

# Glioblastoma\_mixomics\_supervised

Loading required package: MASS

Loading required package: lattice

Loading required package: ggplot2

Loaded mixOmics 6.30.0

Thank you for using mixOmics!

Tutorials: <http://mixomics.org>

Bookdown vignette: <https://mixomicsteam.github.io/Bookdown>

Questions, issues: Follow the prompts at <http://mixomics.org/contact-us>

Cite us: citation('mixOmics')

```
load("~/PROJECTS_ALL/DATA_Glioblastoma/preprocessed/glioblastoma_data_no_caseid.RData")
```

```
X <- list(mRNA = glioblastoma_data$data.train$mrna,
          methylation = glioblastoma_data$data.train$methylation,
          protein = glioblastoma_data$data.train$protein
        )
Y <- glioblastoma_data$data.train$class
```

```
Y[is.na(Y)] <- 0  
Y <- factor(Y, levels = c(0, 1))  
print(Y)
```

```
design <- matrix(0.1, ncol = length(X), nrow = length(X),
                  dimnames = list(names(X), names(X)))
diag(design) <- 0
design
```

```
mRNA methylation protein
mRNA      0.0      0.1      0.1
methylation 0.1      0.0      0.1
protein    0.1      0.1      0.0
```

```
res1.pls.gbm <- pls(X$mRNA, X$protein, ncomp = 1)
cor(res1.pls.gbm$variates$X, res1.pls.gbm$variates$Y)
```

```
comp1
comp1 0.9464596
```

```
res2.pls.gbm <- pls(X$mRNA, X$methylation, ncomp = 1)
cor(res2.pls.gbm$variates$X, res2.pls.gbm$variates$Y)
```

```
comp1
comp1 0.4892087
```

```
res3.pls.gbm <- pls(X$protein, X$methylation, ncomp = 1)
cor(res3.pls.gbm$variates$X, res3.pls.gbm$variates$Y)
```

```
comp1
comp1 0.5068804
```

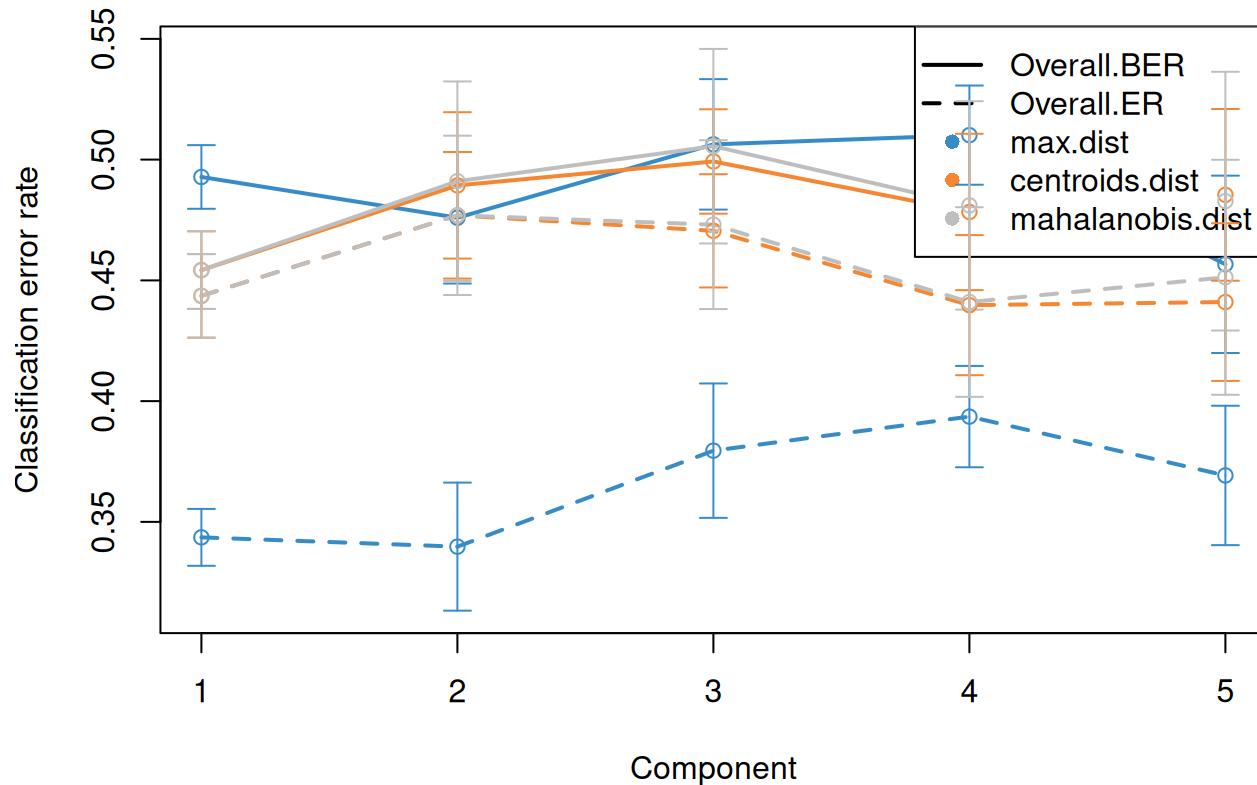
```
diablo.gbm <- block.plsda(X, Y, ncomp = 5, design = design)
```

```
Design matrix has changed to include Y; each block will be
linked to Y.
```

```
set.seed(123)
```

```
perf.diablo.gbm = perf(diablo.gbm, validation = 'Mfold', folds = 10, nrepeat = 10)
```

```
plot(perf.diablo.gbm)
```



```
perf.diablo.gbm$choice.ncomp$WeightedVote
```

