R Notebook

```
dir <- "../../real_data/out"
load("../../real_data/out_1/sim_counts_matrix.rda")
load("environment/seList.RData")
load("environment/seListOld.RData")
source("helper_func.R")
library(ggplot2)
library(ggpubr)
```

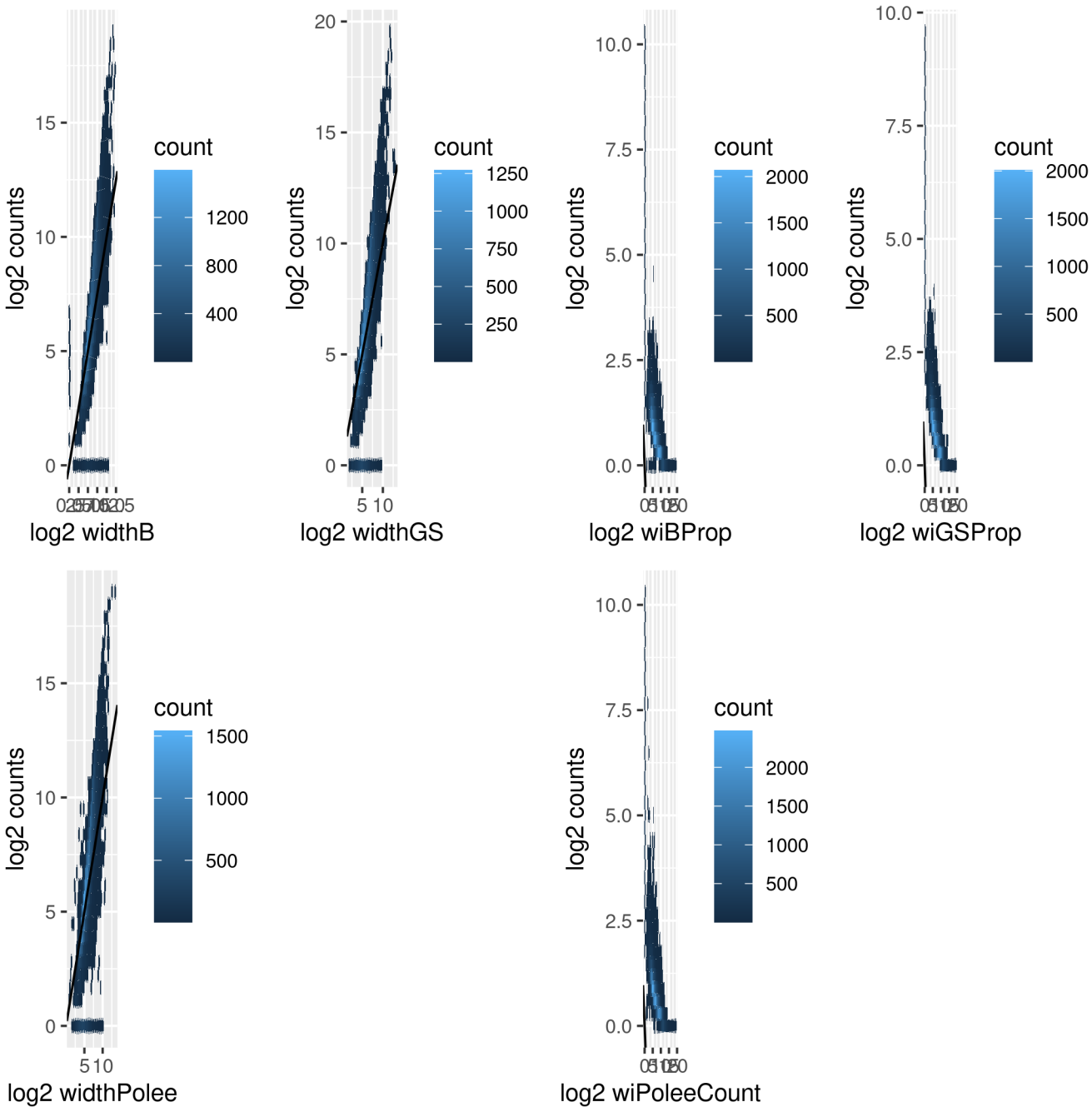
Using the fastq files

```
counts_matrix <- read.table("../../real_data/out_1/sample_01_counts.csv", header=T, row.names=1)
se <- seList[["100"]]
metadata(se)[["trueCounts"]] <- counts_matrix[,1]
se <- computeConfInt(se, sf = T)
```

Plotting Width

The plot on the left is the width with count and width/count with count

```
# df <- createWidthDf(se, counts_matrix)
# p <- plotWidthDf(df)
# pComb <- ggarrange(p[[1]], p[[2]])
knitr::include_graphics("pWidth.png")
```



Plotting the coverage difference

I vary the number of bins, which means number of transcripts decrease inside each bin as the b

```
nB = c(20,50,100,200, 500)
pNb <- vector(mode = "list", length = 5)
i=1
for(b in nB)
{
    cInds <- extractBinInds(counts_matrix[,1], breaks = b)
```

