

## Library Size 5e6

### Loading Data

```
library(tximeta)
dir <- "/fs/cbcb-lab/rob/students/noor/Uncertainty/boot_gibbs/quant_output/Drosophilla/sim_5e6"
source("/fs/cbcb-lab/rob/students/noor/Uncertainty/boot_gibbs/R/helper_func.R")
load("/fs/cbcb-lab/rob/students/noor/Uncertainty/ase-sim/sim_5e6/sim_counts_matrix.rda") ##True Counts

fileNames <- paste("sample", sort(as.vector(outer(c("01", "02"), c("B", "GS"), paste, sep="_"))), sep=
files <- file.path(dir, fileNames, "quant.sf")
coldata <- data.frame(files, names = fileNames, infType = rep(c("Boot", "GS"), times = 2), stringsAsFac
seSamp1 <- tximeta(coldata[c(1:2),])
seSamp2 <- tximeta(coldata[c(3:4),])

seSamp1 <- appTrueCounts(seSamp1, counts_matrix[,1])
seSamp2 <- appTrueCounts(seSamp2, counts_matrix[,2])
```

### Computing size factors and confidence intervals

```
seSamp1 <- computeSizeFactors(seSamp1)

## [1] 1
## [1] 2

seSamp2 <- computeSizeFactors(seSamp2)

## [1] 1
## [1] 2

seSamp1 <- computeConfInt(seSamp1, sf = T)
seSamp2 <- computeConfInt(seSamp2, sf = T)

teSamp1 <- computeAlleleConfInt(seSamp1)
teSamp2 <- computeAlleleConfInt(seSamp2)
```

### Coverages for the replicates

```
print(computeCoverage(counts_matrix[,1], seSamp1, list(seq(nrow(seSamp1)/2)), allele = T, trSe = teSamp1)

## sample_01_B sample_01_GS
## 1 0.8203892 0.9867114

print(computeCoverage(counts_matrix[,2], seSamp2, list(seq(nrow(seSamp2)/2)), allele = T, trSe = teSamp2)

## sample_02_B sample_02_GS
## 1 0.8194995 0.987142
```