	max.dist	centroids.dist	mahalanobis.dist
Overall.ER	1	1	1
Overall.BER	5	1	1

```
ncomp <- perf.diablo.gbm$choice.ncomp$WeightedVote["Overall.BER", "centroids.dist"]
```

```
set.seed(123)
```

```
test.keepX <- list(mRNA = c(5:9, seq(10, 25, 5)),
                     methylation = c(5:9, seq(10, 20, 2)),
                     proteomics = c(seq(5, 25, 5)))
```

```
tune.diablo.gbm <- tune.block.splsda(X, Y, ncomp = 2,
                                         test.keepX = test.keepX, design = design,
                                         validation = 'Mfold', folds = 10, nrepeat = 2,
                                         BPPARAM = BiocParallel::SnowParam(workers = 2),
                                         dist = "centroids.dist")
```

Design matrix has changed to include Y; each block will be linked to Y.

You have provided a sequence of keepX of length: 9 for block mRNA and 11 for block methylation and 5 for block proteomics.

This results in 495 models being fitted for each component and each nrepeat, this may take some time to run, be patient!

```
list.keepX <- tune.diablo.gbm$choice.keepX
```

```
list.keepX
```

```
$mRNA
```

```
[1] 5 8
```

```
$methylation
```

```
[1] 6 8
```

```
$protein
```

```
[1] 5 5
```

```
diablo.gbm <- block.splsda(X, Y, ncomp = 2,  
                           keepX = list.keepX, design = design)
```

Design matrix has changed to include Y; each block will be  
linked to Y.

```
diablo.gbm$design
```

	mRNA	methylation	protein	Y
mRNA	0.0	0.1	0.1	1
methylation	0.1	0.0	0.1	1
protein	0.1	0.1	0.0	1
Y	1.0	1.0	1.0	0

```
# mRNA variables selected on component 1  
selectVar(diablo.gbm, block = 'mRNA', comp = 1)
```

```
$mRNA  
$mRNA$name  
[1] "ENSG00000183034.13" "ENSG00000176209.11" "ENSG00000172794.20"  
[4] "ENSG00000140993.11" "ENSG00000006283.18"
```