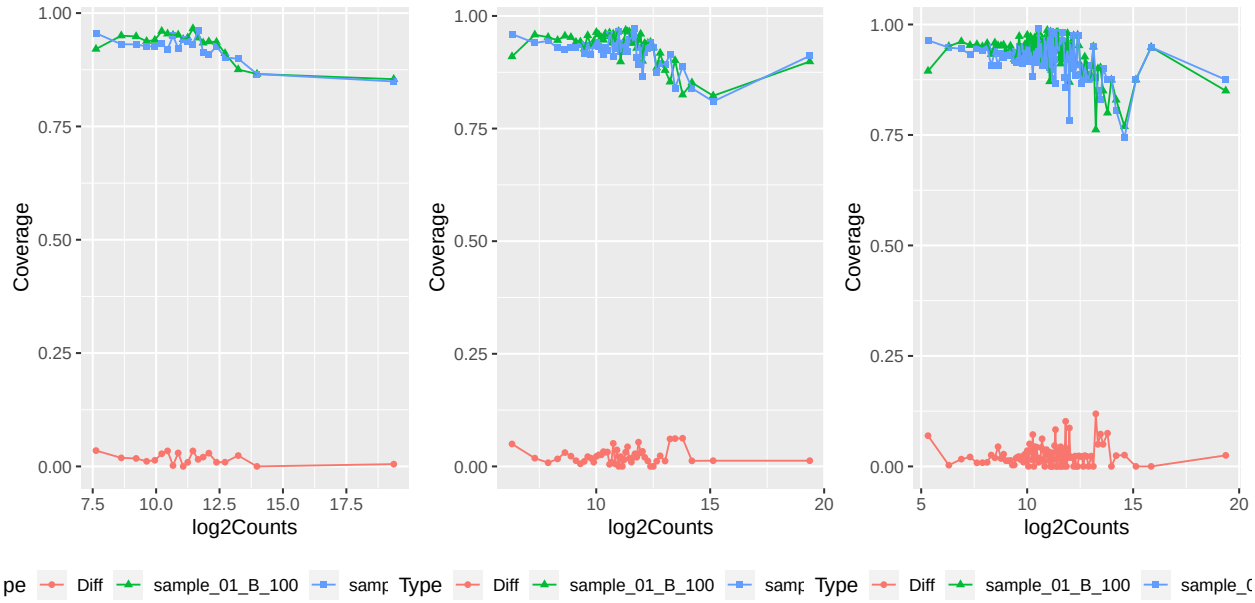
```
    covDf <- createCovDf(se, counts_matrix[,1], cInds)
    covDf[(b*2+1):(b*3),3] = abs(covDf[1:b,3] - covDf[(b*1+1):(b*2),3])
    covDf[,2] = as.character(covDf[,2])
    covDf[(b*2+1):(b*3),2] = "Diff"
    pNb[[i]] <- plotCovDf(covDf, line=T)
    print(sort(covDf[(b*2+1):(b*3),3],decreasing=T)[1:20]*100)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

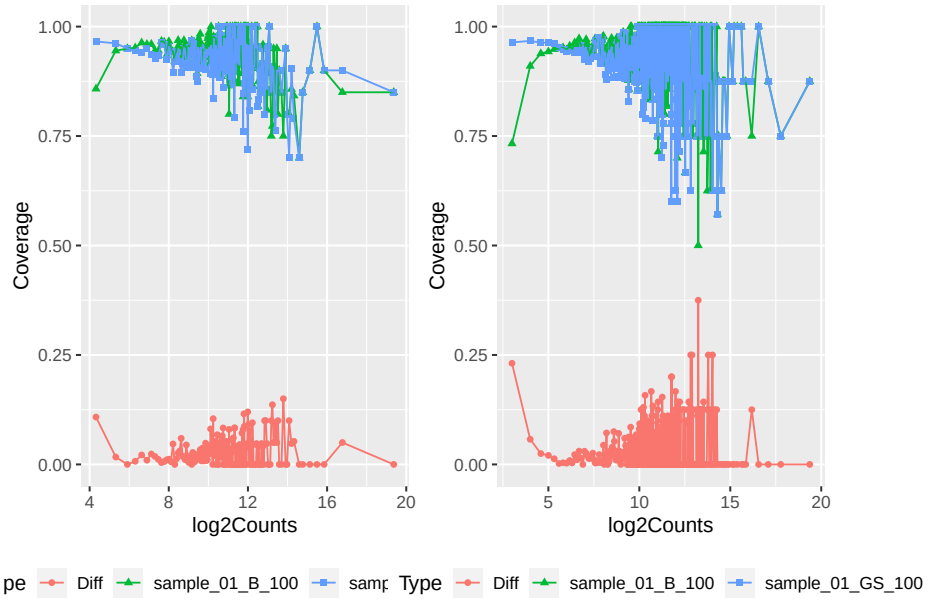
```
##
## Attaching package: 'reshape'
```

```
## The following objects are masked from 'package:S4Vectors':
##
##     expand, rename
```

```
##  [1] 3.4875881 3.4201954 3.4129693 2.9680365 2.9288703 2.7888446 2.3923445
##  [8] 2.0491803 1.8863636 1.7596567 1.5564202 1.3525699 1.1291461 0.9345794
## [15] 0.9132420 0.9036145 0.5025126 0.1865672 0.0000000 0.0000000
##  [1] 6.250000 6.172840 6.097561 5.376344 5.113636 4.981374 4.347826 3.658537
##  [9] 3.333333 3.278689 3.196347 3.149606 3.137255 3.071429 3.000000 2.803738
## [17] 2.573529 2.555911 2.352941 2.274795
##  [1] 11.904762 10.204082  8.695652  8.333333  7.500000  7.317073  7.207207
##  [8]  6.947621  6.250000  5.095541  5.000000  5.000000  4.687500  4.477612
## [15]  4.458599  4.444444  4.347826  4.255319  4.000000  4.000000
##  [1] 15.000000 13.636364 12.000000 11.538462 10.808005 10.447761 10.000000
##  [8] 10.000000 10.000000 10.000000 10.000000  9.523810  9.523810  9.090909
## [15]  8.695652  8.333333  8.333333  8.333333  8.108108  8.000000
##  [1] 37.50000 25.00000 25.00000 25.00000 25.00000 23.10164 20.00000 20.00000
##  [9] 16.66667 16.66667 15.78947 15.38462 14.28571 14.28571 14.28571 14.28571
## [17] 14.28571 13.33333 13.04348 12.50000
```

