```
# This script processes and visualizes methylation data from brain primary and
organoid samples.
# Author: Yingjie Zhu

# Load preprocessCore explicitly and check
library(preprocessCore)

# Also register serial backend for BiocParallel (if used indirectly)
library(BiocParallel)
register(SerialParam())

library(data.table)
library(minfi)
library(SummarizedExperiment)
library(Rtsne)
library(ggplot2)
library(matrixStats)
library(dplyr)
library(IlluminaHumanMethylationEPICv2anno.20a1.hg38)

outdir <- "output"; dir.create(outdir, showWarnings = FALSE)

# 1. Prepare data

# Generate sample sheet
idat_files <- list.files("methylation_files", pattern = "\\.idat$", full.names = TRUE,
recursive = TRUE)

# remove _Grn.dat and _Red.dat for idat_files
idat_files <- gsub("_Grn\\.idat$", "", idat_files)
idat_files <- gsub("_Red\\.idat$", "", idat_files)
idat_files <- unique(idat_files)

sample_sheet <- data.frame(
      Sample_Name = sub(".ORG/.*","",sub("methylation_files/", "", idat_files)),
      Sample_Group = sapply(idat_files, function(x) strsplit(split = "/", x)[[1]][3]),
      Basename = idat_files) %>%
      dplyr::mutate(
         Sample_ID=gsub(".*/","",Basename),
         Sample_Group = ifelse(grepl("ORG", Sample_Group), "Organoid", "Primary")
      )

# Filter out HGG_05 since it does not have control probes
sample_sheet <- sample_sheet %>% dplyr::filter(Sample_Name != "HGG_05")

# Save sample sheet
write.csv(sample_sheet, file = file.path(outdir, "samplesheet.csv"), row.names =
FALSE)
```

```
# 2. Load and Preprocess the Data
# Load the raw data from the .idat files
#  This creates an RGChannelSet object
rgSet <- read.metharray.exp(targets = sample_sheet, force = TRUE)

# Check control probes for each sample
controls <- getControlAddress(rgSet)
for(i in 1:ncol(rgSet)) {
  cat("Sample", i, ":", colnames(rgSet)[i], "\n")
  ctrl_data <- getRed(rgSet)[controls, i]
  cat("MAD:", mad(ctrl_data, na.rm = TRUE), "\n")
  cat("Range:", range(ctrl_data, na.rm = TRUE), "\n\n")
}

# Normalize the data
#  This converts the raw data into a MethylSet

# Set annotation BEFORE normalization
annotation(rgSet) <- c(array = "IlluminaHumanMethylationEPICv2",
                       annotation = "20a1.hg38")

mSet <- preprocessFunnorm(rgSet)


# 4. Calculate M-values
mvals <- getM(mSet)

# save data
save(sample_sheet, mSet, mvals, file = file.path(outdir, "methylation_data.RData"))


# 5. Select the top 5000 most variable probes for t-SNE
probe_vars <- rowVars(mvals)
select_probes <- order(probe_vars, decreasing = TRUE)[1:5000]
head(probe_vars[select_probes])
highly_variable_mvals <- mvals[select_probes, ]

# 6. Run Rtsne (transpose the matrix: samples in rows)
seed <- 42
set.seed(seed) # for reproducibility

tsne_out <- Rtsne(t(highly_variable_mvals),
    perplexity = 2, # Adjust perplexity as needed
    max_iter = 2000,
    check_duplicates = FALSE,
    theta = 0) # Exact t-SNE (slower but more stable)

# 7. Create a data frame for plotting
plot_df <- data.frame(
  Sample = colnames(mvals),
```

```
  TSNE1 = tsne_out$Y[,1],
  TSNE2 = tsne_out$Y[,2],
  Group = colData(mSet)$Sample_Group # Get group info
) %>%
  left_join(
    sample_sheet,
    by = c("Sample" = "Sample_ID"))

fwrite(plot_df, file = file.path(outdir,
paste0("tsne_plot_data.5000probes.seed",seed,".csv")))

# 8. Generate the t-SNE plot 
p <- ggplot(plot_df, aes(x = TSNE1, y = TSNE2, color = Group)) +
  geom_point(size = 4, alpha = 0.8) +
  geom_text(aes(label = Sample_Name), size = 3, vjust = -0.5, hjust = 0.5,
show.legend = FALSE) +
  labs(
    title = "t-SNE Plot of EPICv2 Methylation Data",
    x = "t-SNE Dimension 1",
    y = "t-SNE Dimension 2",
    color = "Sample Type"
  ) +
  theme_bw() +
  scale_color_manual(values = c("Primary" = "firebrick", "Organoid" = "dodgerblue"))
ggsave(file = file.path(outdir, paste0("tsne_plot.5000probes.seed",seed,".pdf")), plot
= p, width = 6, height = 5)
```