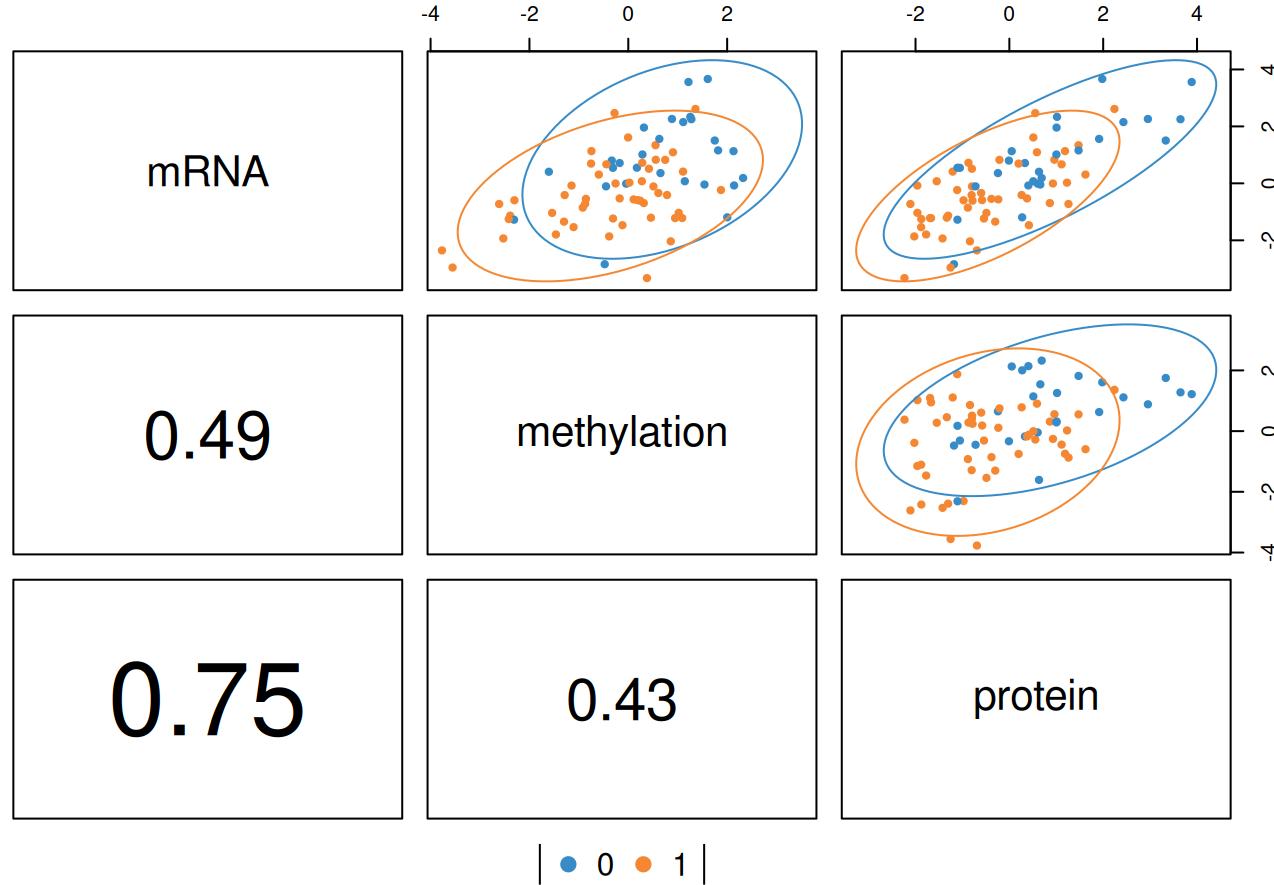
```
$mRNA$value  
          value.var  
ENSG00000183034.13  0.59703579  
ENSG00000176209.11 -0.57171728  
ENSG00000172794.20 -0.53451506  
ENSG00000140993.11  0.16966791  
ENSG00000006283.18 -0.04684085
```

```
$comp  
[1] 1
```

## Plotting

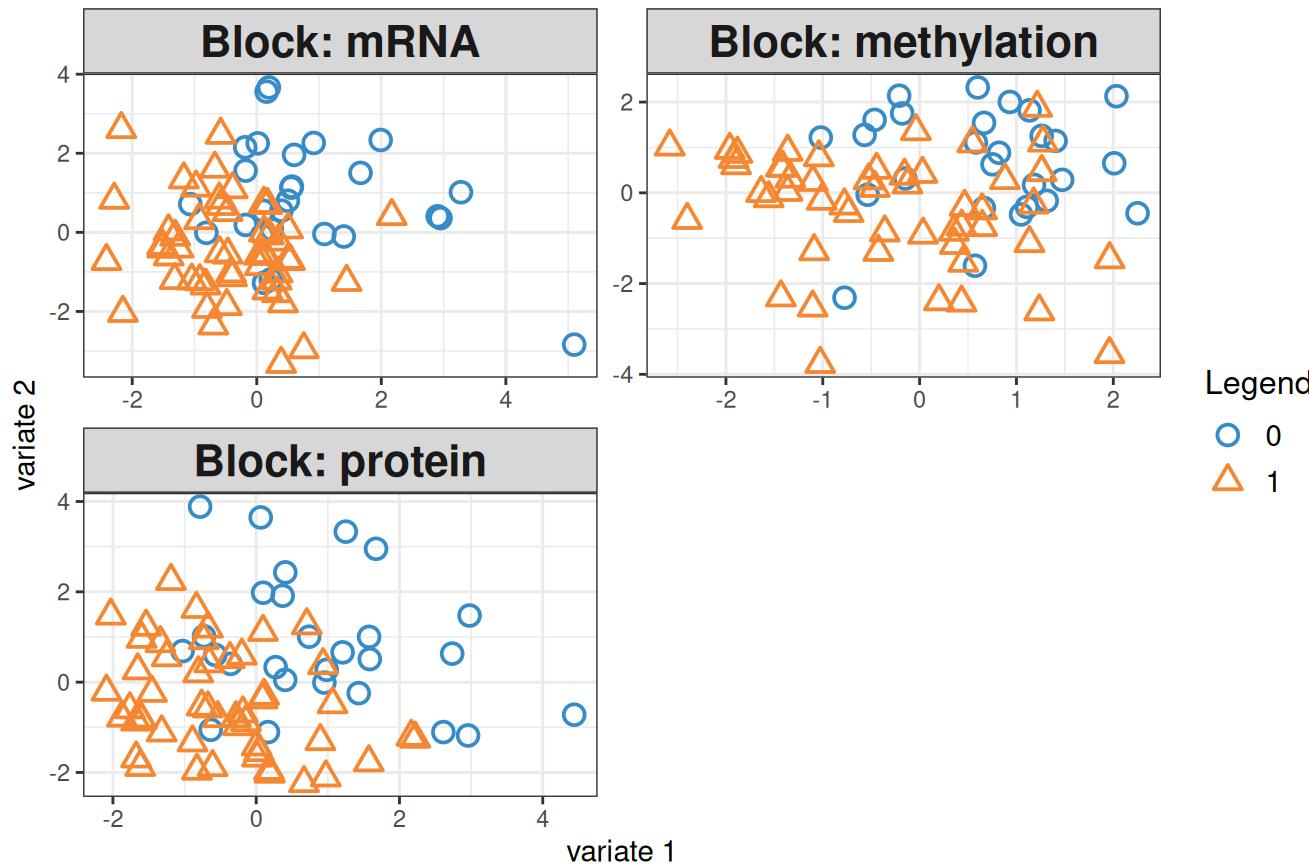
`plotDiablo()` is a diagnostic plot to check whether the correlations between components from each data set were maximised as specified in the design matrix. We specify the dimension to be assessed with the `ncomp` argument

```
plotDiablo(diablo.gbm, ncomp = 2)
```