## By Counts

Plotting the coverage difference across bins of varying expression

```
nB = c(20,50)
pNb <- vector(mode = "list", length = 2)
for(i in seq_along(pNb))
  pNb[[i]] <- vector(mode = "list", length = length(nB))

m <- nrow(counts_matrix)/2
for(j in seq_along(pNb))
{
  i=1
  for(b in nB)
  {

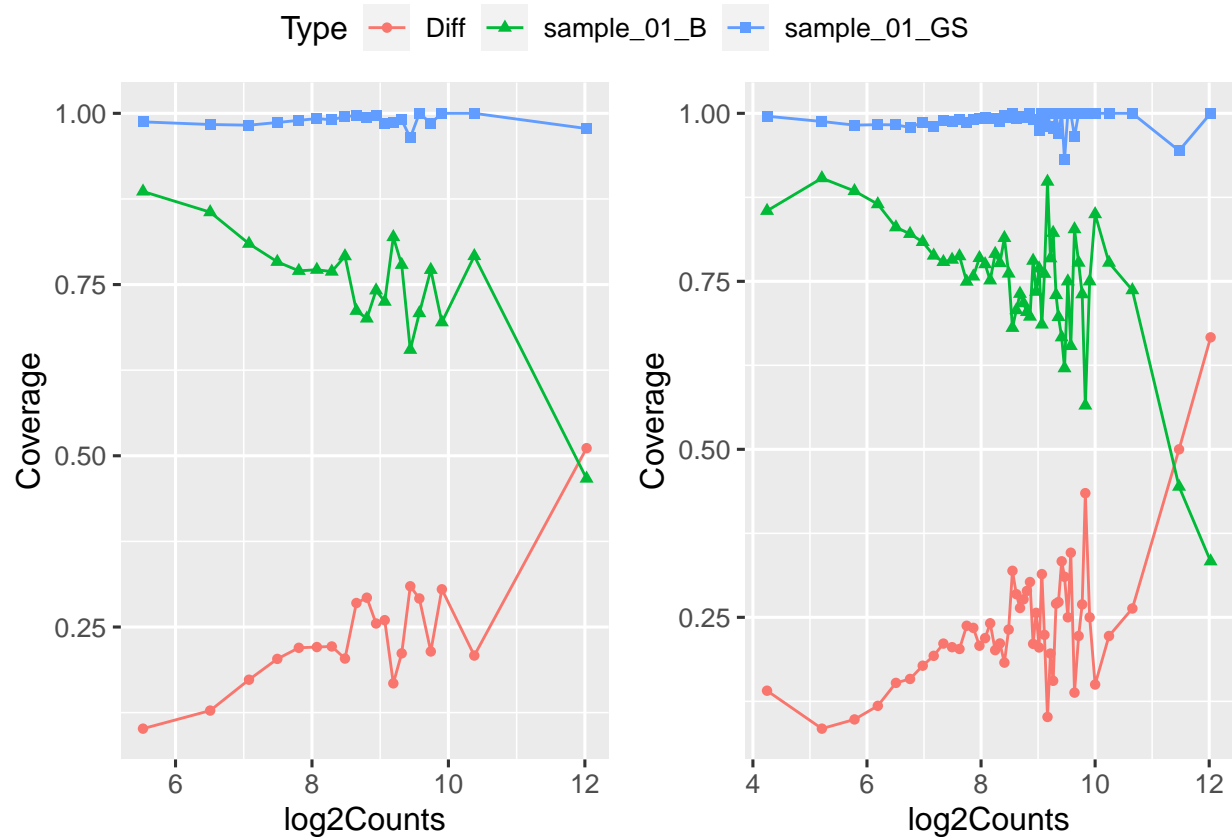
    cMat <- counts_matrix[,j][1:m] + counts_matrix[,j][m+1:m]
    cInds <- extractBinInds(cMat, breaks = b)
    if(j==1)
      covDf <- createCovDf(seSamp1, counts_matrix[,j], cInds, allele = T, trSe = teSamp1)

    else
      covDf <- createCovDf(seSamp2, counts_matrix[,j], cInds, allele = T, trSe = teSamp2)
    covDf[(b*2+1):(b*3),] = covDf[1:b,]
    covDf[(b*2+1):(b*3),3] = abs(covDf[1:b,3] - covDf[(b*1+1):(b*2),3])
    covDf[,2] = as.character(covDf[,2])
    covDf[(b*2+1):(b*3),2] = "Diff"
    pNb[[j]][[i]] <- plotCovDf(covDf, line=T)
    #print(sort(covDf[(b*2+1):(b*3),3],decreasing=T)[1:20]*100)
    i=i+1
  }
}

p1 <- ggarrange(plotlist = pNb[[1]], common.legend = T)
p2 <- ggarrange(plotlist = pNb[[2]], common.legend = T)
```