```
i=1
ggarrange(plotlist = pNb)
```

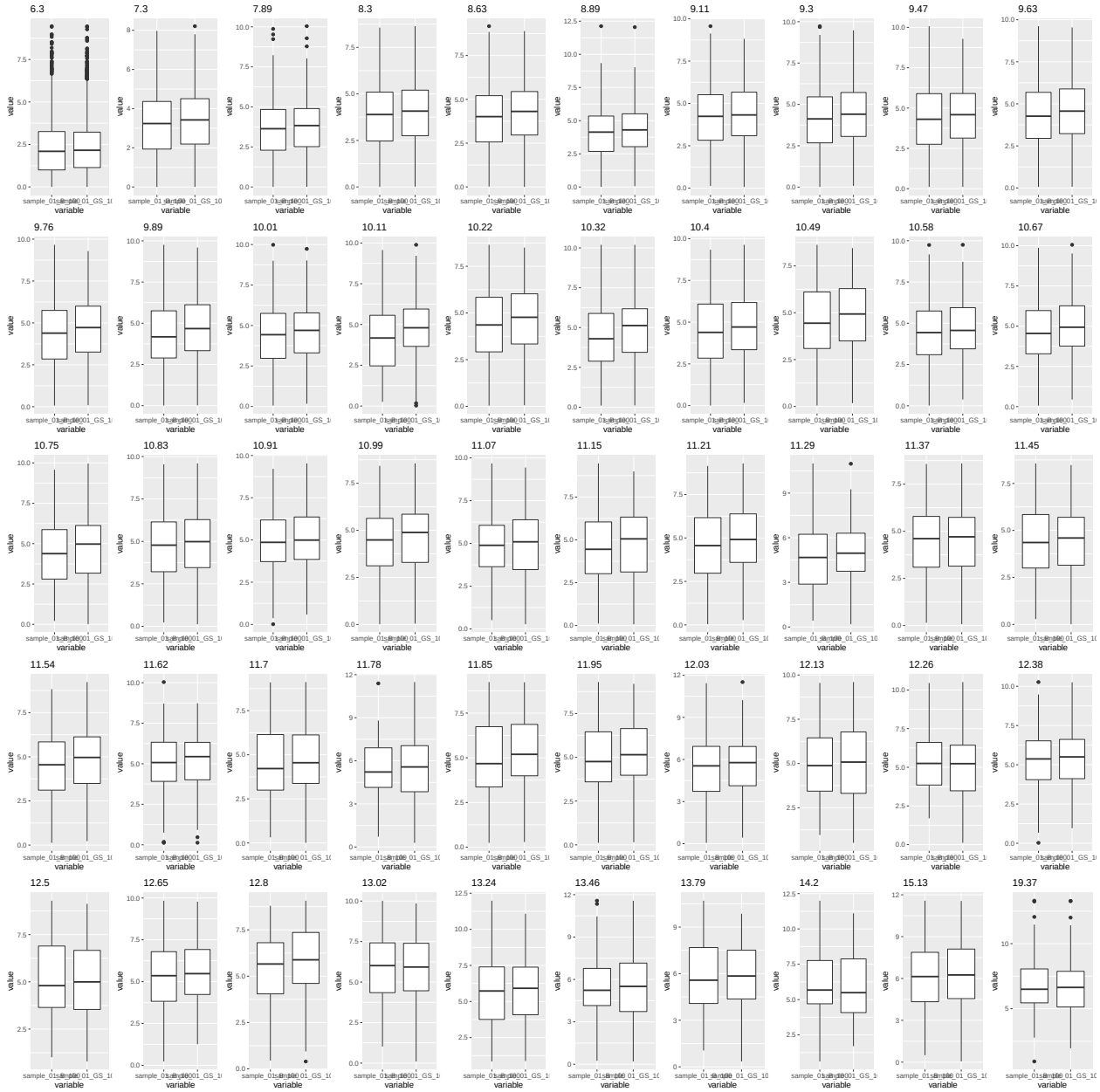




Boxplots for 50 bins

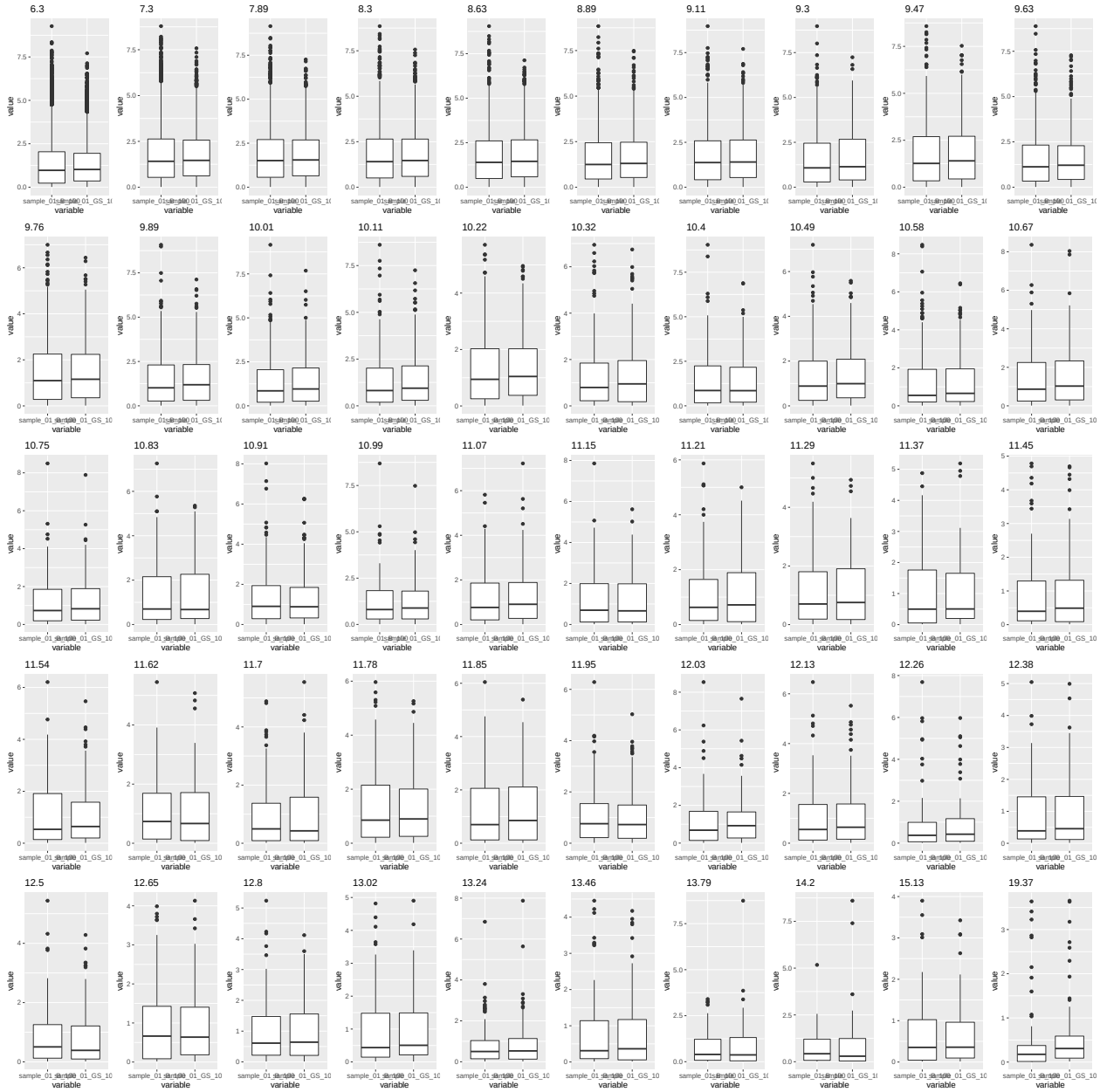
Plotting bias