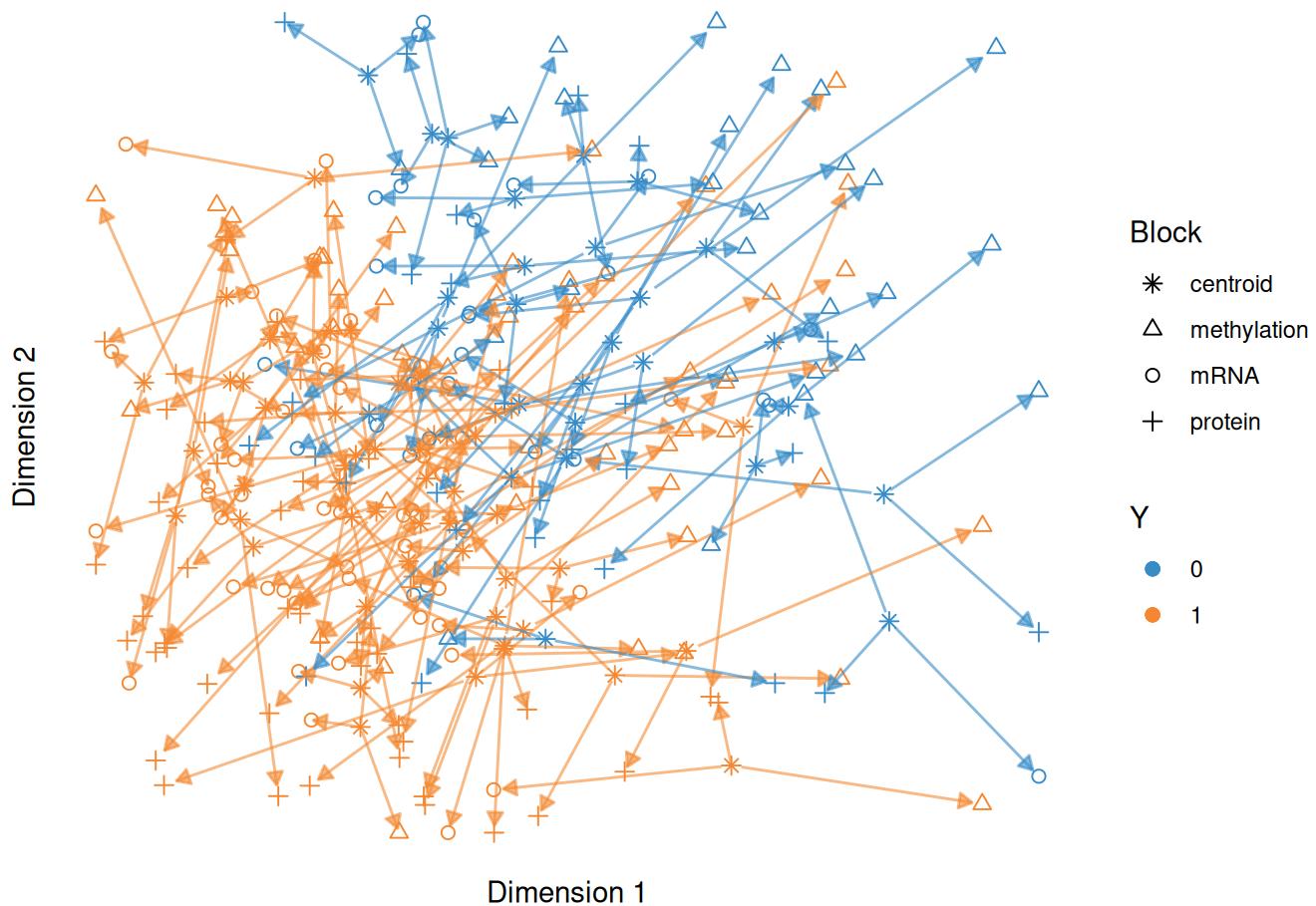


```
plotIndiv(diablo.gbm, ind.names = FALSE, legend = TRUE,  
          title = 'Glioblastoma, DIABLO comp 1 - 2')
```

