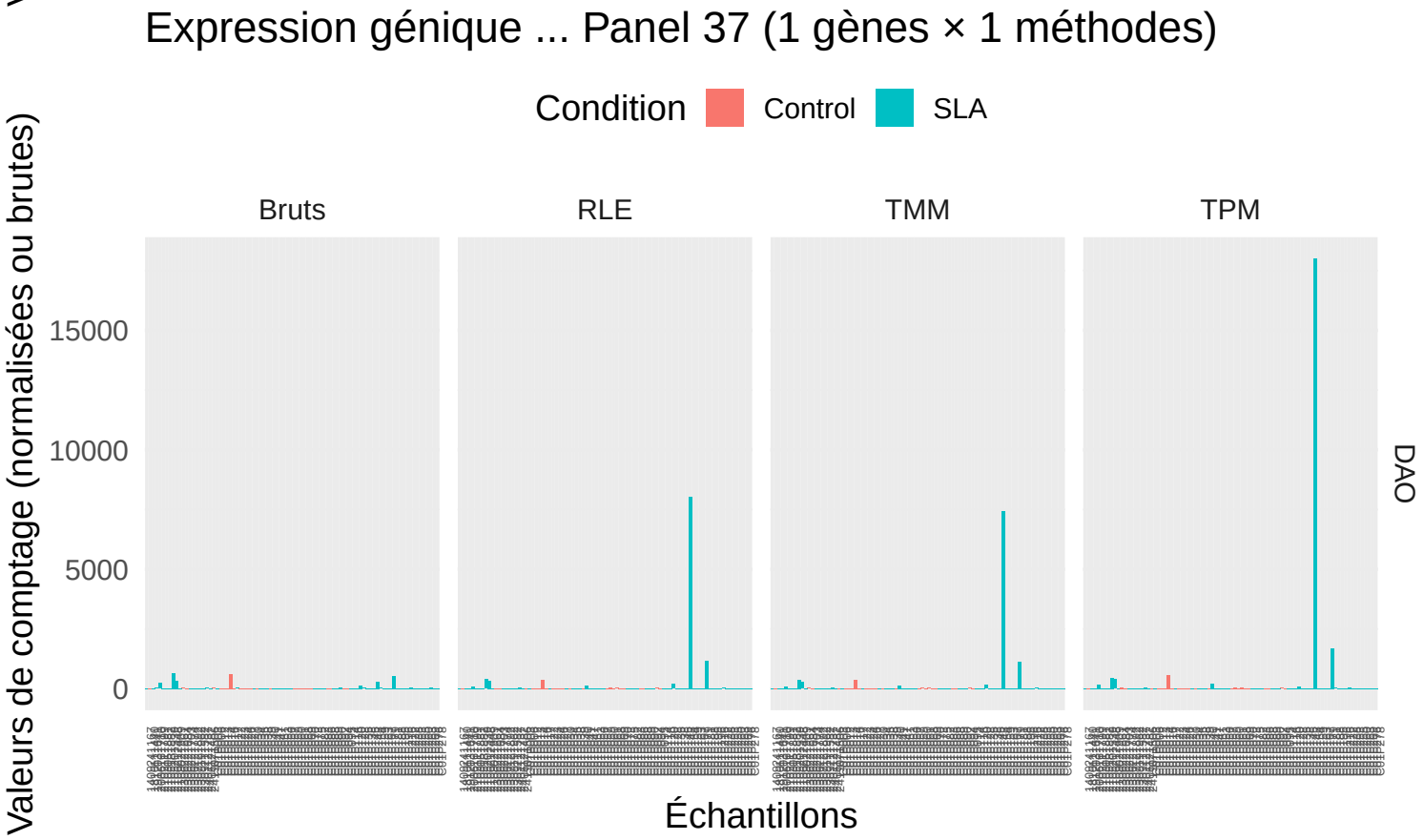

Expression génique ... Panel 32 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
80000
60000
40000
20000
0
ANO6
Valeurs de comptage (normalisées ou brutes)
Échantillons

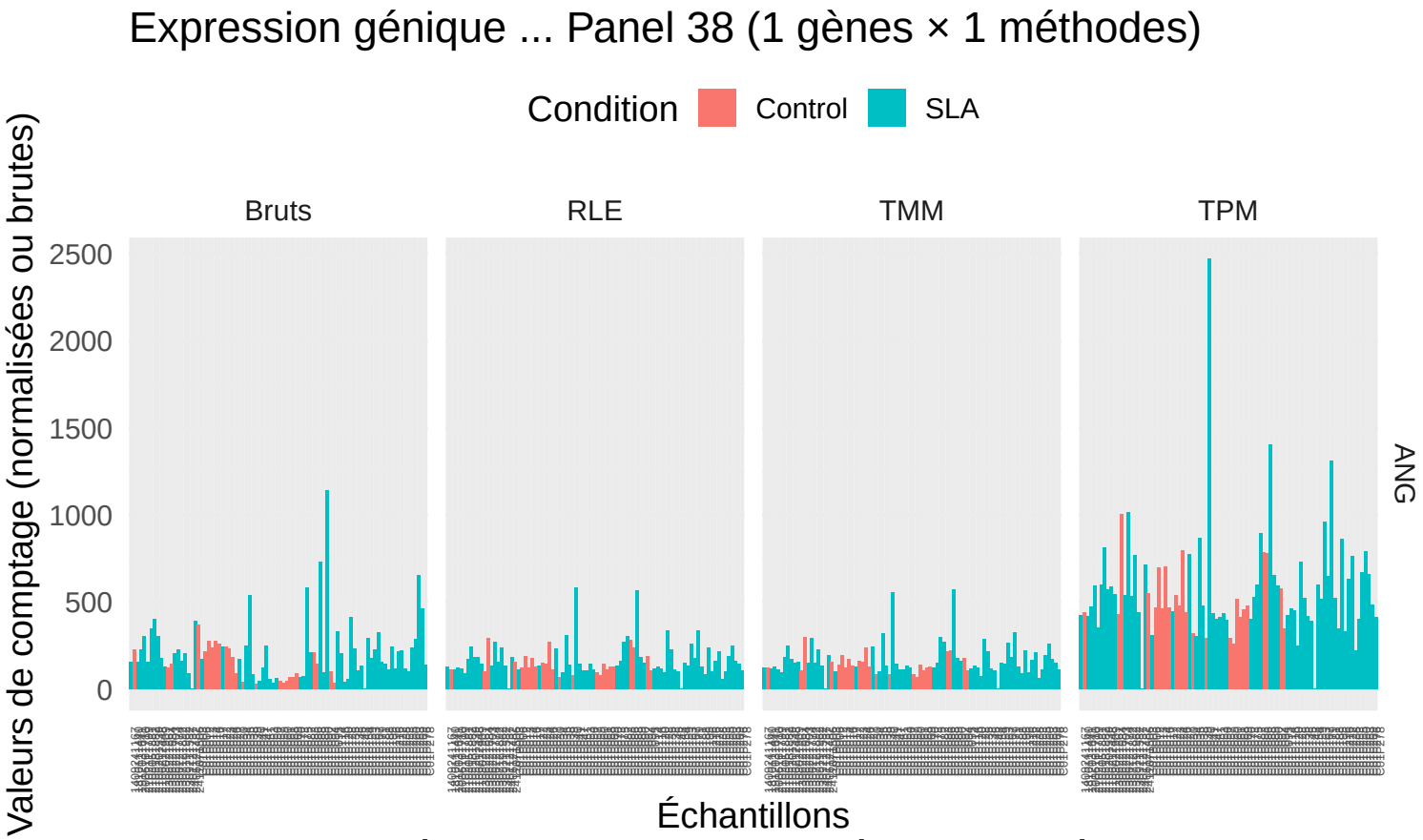

Expression génique ... Panel 33 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
12000
9000
6000
3000
0
PRPH
Valeurs de comptage (normalisées ou brutes)
Échantillons