```
pL <- plotSummary(se, counts_matrix[,1], summQuant="bias", nbreaks = 50, type = "BP")
ggarrange(plotlist = pL, nrow = 5, ncol = 10)
```



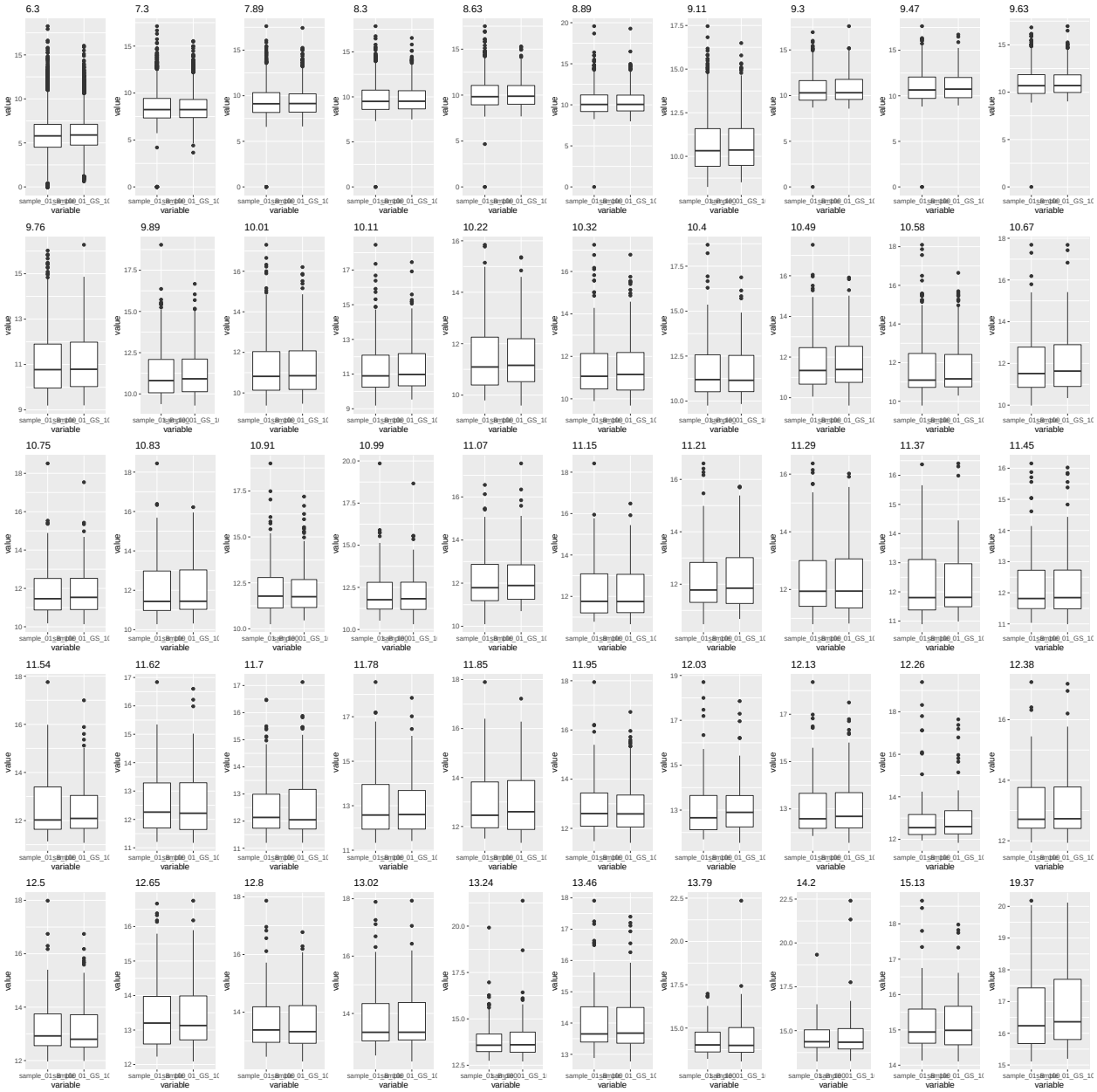
Plotting infRV