## Glioblastoma, DIABLO comp 1 - 2

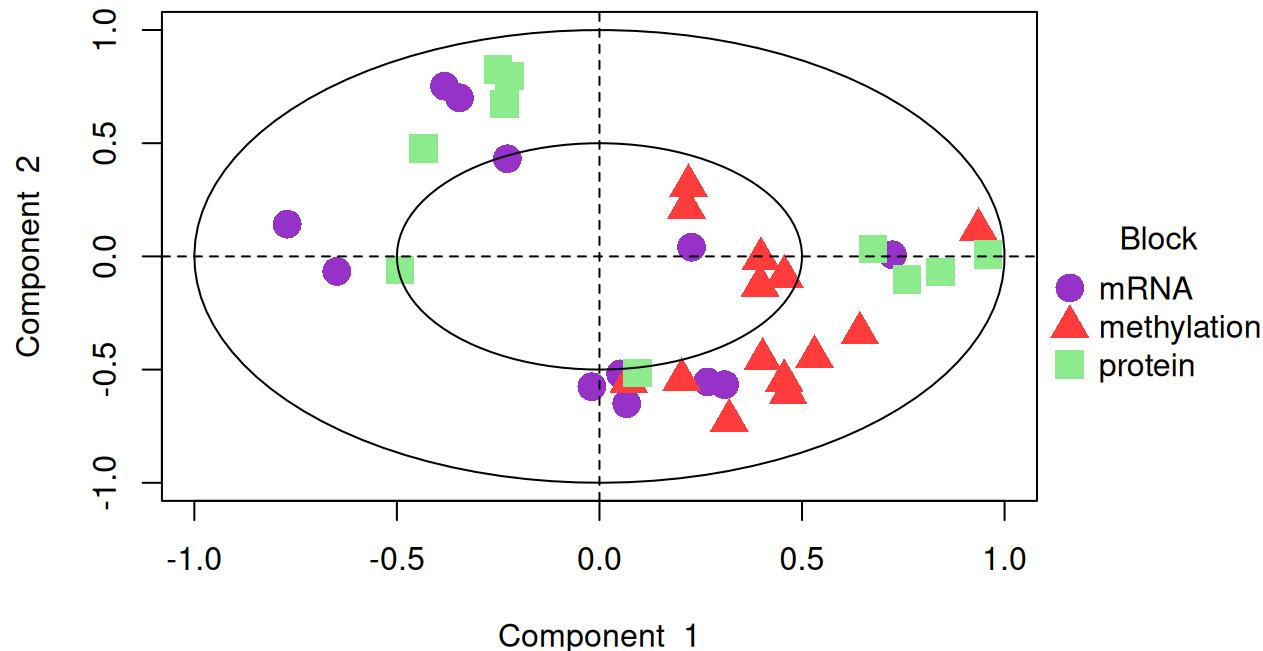


```
plotArrow(diabolo.gbm, ind.names = FALSE, legend = TRUE,  
          title = 'Glioblastoma, DIABLO comp 1 - 2')
```