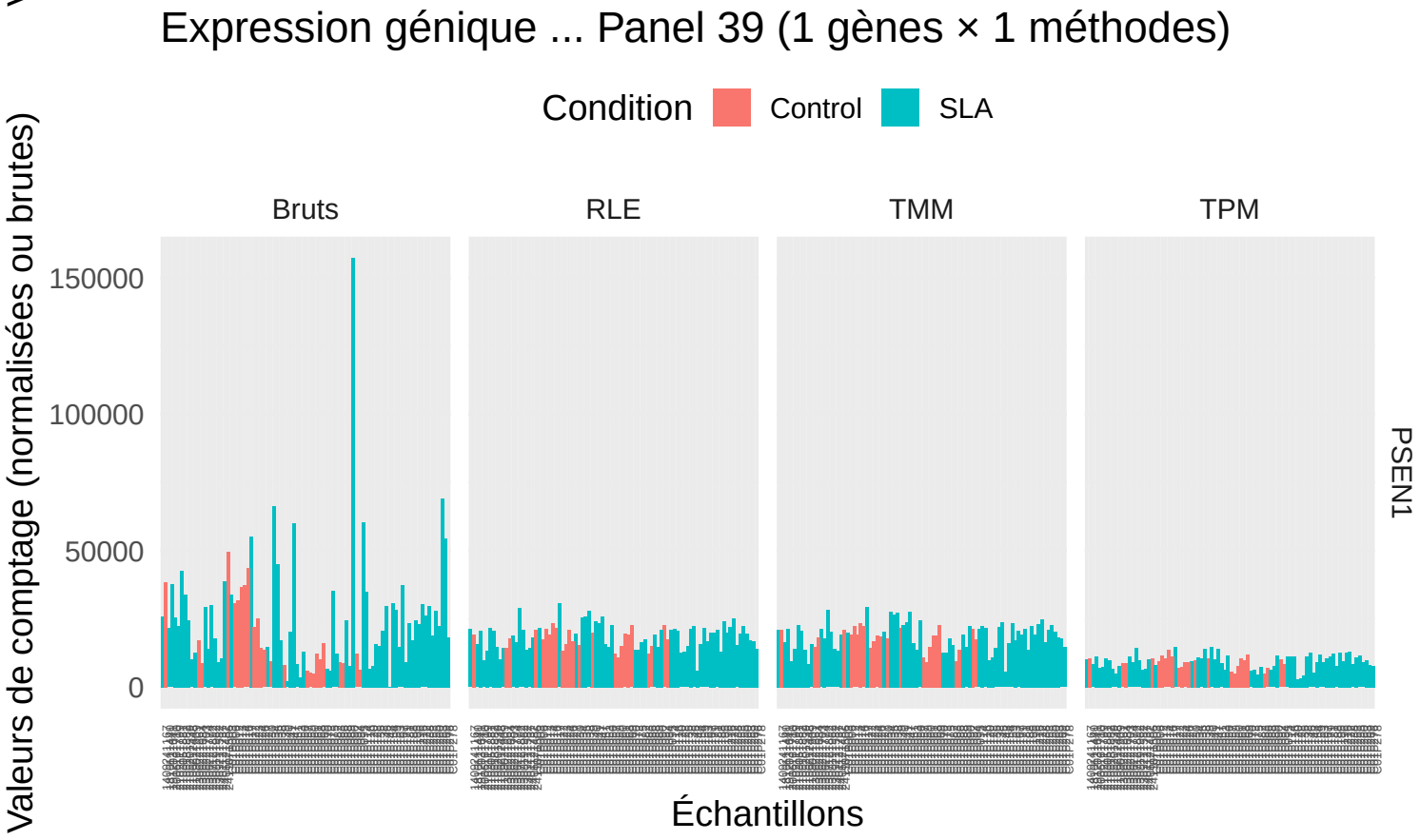

Expression génique ... Panel 34 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
40000
20000
0
HNRNPA1
Échantillons

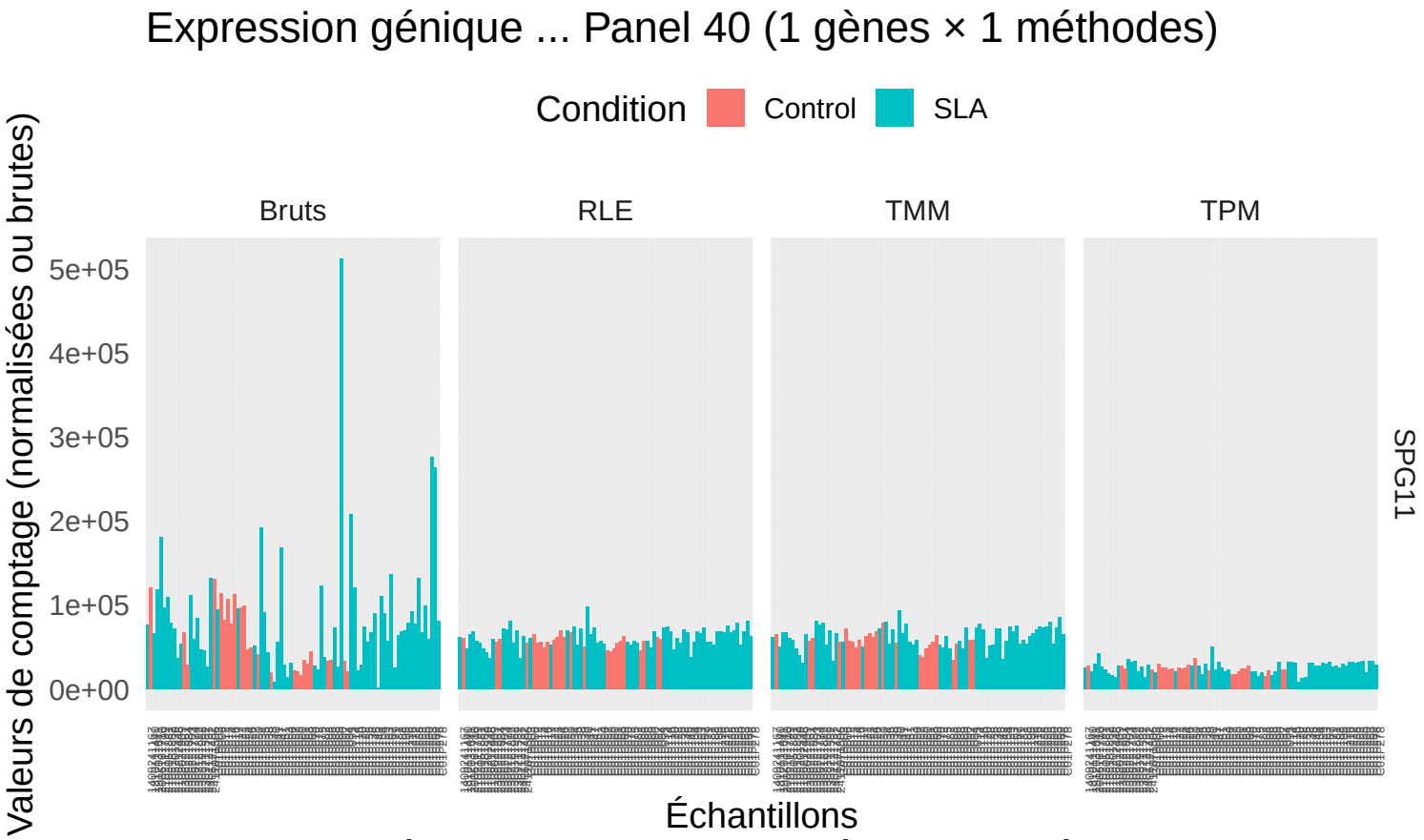

Expression génique ... Panel 35 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
20000
15000
10000
5000
0
KIF5A
Échantillons

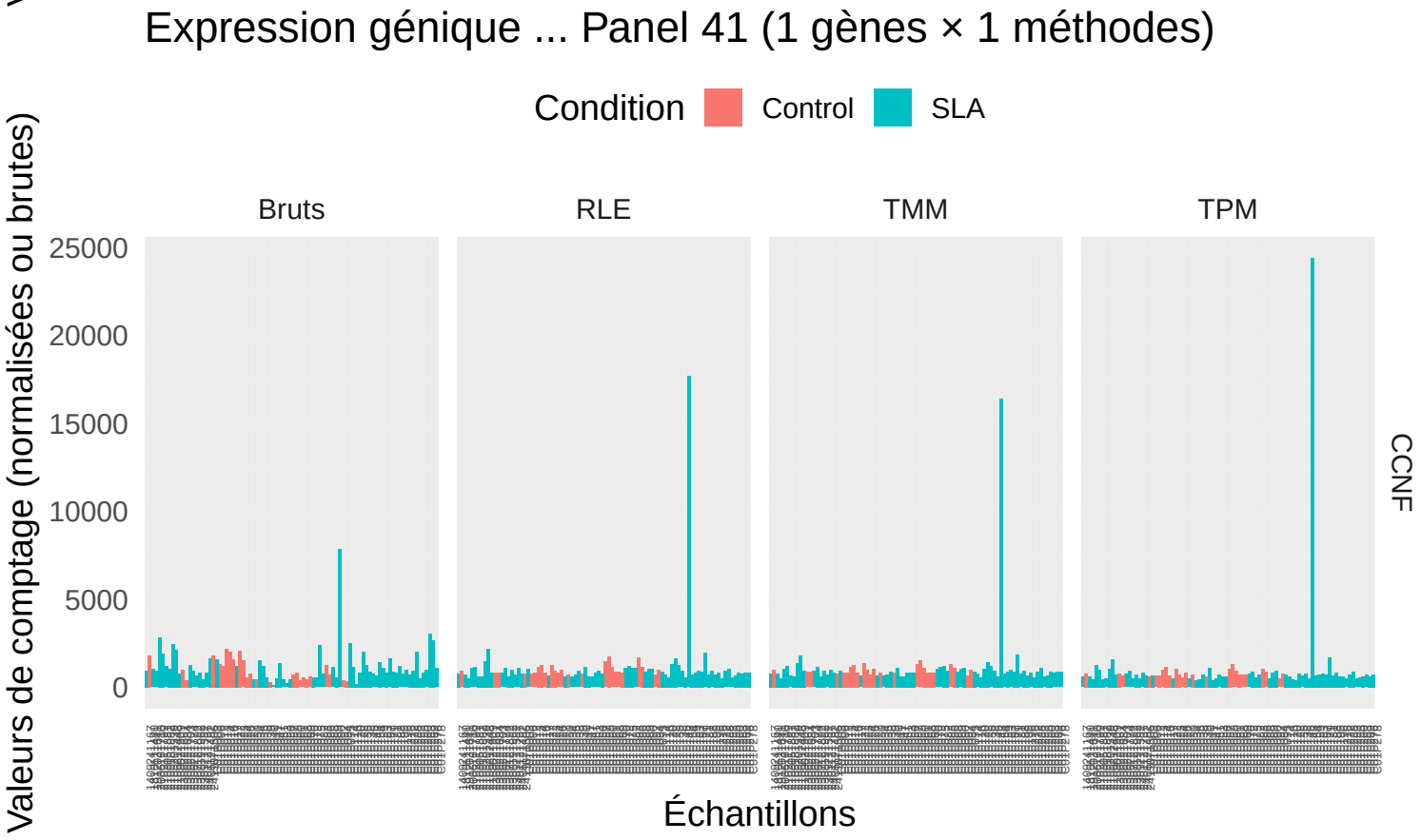

Expression génique ... Panel 36 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
40000
30000
20000
10000
0
Valeurs de comptage (normalisées ou brutes)
TBK1
Échantillons