```
pL <- plotSummary(se, counts_matrix[,1], summQuant="infRV", nbreaks = 50, type = "BP")
ggarrange(plotlist = pL, nrow = 5, ncol = 10)
```



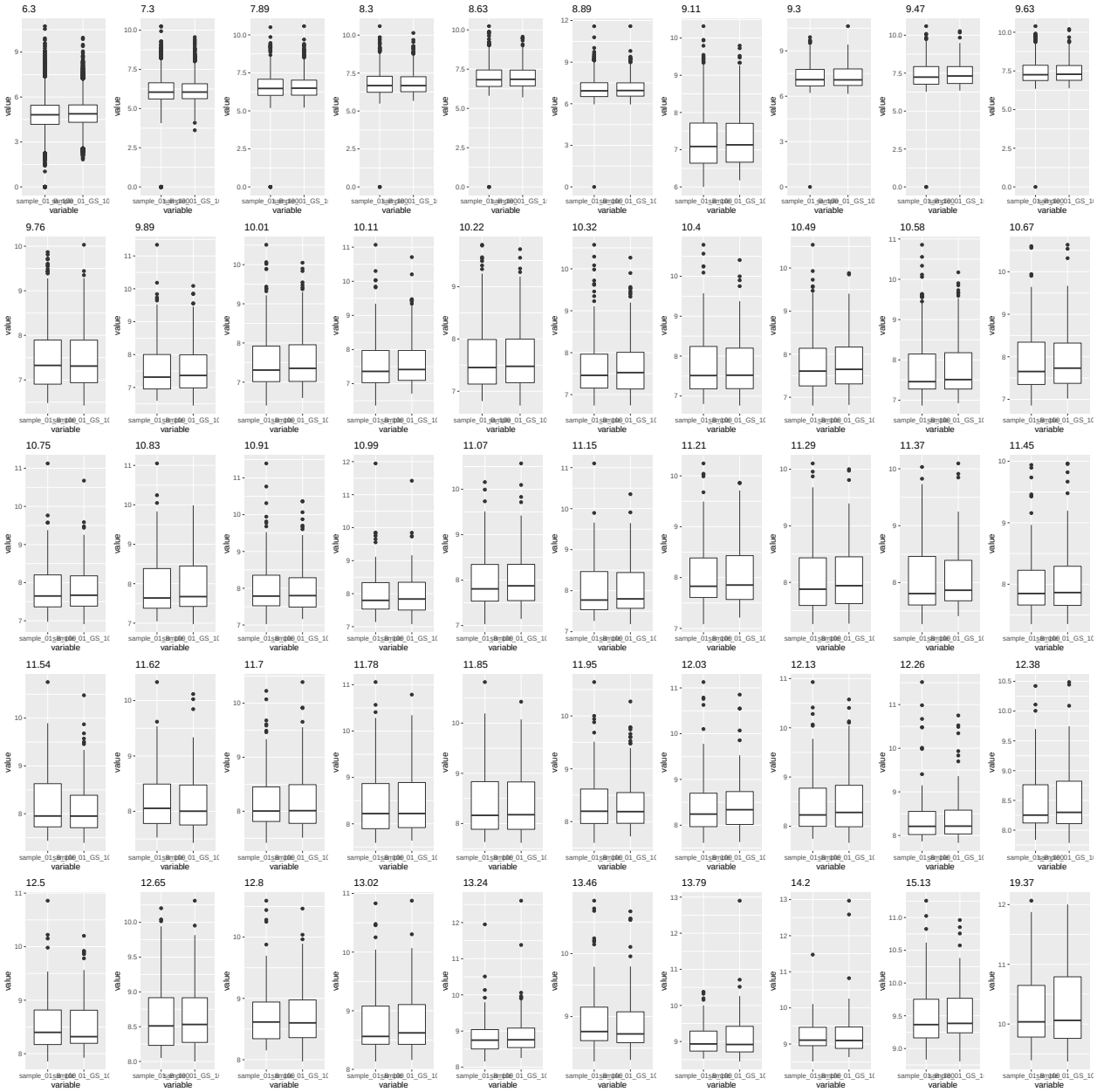
Plotting variance