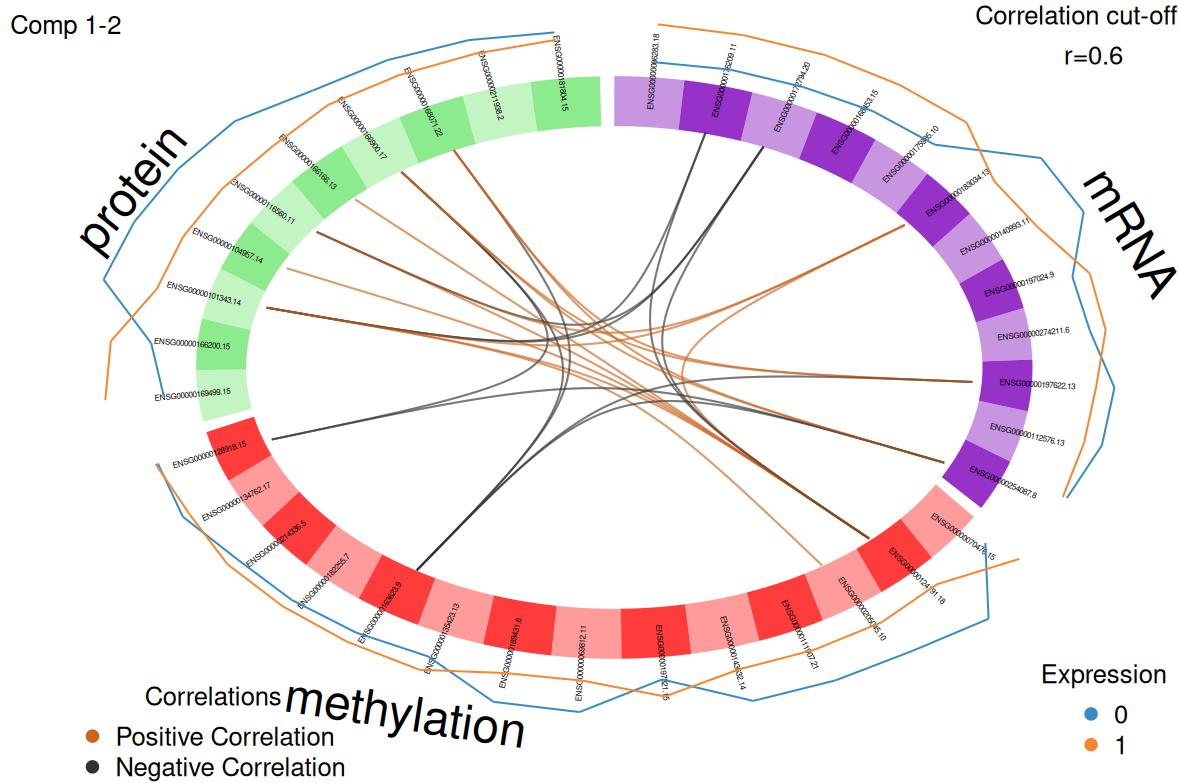
```
plotVar(diabolo.gbm, var.names = FALSE, style = 'graphics', legend = TRUE,
        pch = c(16, 17, 15), cex = c(2,2,2),
        col = c('darkorchid', 'brown1', 'lightgreen'),
        title = 'Glioblastoma, DIABLO comp 1 - 2')
```