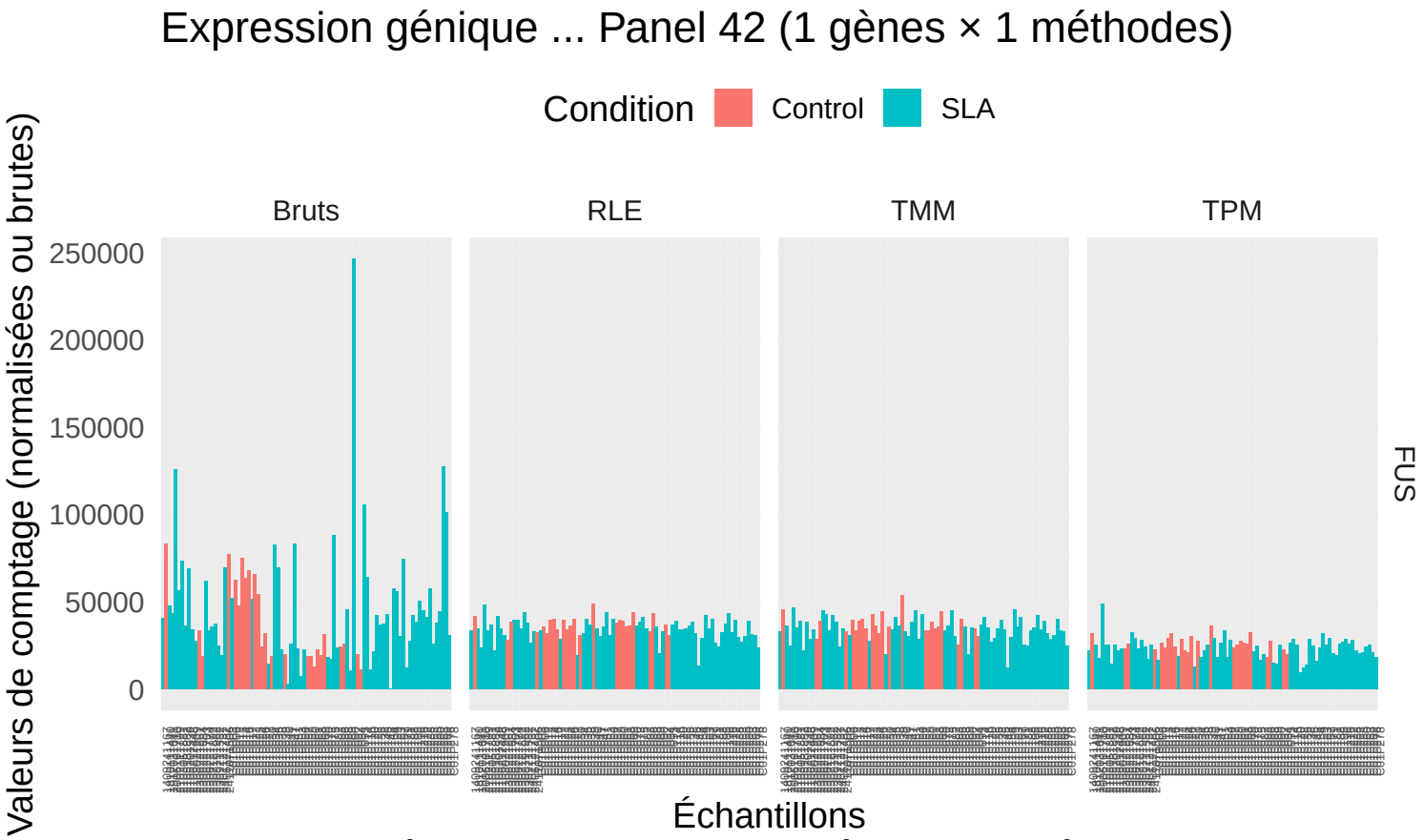

Expression génique ... Panel 37 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
15000
10000
5000
0
Valeurs de comptage (normalisées ou brutes)
DAO
Échantillons

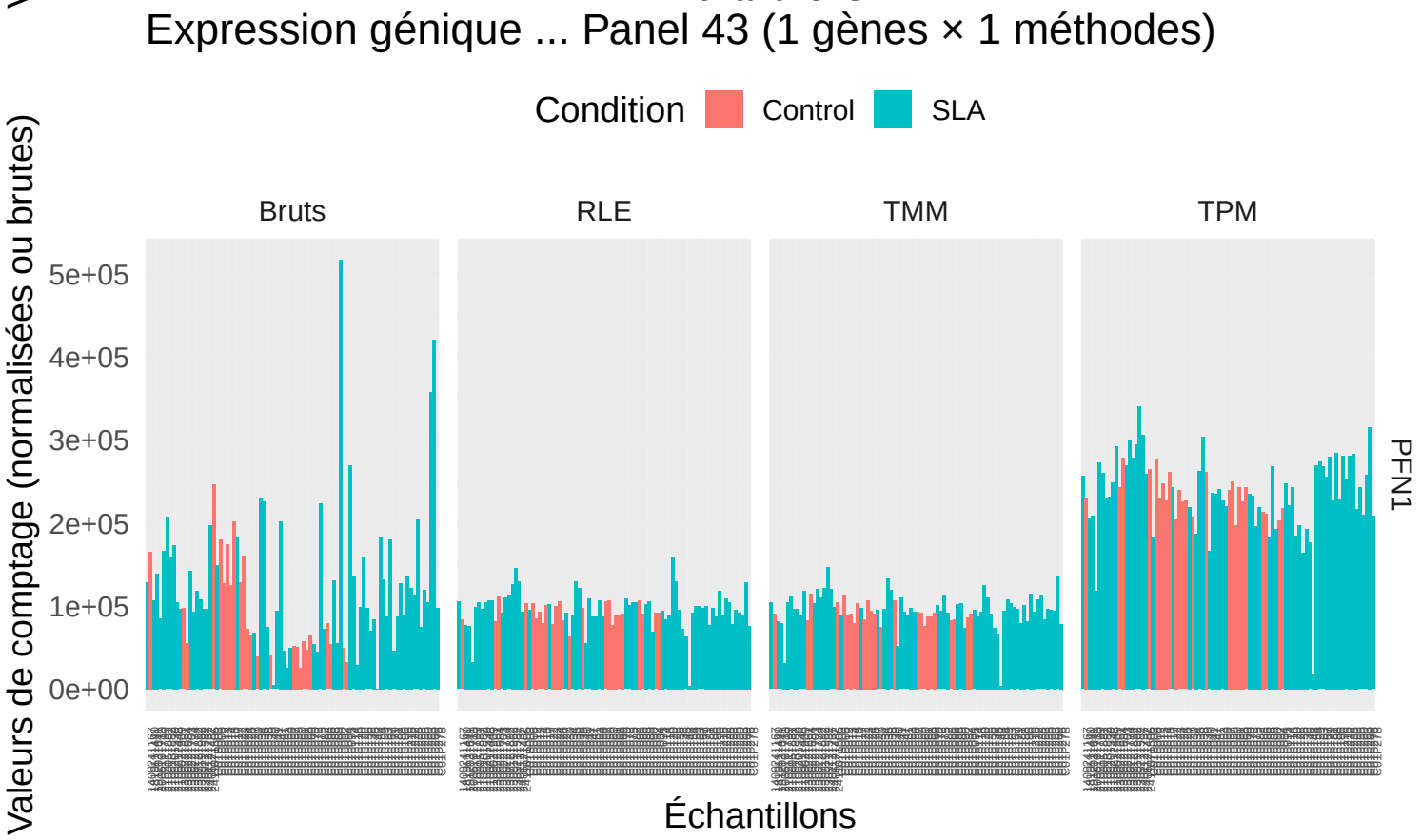

Expression génique ... Panel 38 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
2500
2000
1500
1000
500
0
ANG
Valeurs de comptage (normalisées ou brutes)
Échantillons

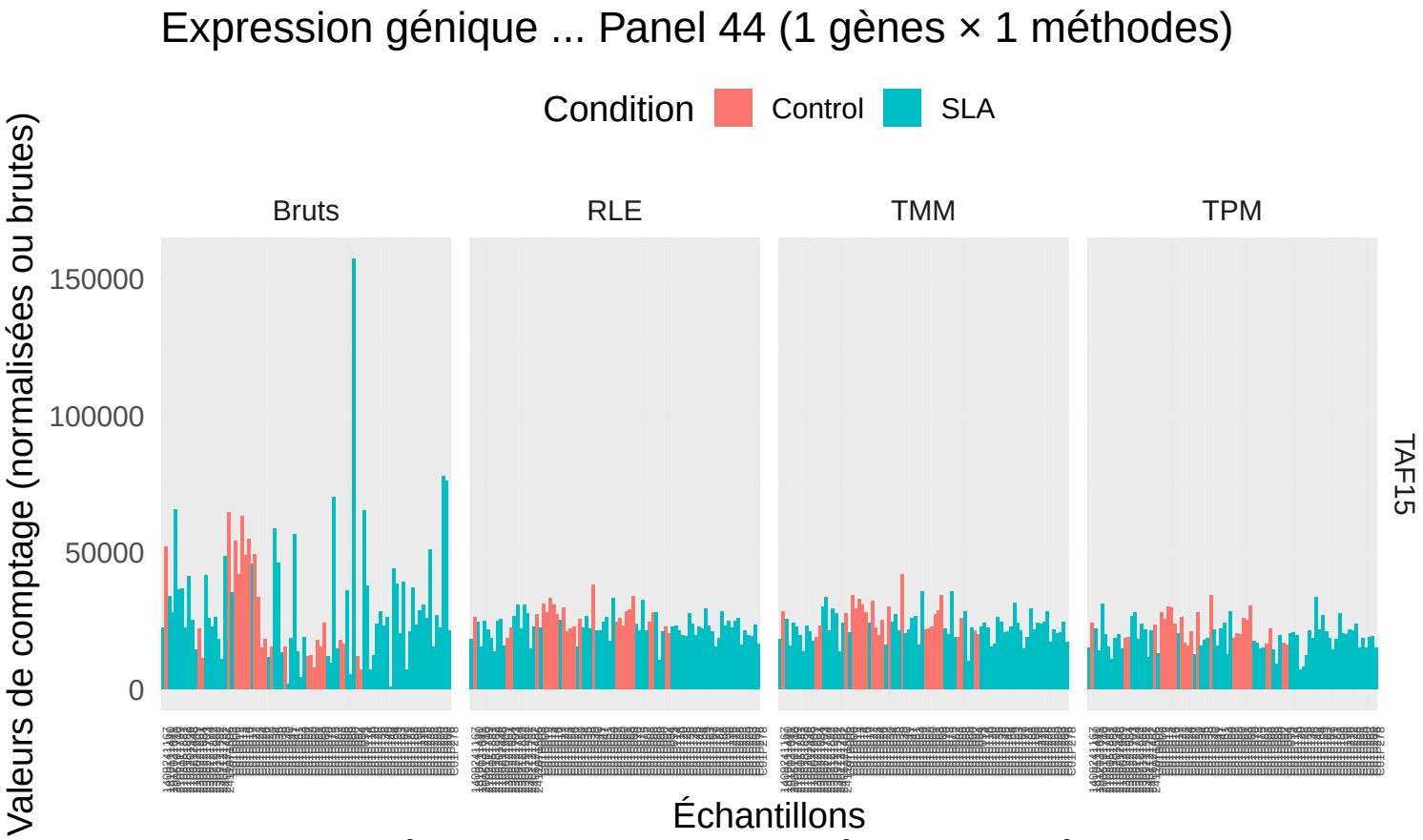

Expression génique ... Panel 39 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
150000
100000
50000
0
PSEN1
Valeurs de comptage (normalisées ou brutes)
Échantillons