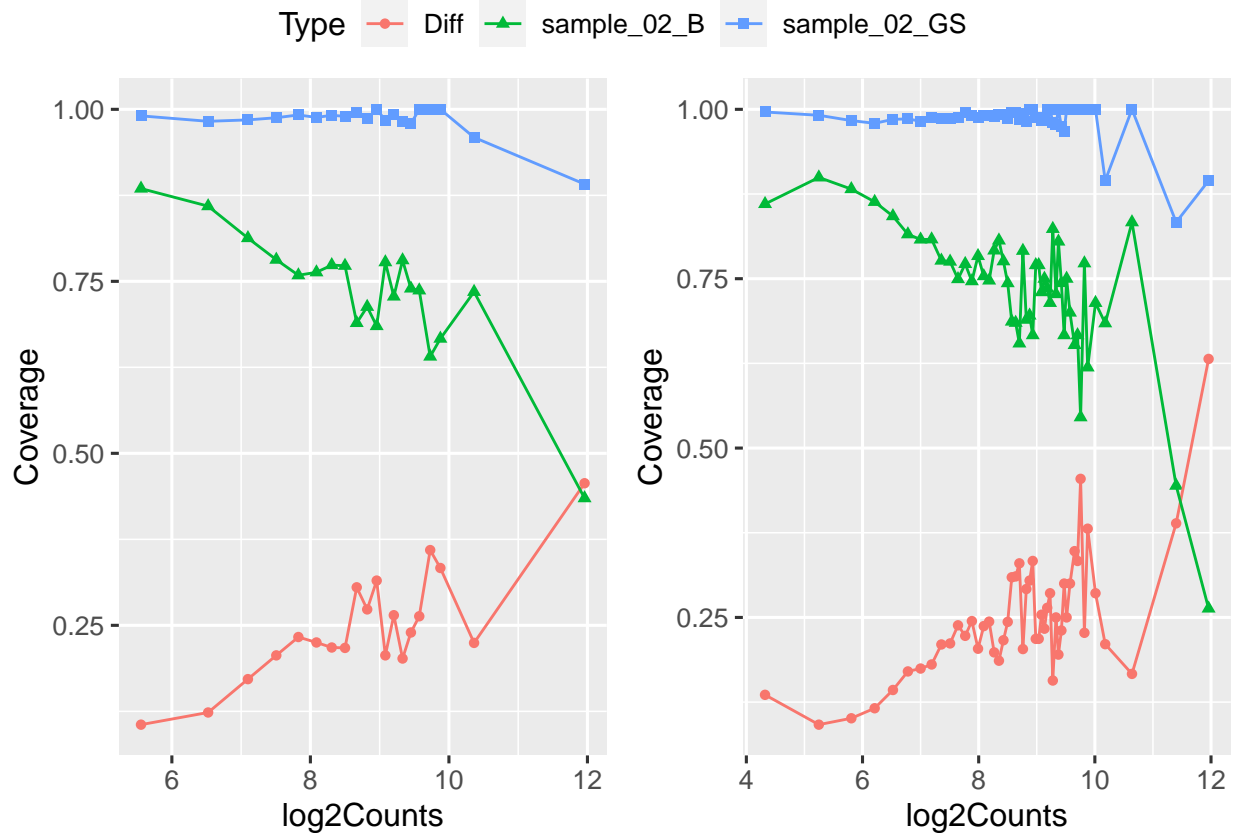
Coverage for Replicate 1 across 20 and 50 bins

```
print(p1)
```



Coverage for Replicate 2 across 20 and 50 bins

```
print(p2)
```



## Plotting by ratios

### Plotting the coverage difference across bins of varying expression

```
for(j in seq_along(pNb))
{
  i=1
  for(b in nB)
  {
    if(j==1)
    {
      ratio <- computeRatio(counts_matrix[,j])
      cInds <- extractBinInds(ratio, breaks = b)
      covDf <- createCovDf(seSamp1, counts_matrix[,j], cInds, allele = T, trSe = teSamp1, logC = 1)
    }

    else
    {
      ratio <- computeRatio(counts_matrix[,j])
      cInds <- extractBinInds(ratio, breaks = b)
      covDf <- createCovDf(seSamp2, counts_matrix[,j], cInds, allele = T, trSe = teSamp2, logC = 1)
    }
    covDf[(b*2+1):(b*3),] = covDf[1:b,]
    covDf[(b*2+1):(b*3),3] = abs(covDf[1:b,3] - covDf[(b*1+1):(b*2),3])
    covDf[,2] = as.character(covDf[,2])
    covDf[(b*2+1):(b*3),2] = "Diff"
  }
}
```

```

pNb[[j]][[i]] <- plotCovDf(covDf, line=T, ratio=T)
#print(sort(covDf[(b*2+1):(b*3),3], decreasing=T)[1:20]*100)
i=i+1
}
}

p1 <- ggarrange(plotlist = pNb[[1]], common.legend = T)
p2 <- ggarrange(plotlist = pNb[[2]], common.legend = T)

```

Coverage for Replicate 1 across 20 and 50 bins

```
print(p1)
```



Coverage for Replicate 2 across 20 and 50 bins

```
print(p2)
```