## Glioblastoma, DIABLO comp 1 - 2



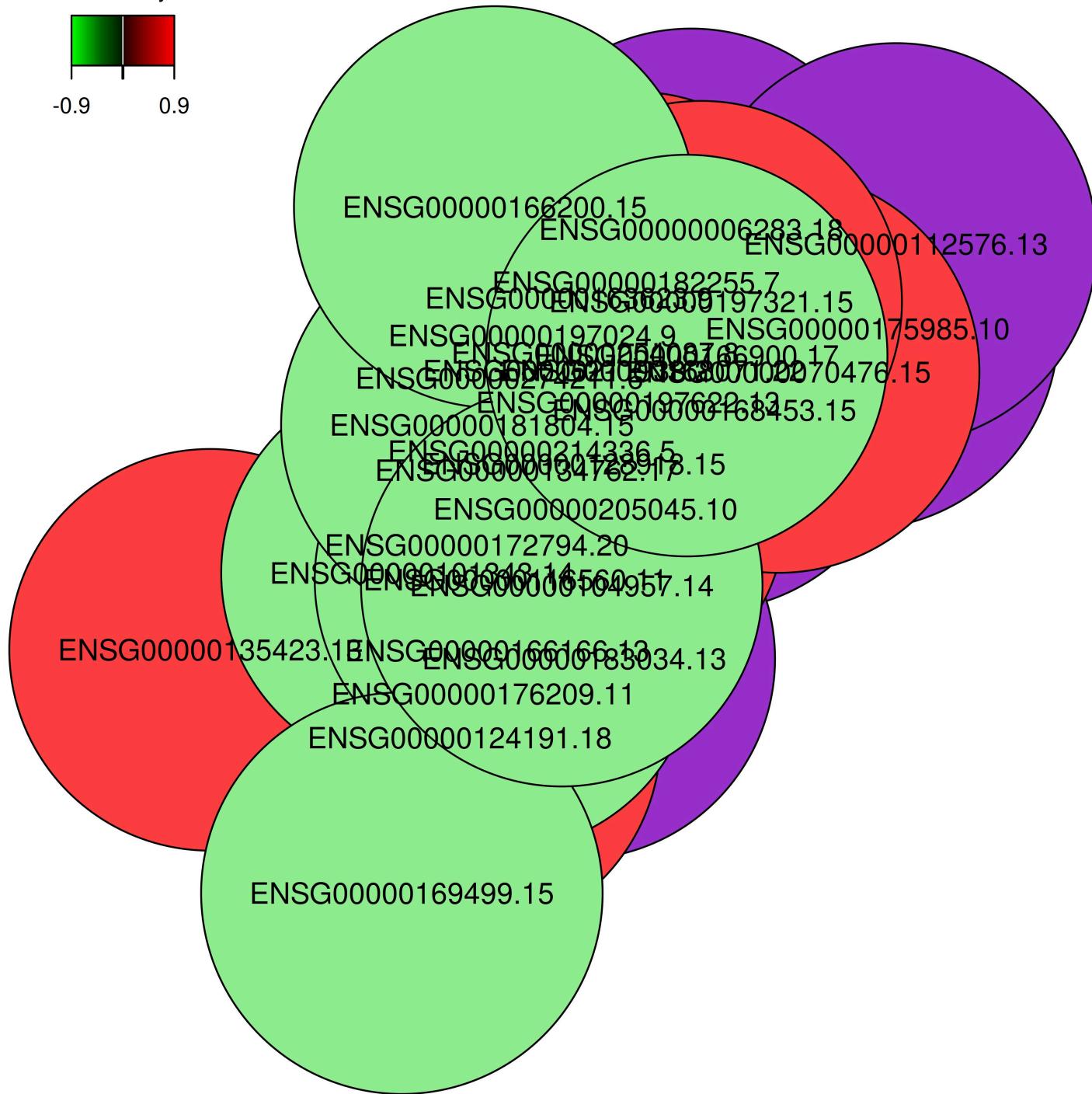
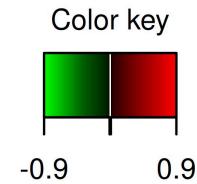
**Correlation circle plot from multiblock sPLS-DA performed on the [glioblastoma](#) data.** The variable coordinates are defined according to their correlation with the first and second components for each data set. Variable types are indicated with different symbols and colours, and are overlaid on the same plot. The plot highlights the potential associations within and between different variable types when they are important in defining their own component.

```
circosPlot(diabolo.gbm, cutoff = 0.6, line = TRUE,  
          color.blocks = c('darkorchid', 'brown1', 'lightgreen'),  
          color.cor = c("chocolate3", "grey20"), size.labels = 1.5)
```



**Circos plot from multiblock sPLS-DA performed on the glioblastoma data.** The plot represents the correlations greater than 0.6 between variables of different types, represented on the side quadrants. The internal connecting lines show the positive (negative) correlations. The outer lines show the expression levels of each variable in each sample group.

```
network(diablo.gbm, blocks = c(1,2,3),
        cutoff = 0.4,
        color.node = c('darkorchid', 'brown1', 'lightgreen'),
        save = 'png', name.save = 'diablo-network'
      )
```



**Relevance network for the variables selected by multiblock sPLS-DA performed on the glioblastoma data on component 1.** Each node represents a selected variable with colours indicating their type. The colour of the edges represent positive or negative correlations.

## Model performance and prediction

```
set.seed(123)
```

```
perf.diabolo.gbm <- perf(diabolo.gbm, validation = 'Mfold', folds = 10,  
                           nrepeat = 10, dist = 'centroids.dist')
```

```
# Performance with Majority vote  
perf.diabolo.gbm$MajorityVote.error.rate
```