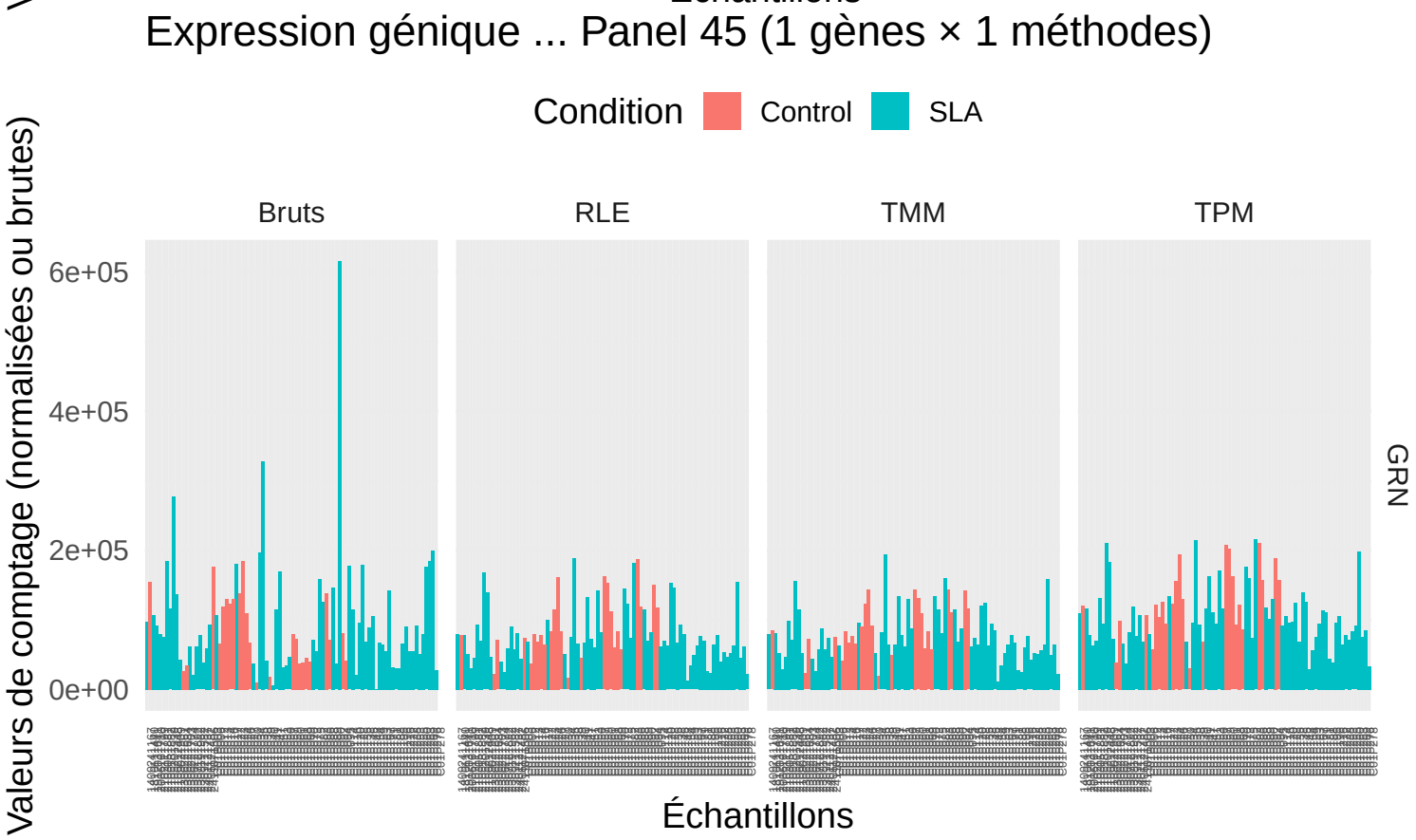

Expression génique ... Panel 40 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
5e+05
4e+05
3e+05
2e+05
1e+05
0e+00
Valeurs de comptage (normalisées ou brutes)
SPG11
Échantillons

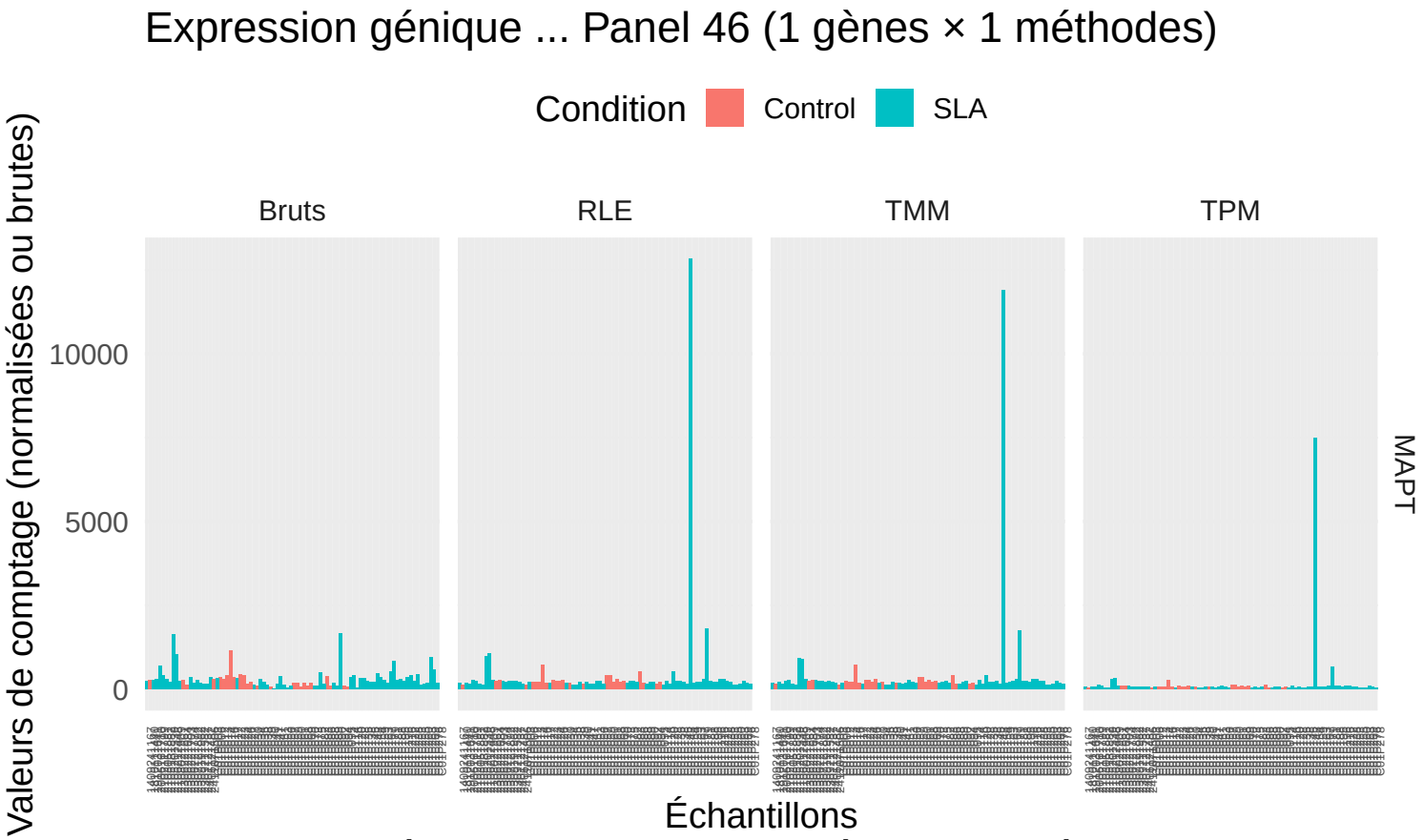

Expression génique ... Panel 41 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
25000
20000
15000
10000
5000
0
Valeurs de comptage (normalisées ou brutes)
CCNF
Échantillons