```
pL <- plotSummary(se, counts_matrix[,1], summQuant="variance", nbreaks = 50, type = "BP")
ggarrange(plotlist = pL, nrow = 5, ncol = 10)
```



Plotting Width

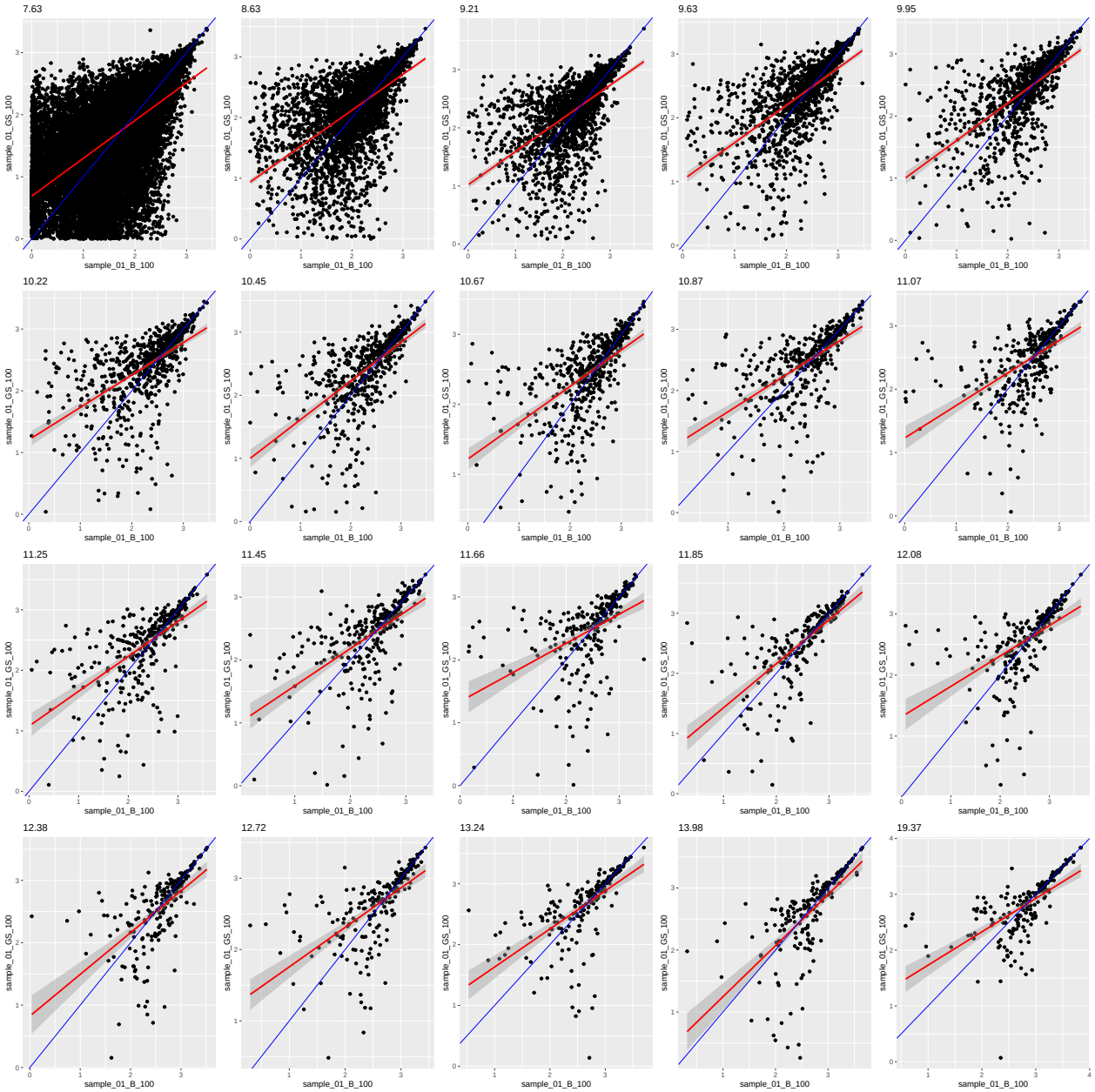
```
pL <- plotSummary(se, counts_matrix[,1], summQuant="Width", nbreaks = 50, type = "BP")
ggarrange(plotlist = pL, nrow = 5, ncol = 10)
```



Scatterplot for 20 bins

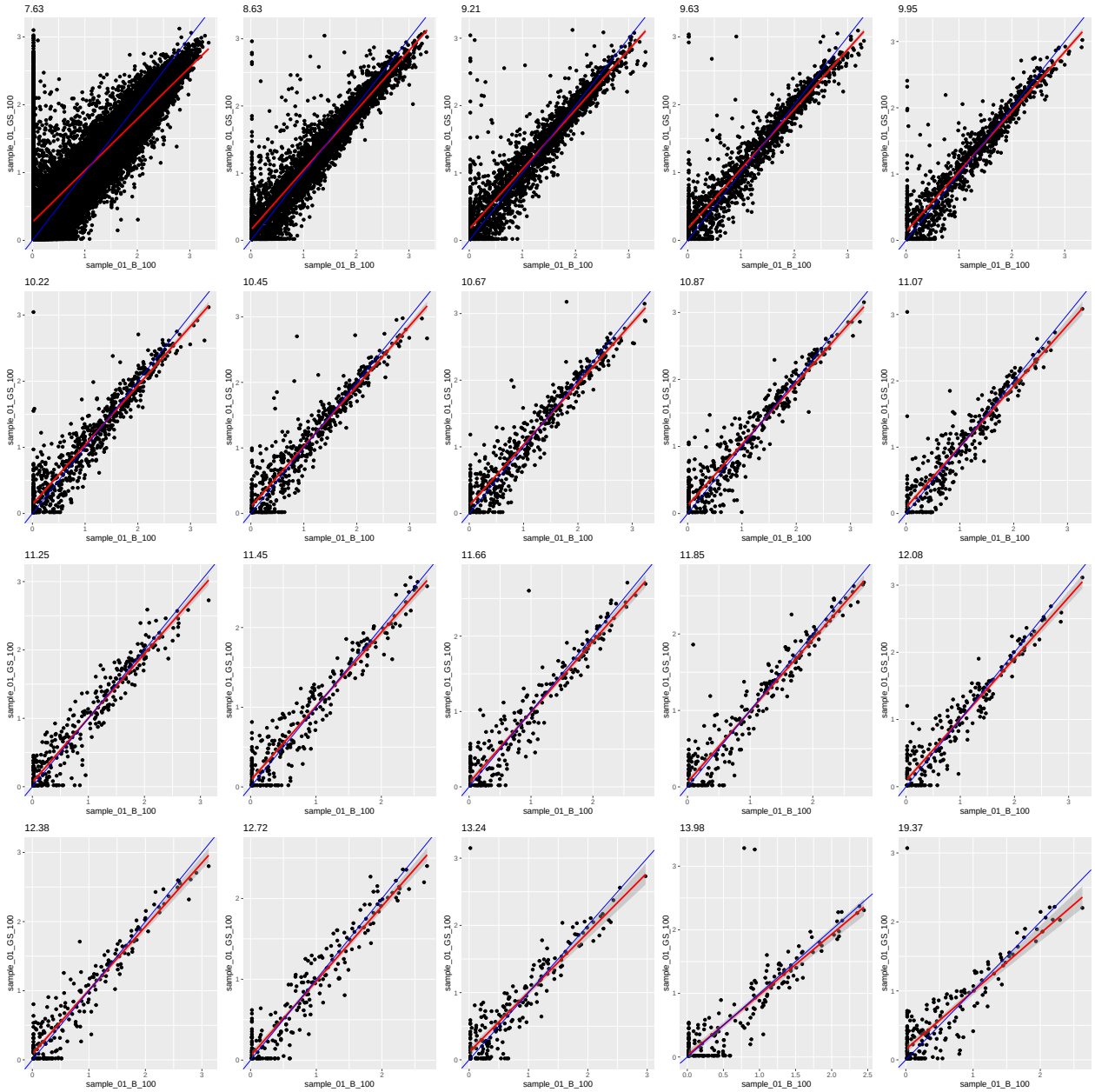
Plotting bias

