

Task5

2025-09-29

```
#Average and median for Quebec
quebec <- subset(CO2, Type == "Quebec")
mean(quebec$uptake)

## [1] 33.54286

median(quebec$uptake)

## [1] 37.15

#Average and median for Mississippi
mississippi <- subset(CO2, Type == "Mississippi")
mean(mississippi$uptake)

## [1] 20.88333

median(mississippi$uptake)

## [1] 19.3

#if (!require("BiocManager", quietly = TRUE))
#install.packages("BiocManager")

#BiocManager::install("airway")

library(airway)

## Loading required package: SummarizedExperiment

## Loading required package: MatrixGenerics

## Loading required package: matrixStats

##
## Attaching package: 'MatrixGenerics'

## The following objects are masked from 'package:matrixStats':
##
##      colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
##      colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
##      colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
##      colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
##      colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
##      colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
##      colWeightedMeans, colWeightedMedians, colWeightedSds,
##      colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
##      rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
##      rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
##      rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
##      rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
##      rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
##      rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
##      rowWeightedSds, rowWeightedVars

## Loading required package: GenomicRanges

## Loading required package: stats4

## Loading required package: BiocGenerics

## Loading required package: generics

##
## Attaching package: 'generics'

## The following objects are masked from 'package:base':
##
##      as.diffftime, as.factor, as.ordered, intersect, is.element, setdiff,
##      setequal, union

##
## Attaching package: 'BiocGenerics'

## The following objects are masked from 'package:stats':
##
##      IQR, mad, sd, var, xtabs

## The following objects are masked from 'package:base':
##
##      anyDuplicated, aperm, append, as.data.frame, basename, cbind,
##      colnames, dirname, do.call, duplicated, eval, evalq, Filter, Find,
##      get, grep, grepl, is.unsorted, lapply, Map, mapply, match, mget,
##      order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
##      rbind, Reduce, rownames, sapply, saveRDS, table, tapply, unique,
##      unsplit, which.max, which.min

## Loading required package: S4Vectors

##
## Attaching package: 'S4Vectors'

## The following object is masked from 'package:utils':
##
##      findMatches

## The following objects are masked from 'package:base':
##
##      expand.grid, I, unname

## Loading required package: IRanges

## Loading required package: GenomeInfoDb

## Loading required package: Biobase

## Welcome to Bioconductor
##
##      Vignettes contain introductory material; view with
##      'browseVignettes()'. To cite Bioconductor, see
##      'citation("Biobase")', and for packages 'citation("pkgname")'.

##
## Attaching package: 'Biobase'

## The following object is masked from 'package:MatrixGenerics':
##
##      rowMedians

## The following objects are masked from 'package:matrixStats':
##
##      anyMissing, rowMedians

data(airway)
# Convert to matrix and check expression
exprs <- assay(airway)
# Genes expressed in each sample
expressed_per_sample <- colSums(exprs > 0)

expressed_result <- data.frame(sample = colnames(exprs), expression = expressed_per_sample, row.names = NULL)
list(expressed_result)

## [[1]]
##      sample expression
## 1 SRR1039508      24633
## 2 SRR1039509      24527
## 3 SRR1039512      25699
## 4 SRR1039513      23124
## 5 SRR1039516      25508
## 6 SRR1039517      25998
## 7 SRR1039520      24662
## 8 SRR1039521      23991

# Genes not expressed in any sample
not_expressed_any <- sum(rowSums(exprs > 0) == 0)
print(not_expressed_any)

## [1] 30208
```