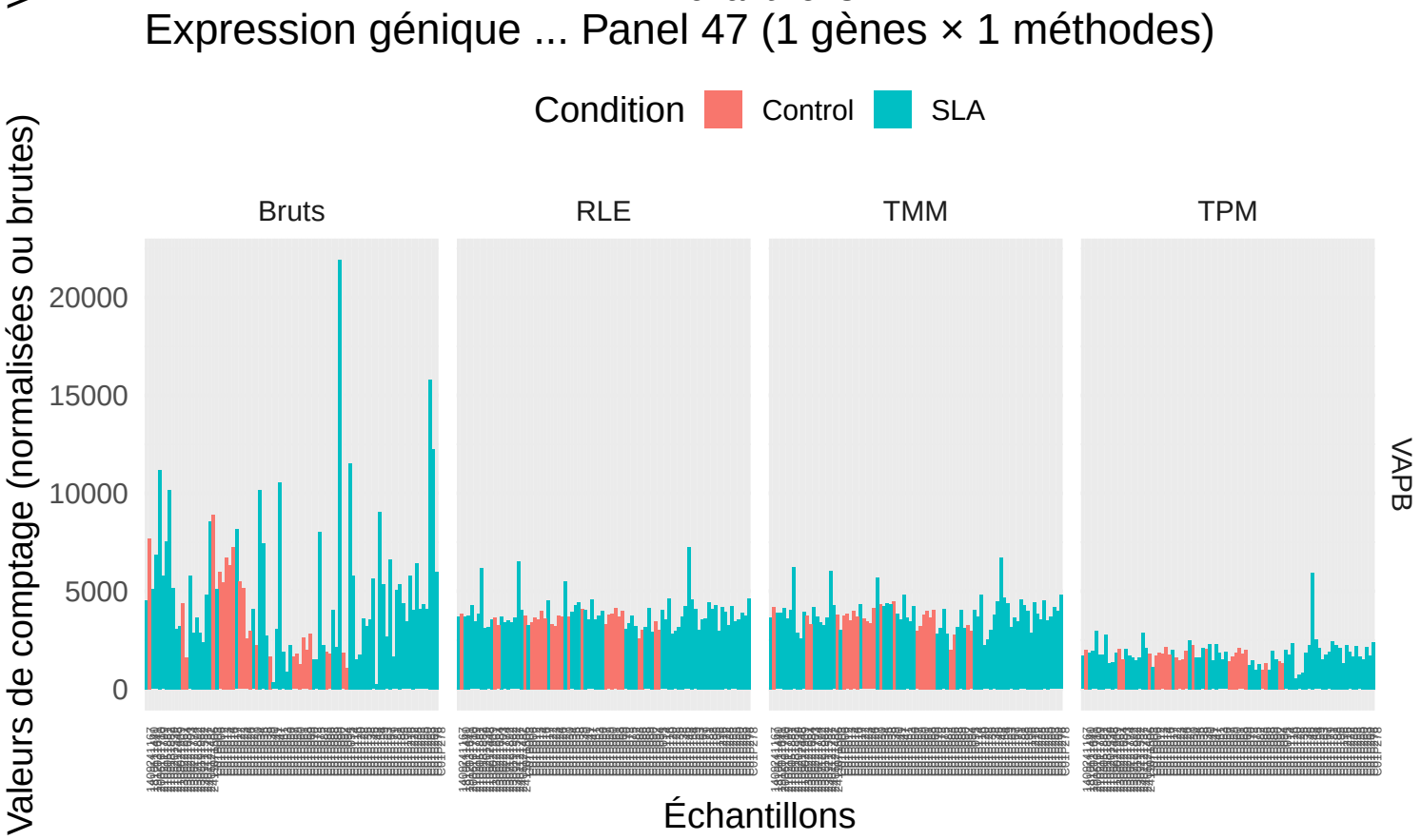

Expression génique ... Panel 42 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
250000
200000
150000
100000
50000
0
FUS
Valeurs de comptage (normalisées ou brutes)
Échantillons

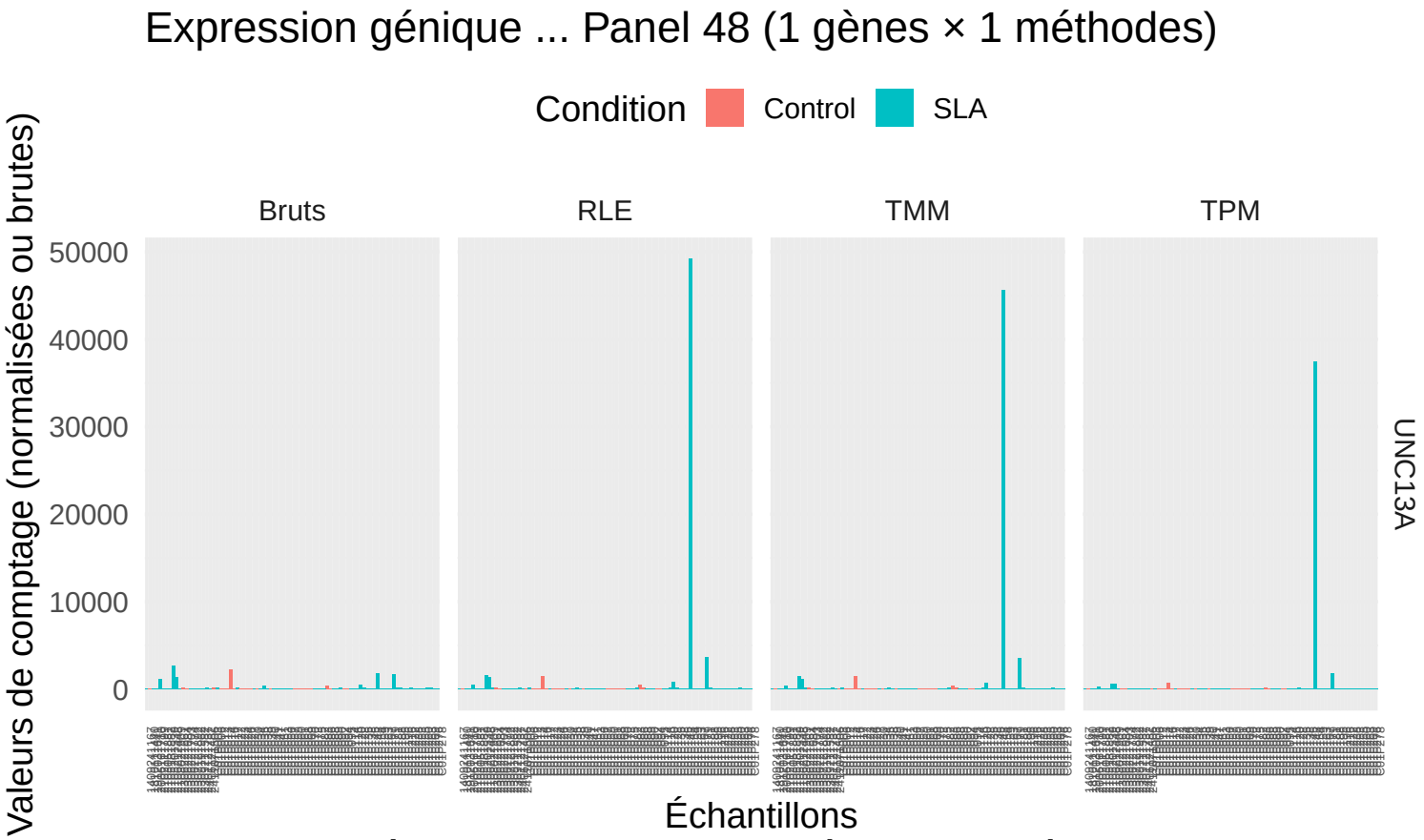

Expression génique ... Panel 43 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
5e+05
4e+05
3e+05
2e+05
1e+05
0e+00
PFN1
Valeurs de comptage (normalisées ou brutes)
Échantillons

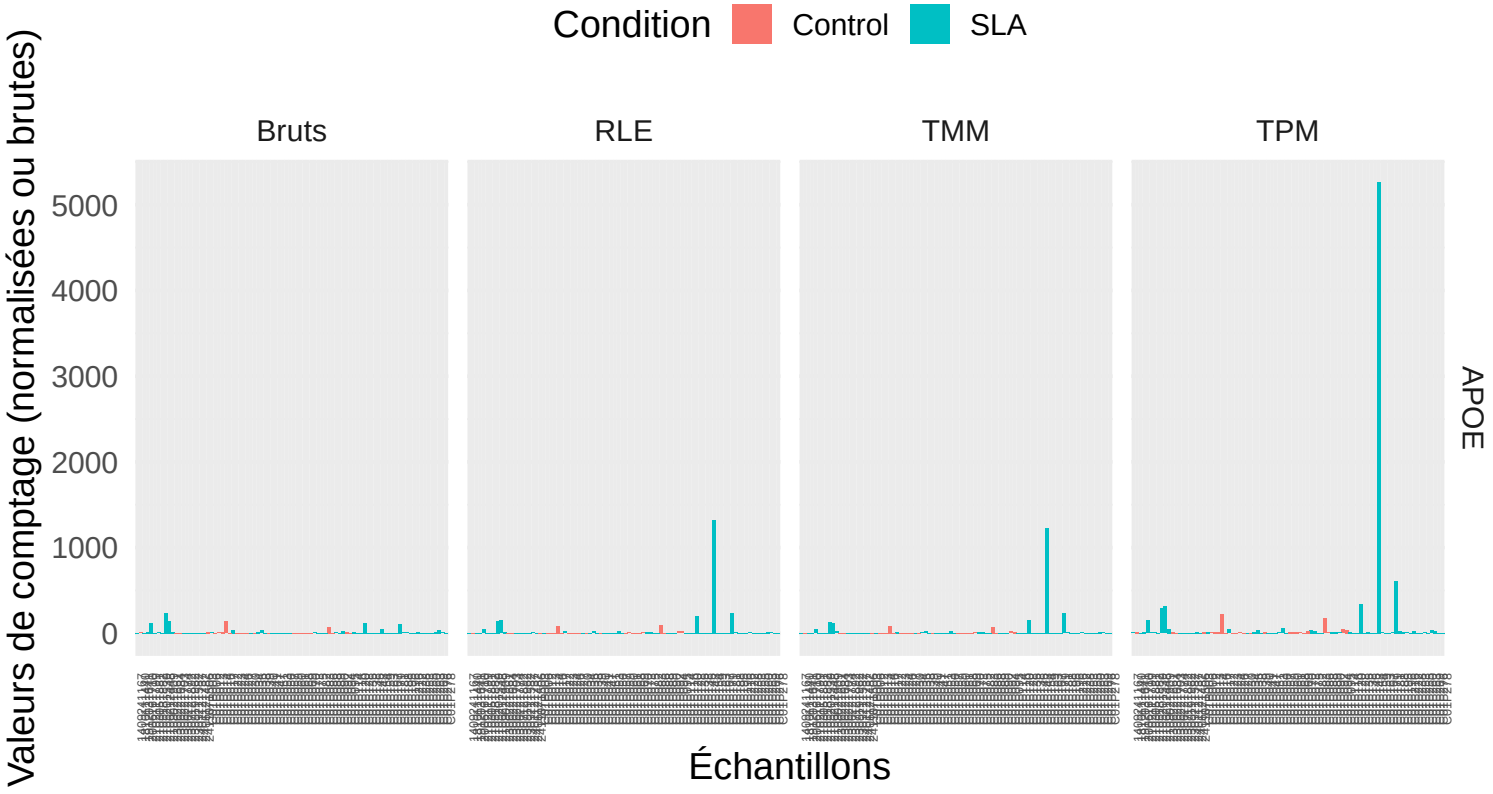

Expression génique ... Panel 44 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
150000
100000
50000
0
Valeurs de comptage (normalisées ou brutes)
TAF15
Échantillons