```
pL <- plotSummary(se, counts_matrix[,1], summQuant="bias", nbreaks = 20)
ggarrange(plotlist = pL, nrow = 4, ncol = 5)
```



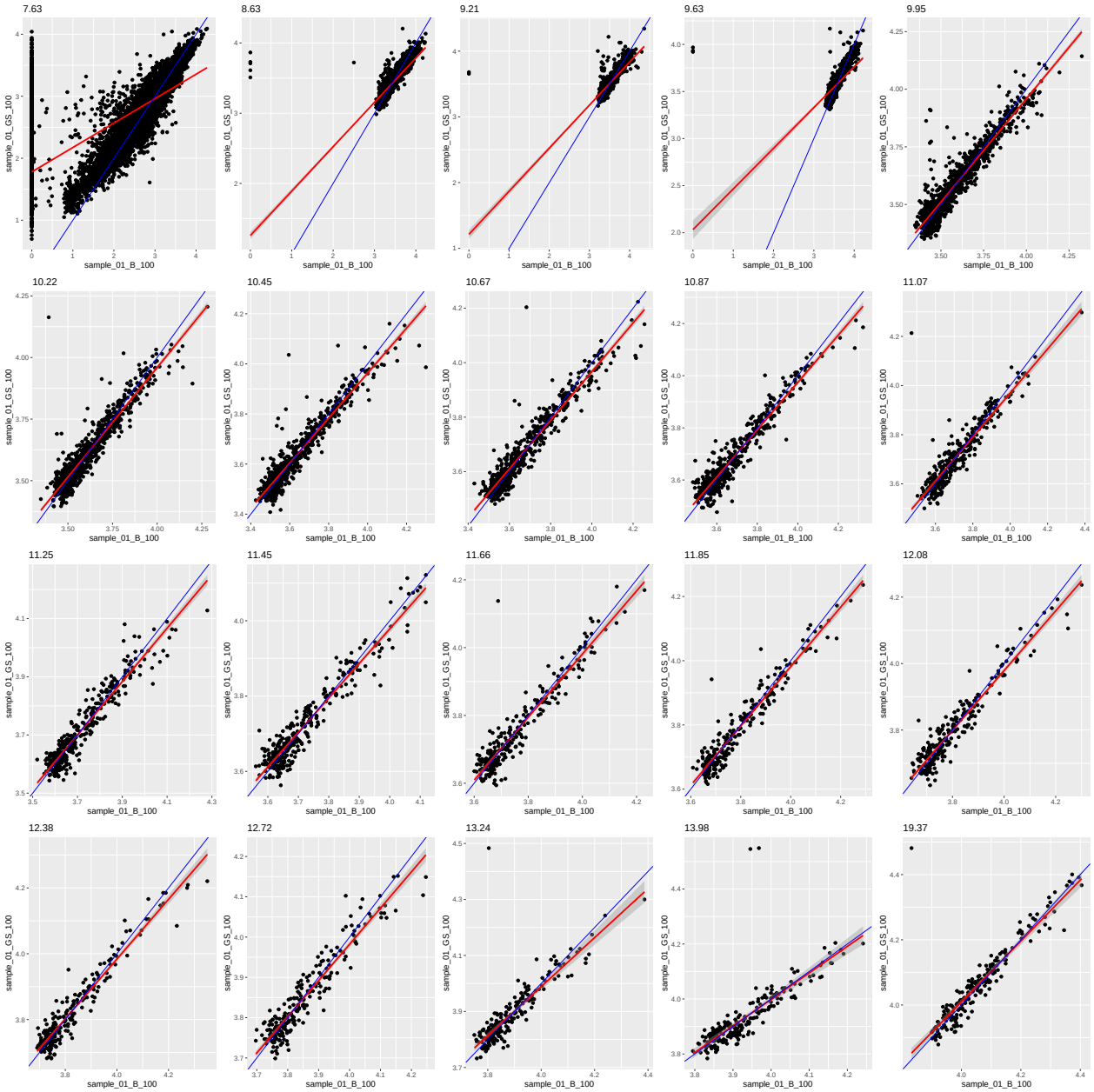
Plotting infRV