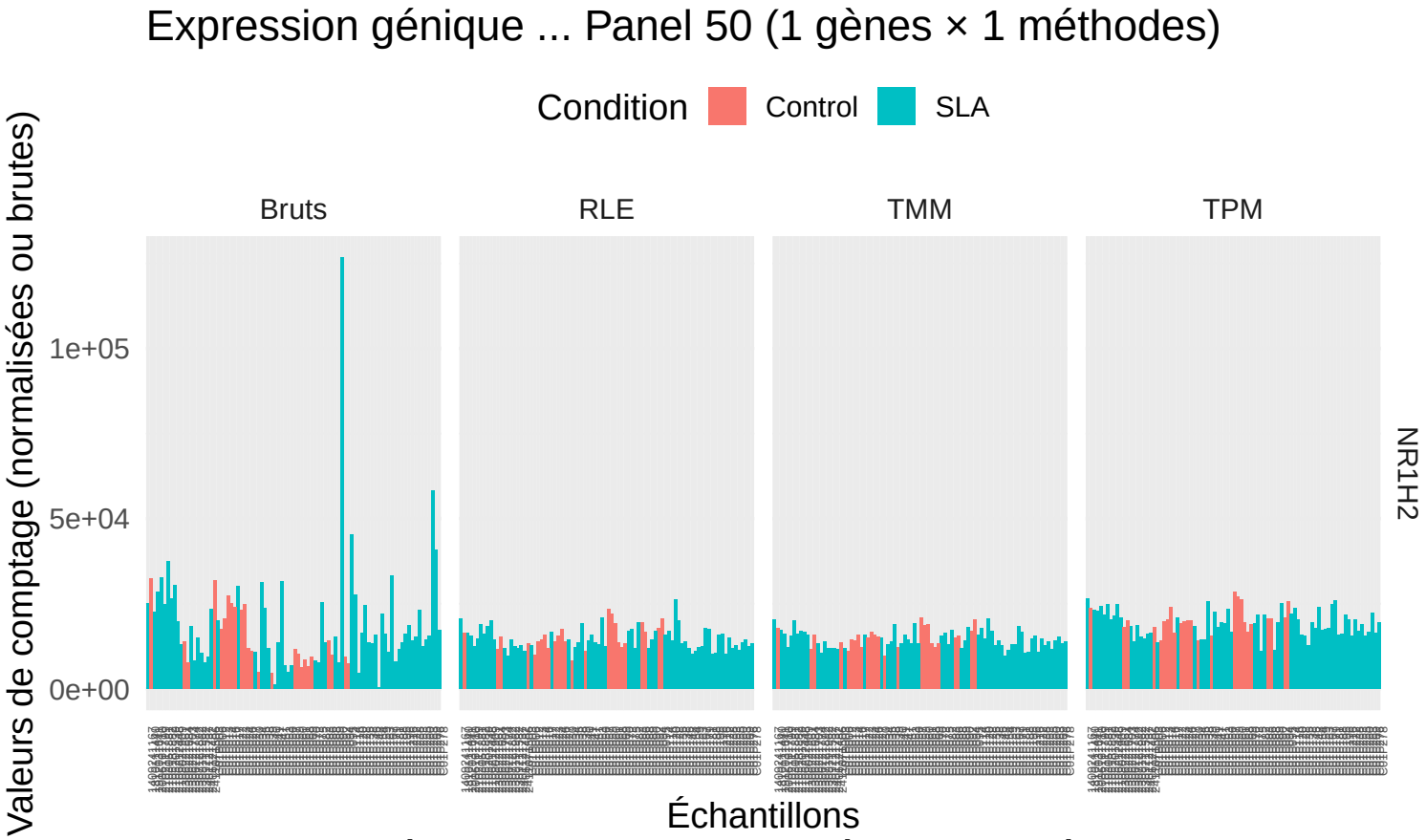

Expression génique ... Panel 45 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
6e+05
4e+05
2e+05
0e+00
Valeurs de comptage (normalisées ou brutes)
GRN
Échantillons

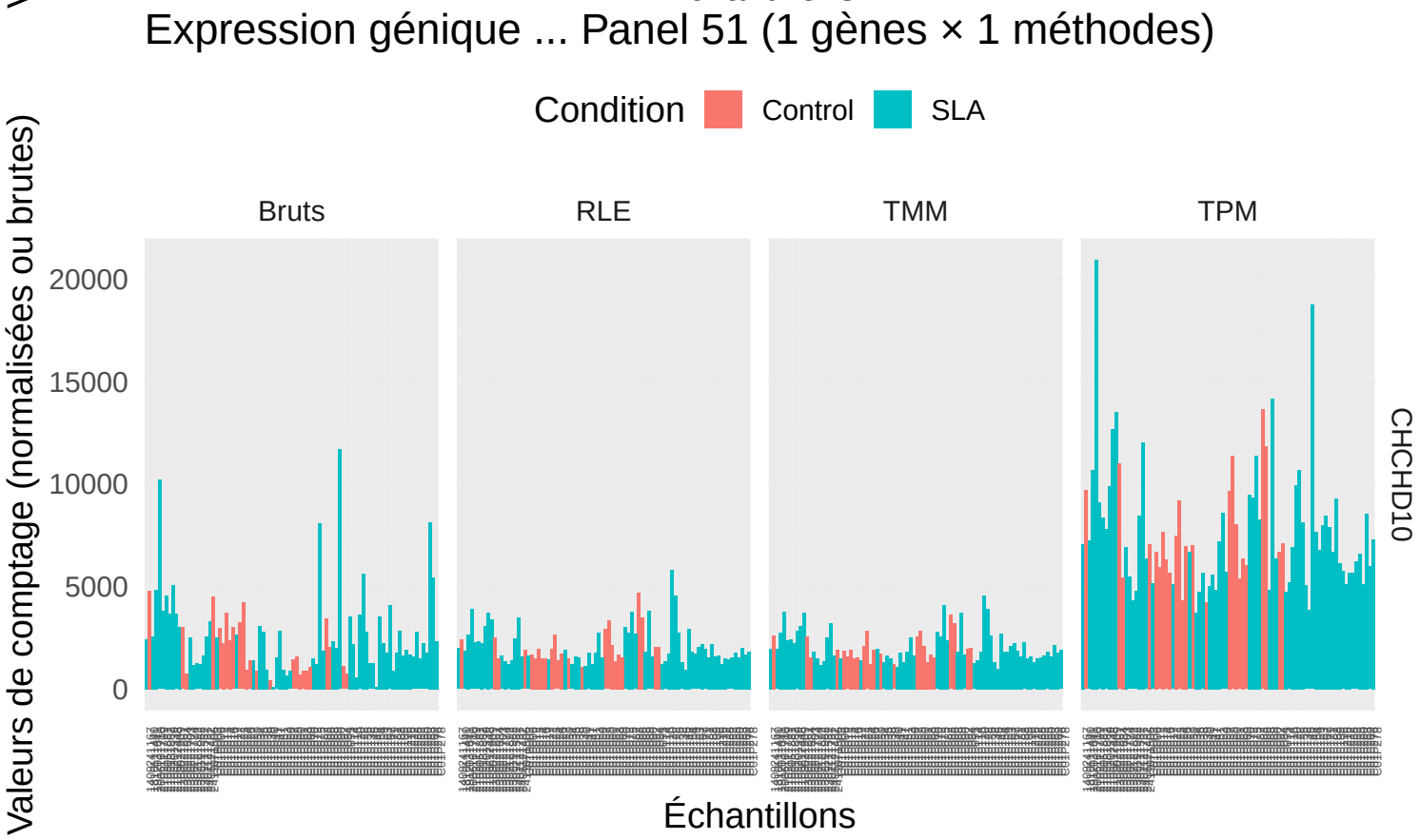

Expression génique ... Panel 46 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
10000
5000
0
MAPT
Échantillons