```
pL <- plotSummary(se, counts_matrix[,1], summQuant="infRV", nbreaks = 20)
ggarrange(plotlist = pL, nrow = 4, ncol = 5)
```



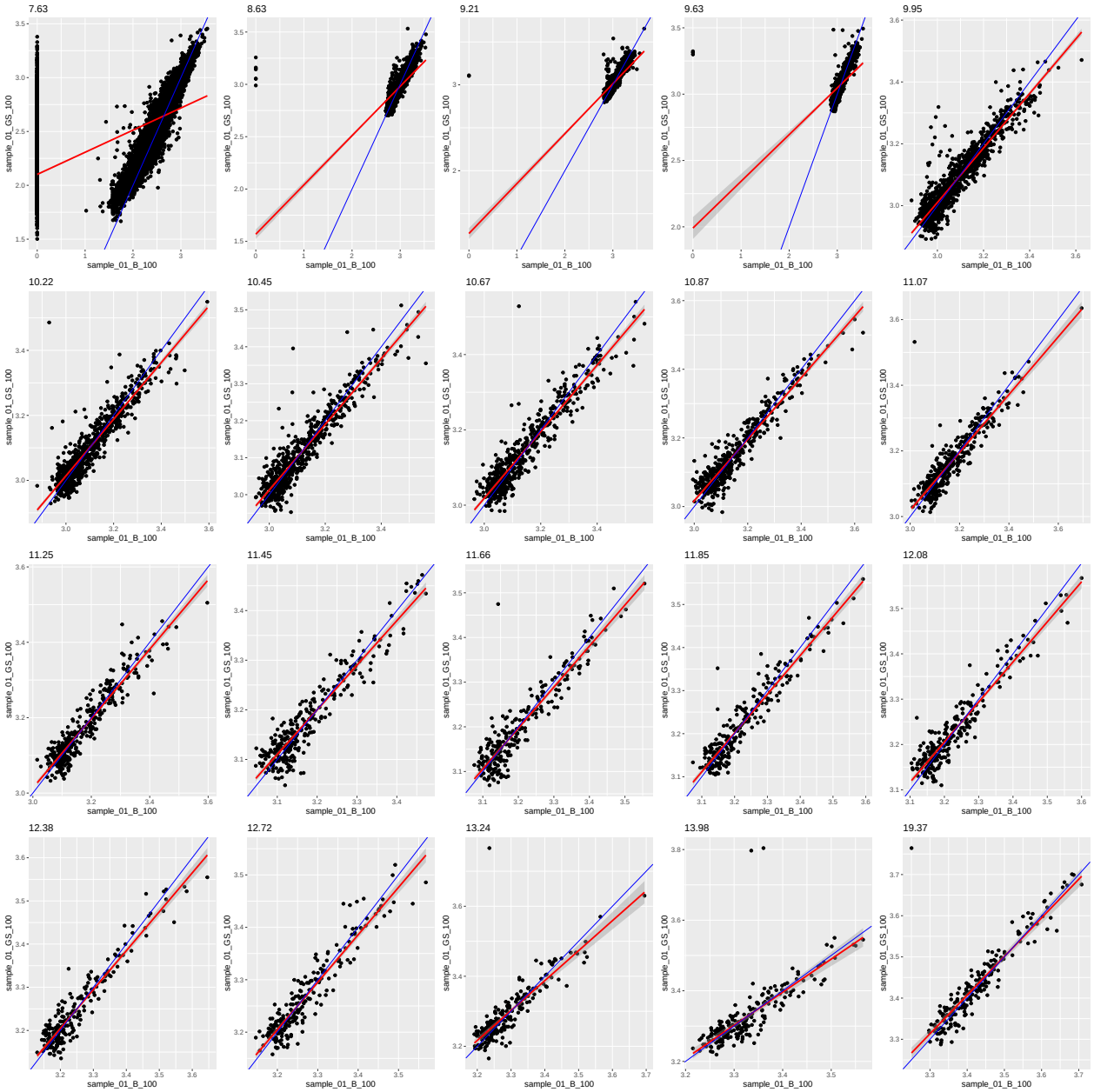
Plotting variance

```
pL <- plotSummary(se, counts_matrix[,1], summQuant="variance", nbreaks = 20)
ggarrange(plotlist = pL, nrow = 4, ncol = 5)
```



Plotting Width

```
pL <- plotSummary(se, counts_matrix[,1], summQuant="Width", nbreaks = 20)
ggarrange(plotlist = pL, nrow = 4, ncol = 5)
```


7.63
8.63
9.21
9.63
9.95
10.22
10.45
10.67
10.87
11.07
11.25
11.45
11.66
11.85
12.08
12.38
12.72
13.24
13.98
19.37
sample_01_GS_100
sample_01_B_100