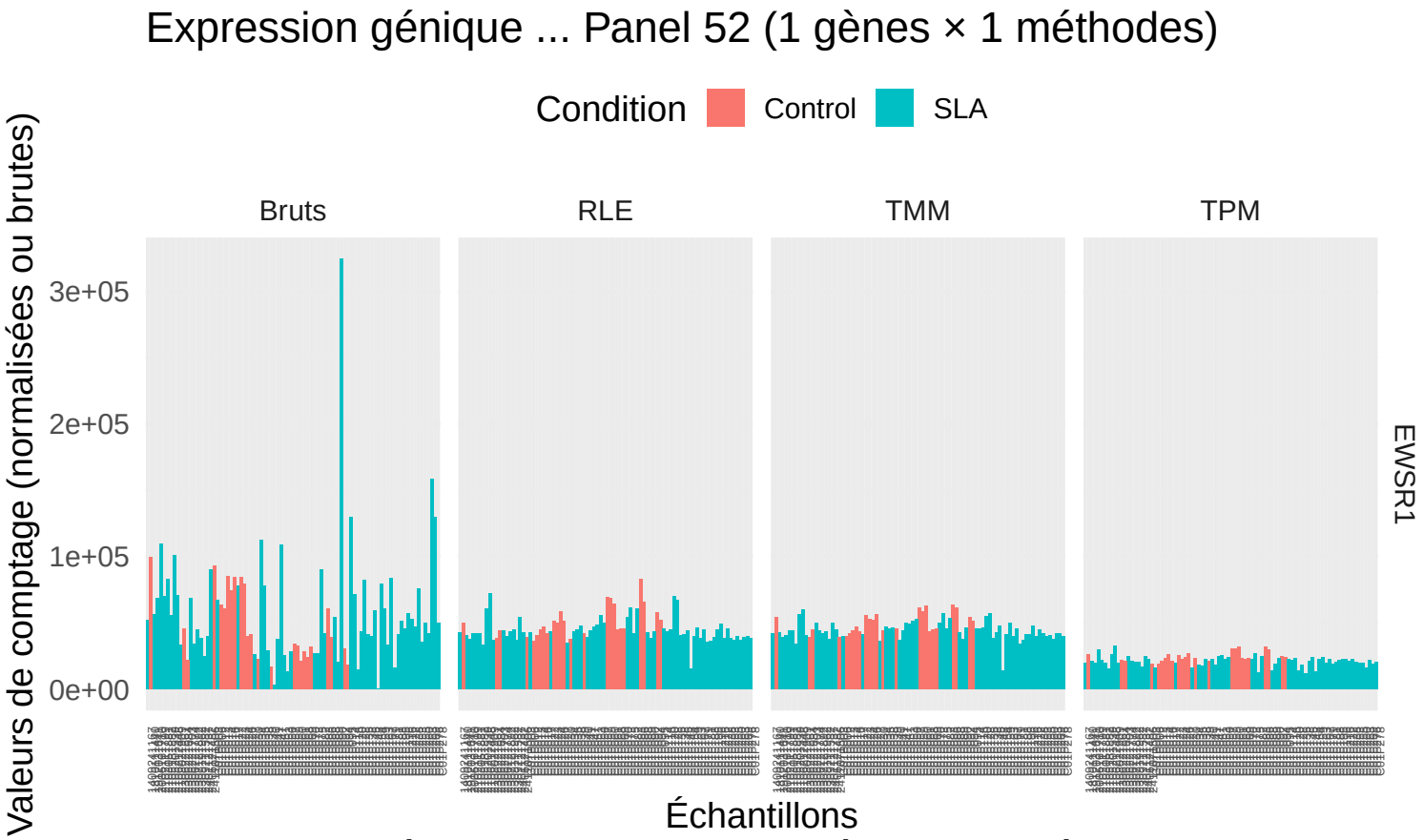

Expression génique ... Panel 47 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
20000
15000
10000
5000
0
VAPB
Échantillons

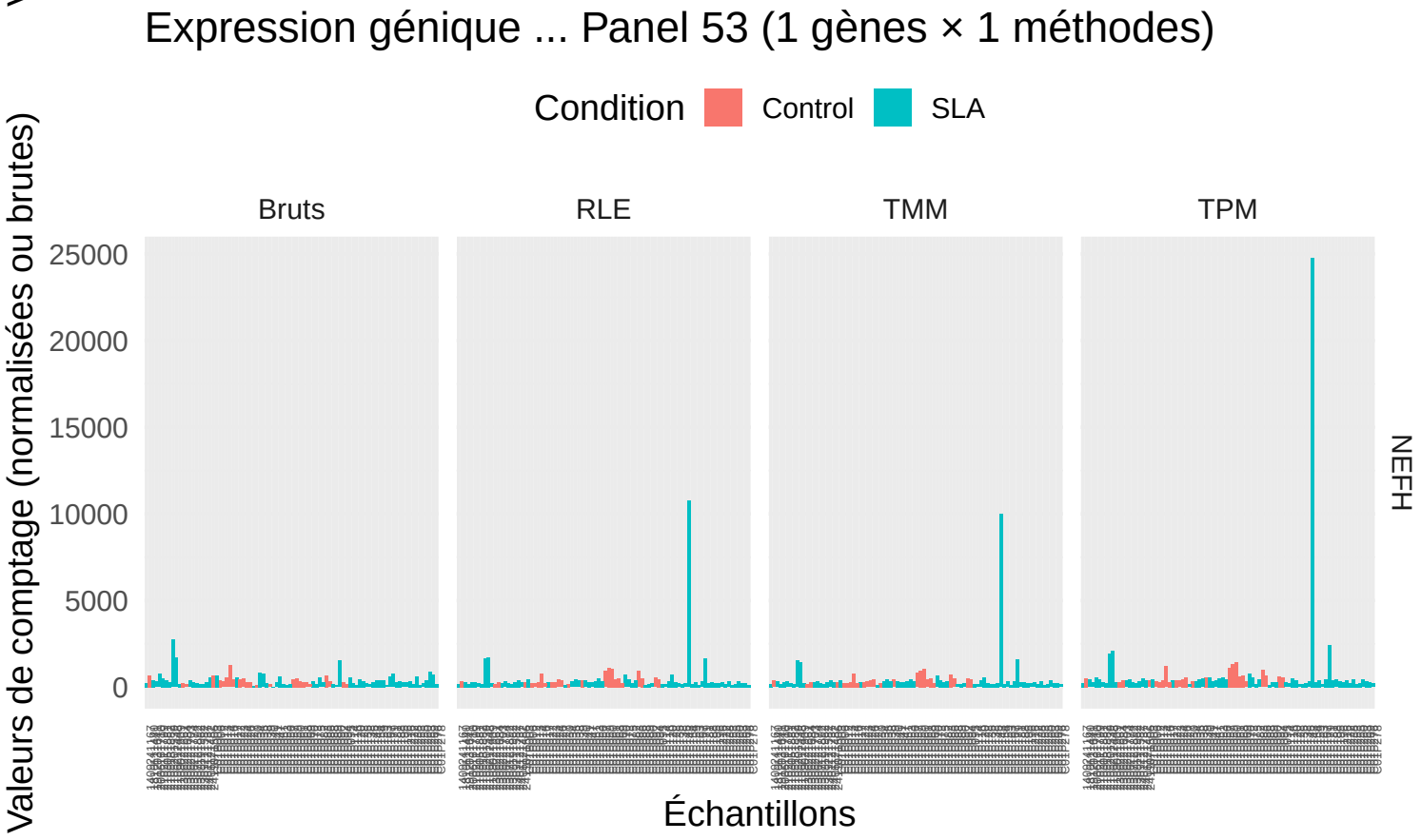

Expression génique ... Panel 48 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
50000
40000
30000
20000
10000
0
Valeurs de comptage (normalisées ou brutes)
UNC13A
Échantillons

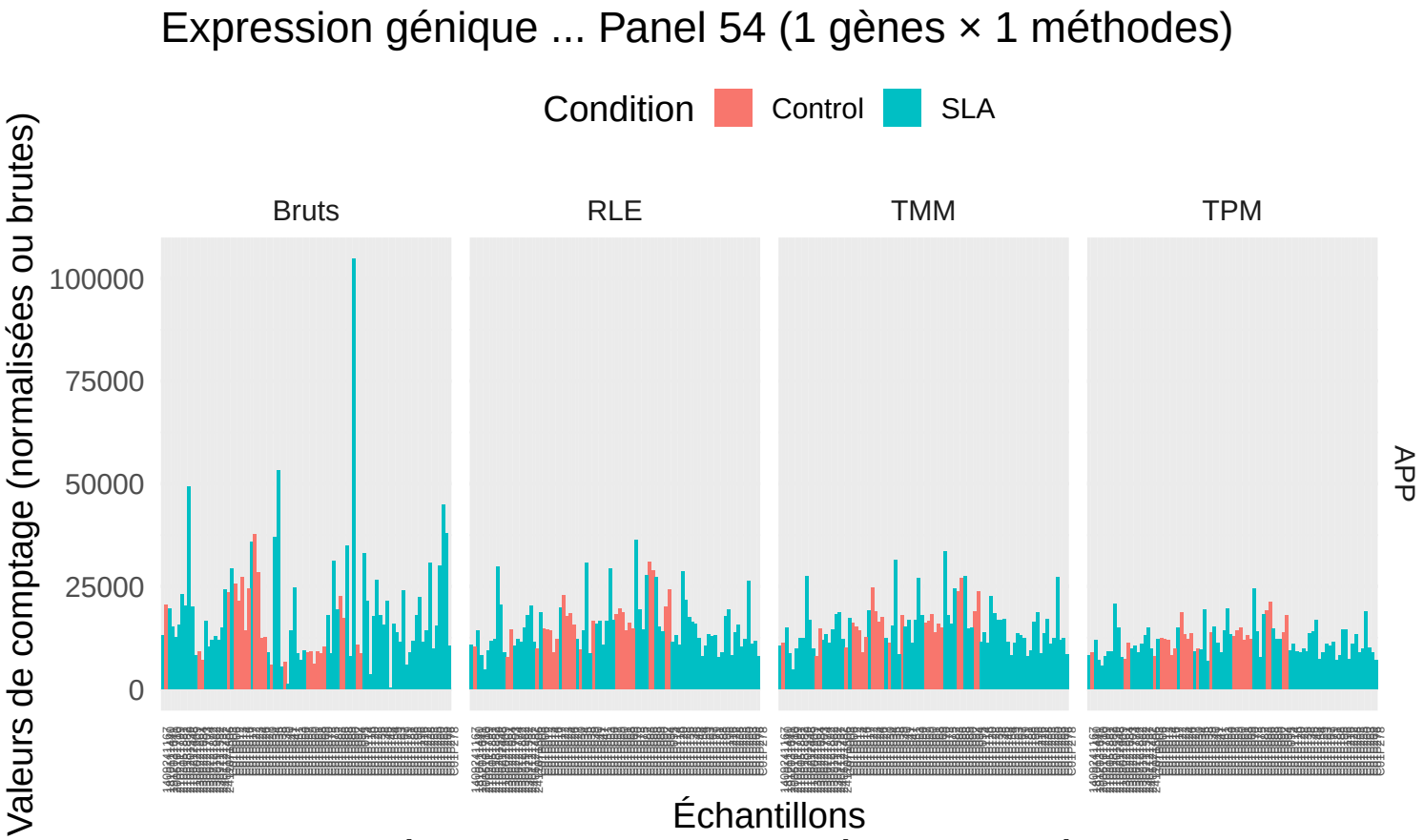

Expression génique ... Panel 49 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
5000
4000
3000
2000
1000
0
Valeurs de comptage (normalisées ou brutes)
APOE
Échantillons

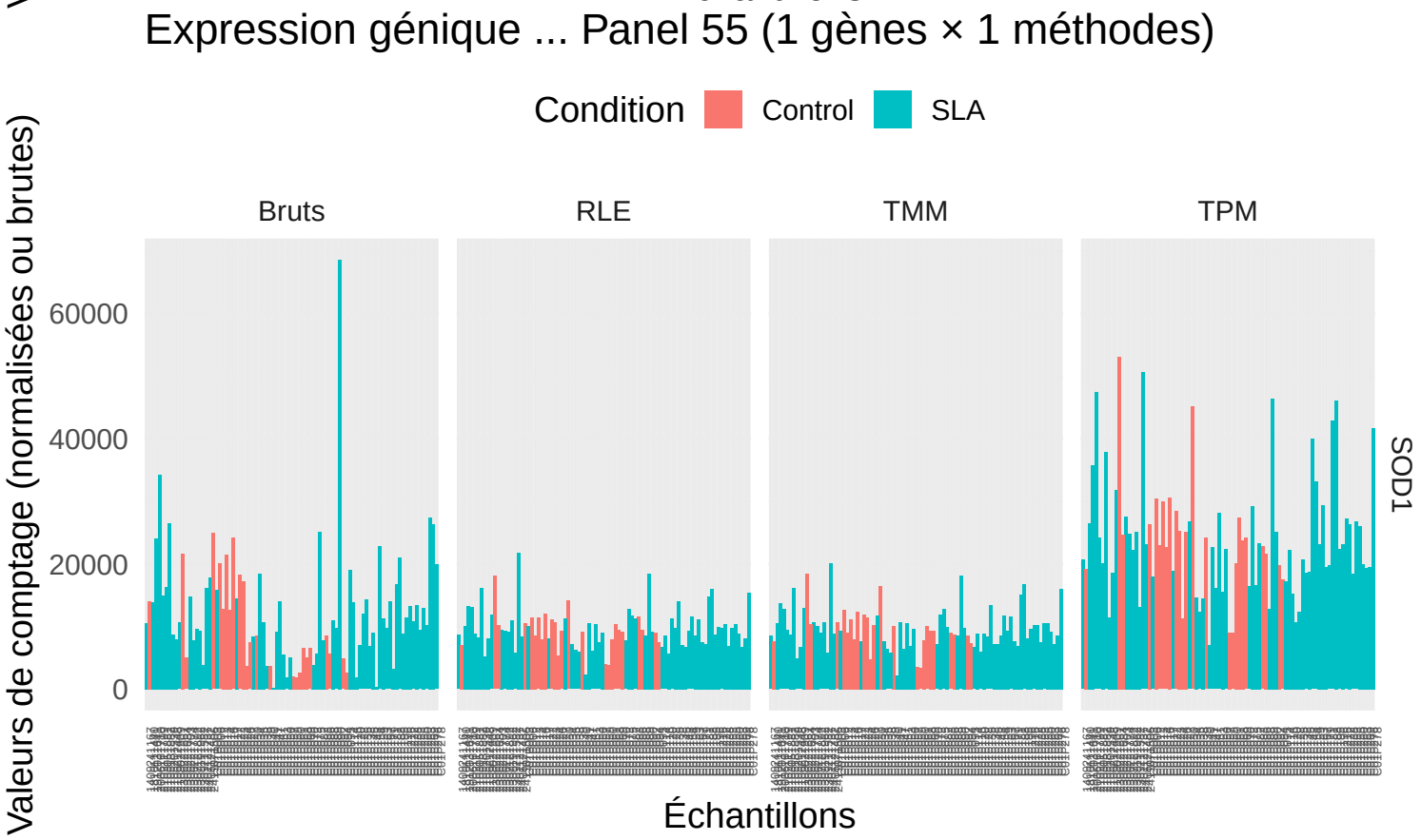

Expression génique ... Panel 50 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
1e+05
5e+04
0e+00
NR1H2
Échantillons

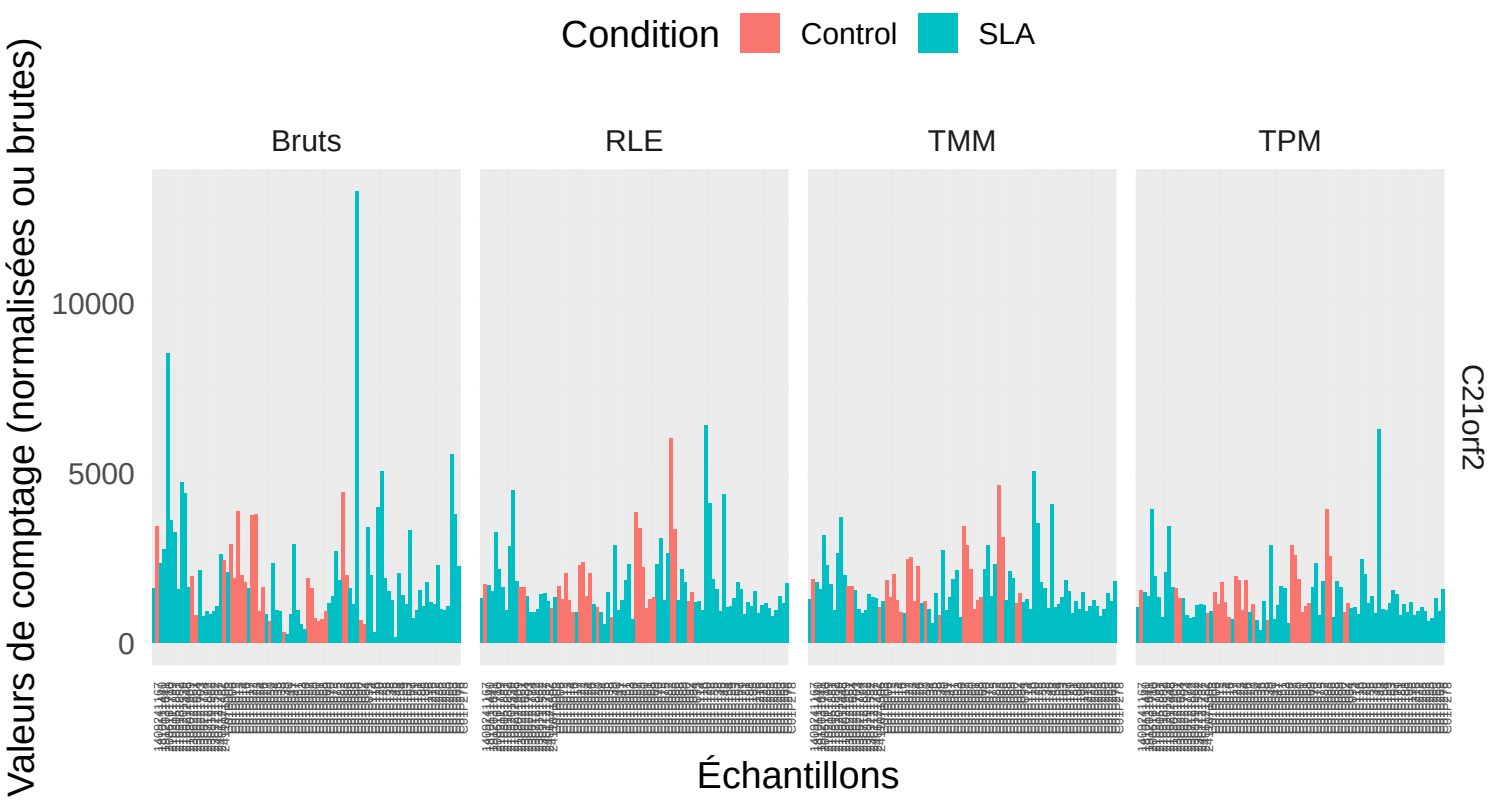

Expression génique ... Panel 51 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
Valeurs de comptage (normalisées ou brutes)
20000
15000
10000
5000
0
CHCHD10
Échantillons


Expression génique ... Panel 52 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
3e+05
2e+05
1e+05
0e+00
Valeurs de comptage (normalisées ou brutes)
EWSR1
Échantillons


Expression génique ... Panel 53 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
25000
20000
15000
10000
5000
0
Valeurs de comptage (normalisées ou brutes)
NEFH
Échantillons


Expression génique ... Panel 54 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
100000
75000
50000
25000
0
APP
Valeurs de comptage (normalisées ou brutes)
Échantillons


Expression génique ... Panel 55 (1 gènes × 1 méthodes)
Condition
Control
SLA
Bruts
RLE
TMM
TPM
60000
40000
20000
0
SOD1
Valeurs de comptage (normalisées ou brutes)
Échantillons




```
# Montage de la matrice des données normalisées pour l'export

matrice_finale_export <- donnees_normalisees %>%
  mutate(TPM_transcrits_par_M = round(TPM_transcrits_par_M, digit=0),
         TMM_cpm = round(TMM_cpm, digit = 0),
         RLE_cpm = round(RLE_cpm, digit = 0)) %>%
  select(echantillon,nom_gene, total_lectures, TPM_transcrits_par_M, TMM_cpm, RLE_cpm) %>%
  arrange(nom_gene, echantillon)

matrice_finale_export <- matrice_finale_export %>%
  rename(patient     = echantillon,
    gene         = nom_gene,
    BRUTES       = total_lectures,
    TPM          = TPM_transcrits_par_M,
    TMM_CPM      = TMM_cpm,
    RLE_CPM      = RLE_cpm
  )
# Export au format CSV
write_csv(matrice_finale_export, file = "~/Matrice_TPM_TMM_RLE.csv")

# Confirmation dans le log
cat("Matrice de comptages normalisés exportée dans 'Matrice_comptages_normalisés_TPM_TMM_RLE.csv'\n")
```

```
## Matrice de comptages normalisés exportée dans 'Matrice_comptages_normalisés_TPM_TMM_RLE.csv'
```

9 Références

Robinson, M.D., & Oshlack, A. (2010). A scaling normalization method for differential expression analysis of RNA-seq data. Genome Biology.

Li, B., & Dewey, C.N. (2011). RSEM: accurate transcript quantification from RNA-Seq data. BMC Bioinformatics.

Risso, D., etal. (2014). Normalization of RNA-seq data using factor analysis of control genes or samples. Nature Biotechnology.