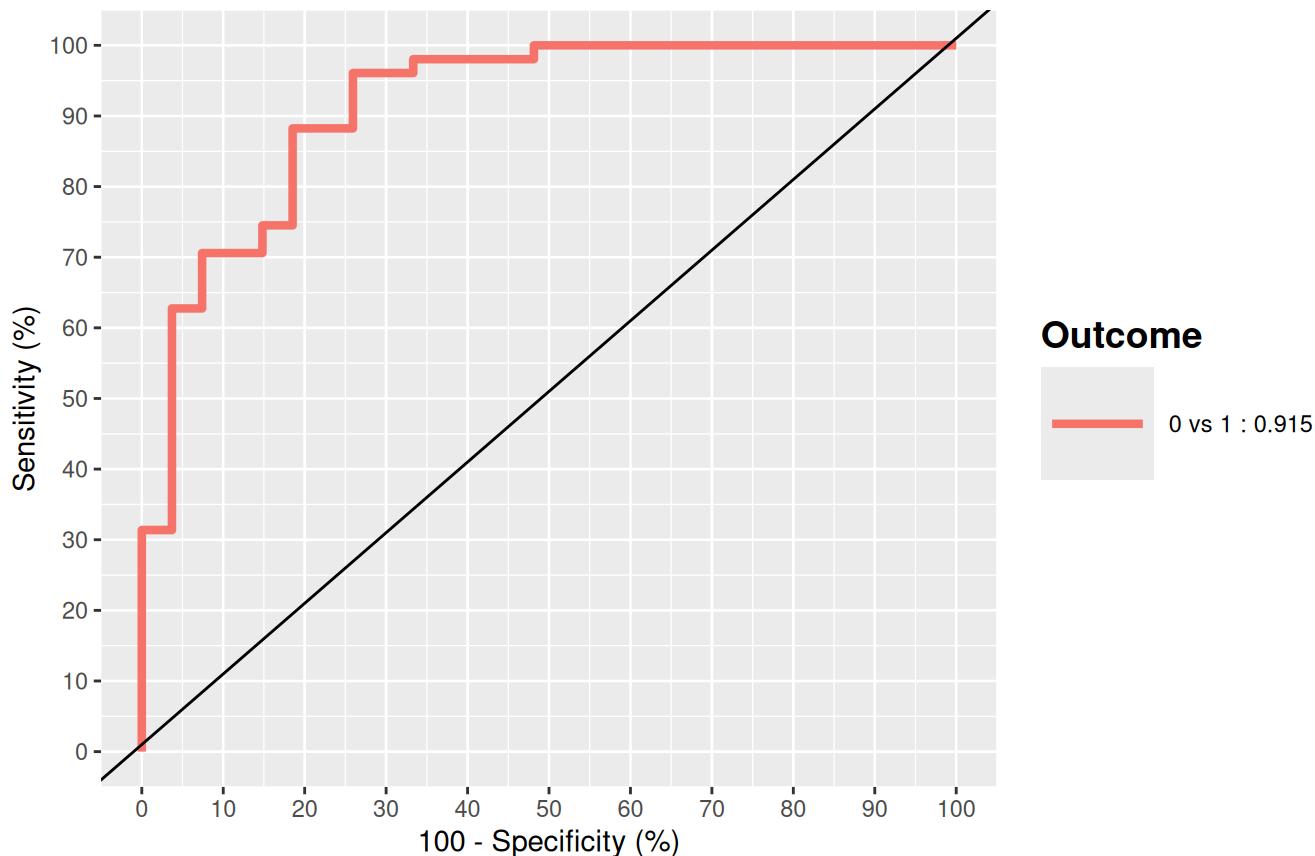
```
$centroids.dist  
      comp1      comp2  
0     0.6222222 0.5851852  
1     0.3882353 0.3450980  
Overall.ER 0.4692308 0.4282051  
Overall.BER 0.5052288 0.4651416
```

```
# Performance with Weighted vote  
perf.diabolo.gbm$WeightedVote.error.rate
```

```
$centroids.dist  
      comp1      comp2  
0     0.6222222 0.5851852  
1     0.3882353 0.3450980  
Overall.ER 0.4692308 0.4282051  
Overall.BER 0.5052288 0.4651416
```

```
auc.diabolo.gbm <- auroc(diabolo.gbm, roc.block = "protein", roc.comp = 2,  
                           print = TRUE)
```

### ROC Curve Block: protein, Using Comp(s): 1, 2



#### Outcome

— 0 vs 1 : 0.915

\$mRNA  
\$mRNA\$comp1  
AUC p-value  
0 vs 1 0.7887 2.981e-05

\$mRNA\$comp2  
AUC p-value  
0 vs 1 0.8802 3.835e-08

\$methylation  
\$methylation\$comp1

```
AUC      p-value  
0 vs 1 0.764 0.0001346
```

```
$methylation$comp2  
      AUC      p-value  
0 vs 1 0.8722 7.335e-08
```

```
$protein  
$protein$comp1  
      AUC      p-value  
0 vs 1 0.8134 5.842e-06
```

```
$protein$comp2  
      AUC      p-value  
0 vs 1 0.915 1.944e-09
```

```
# Prepare test set data: here one block (proteins) is missing  
data.test.gbm <- list(mRNA = glioblastoma_data$data.test$mrna,  
                      protein = glioblastoma_data$data.test$protein)  
  
predict.diablo.gbm <- predict(diablo.gbm, newdata = data.test.gbm)
```

Warning in predict.block.spls(diablo.gbm, newdata = data.test.gbm): Some blocks are missing in 'newdata'; the prediction is based on the following blocks only:  
mRNA, protein

```
confusion.mat.gbm <- get.confusion_matrix(truth = glioblastoma_data$data.test$class,  
                                             predicted = predict.diablo.gbm$WeightedVote$centroids.dist[,2])  
confusion.mat.gbm
```

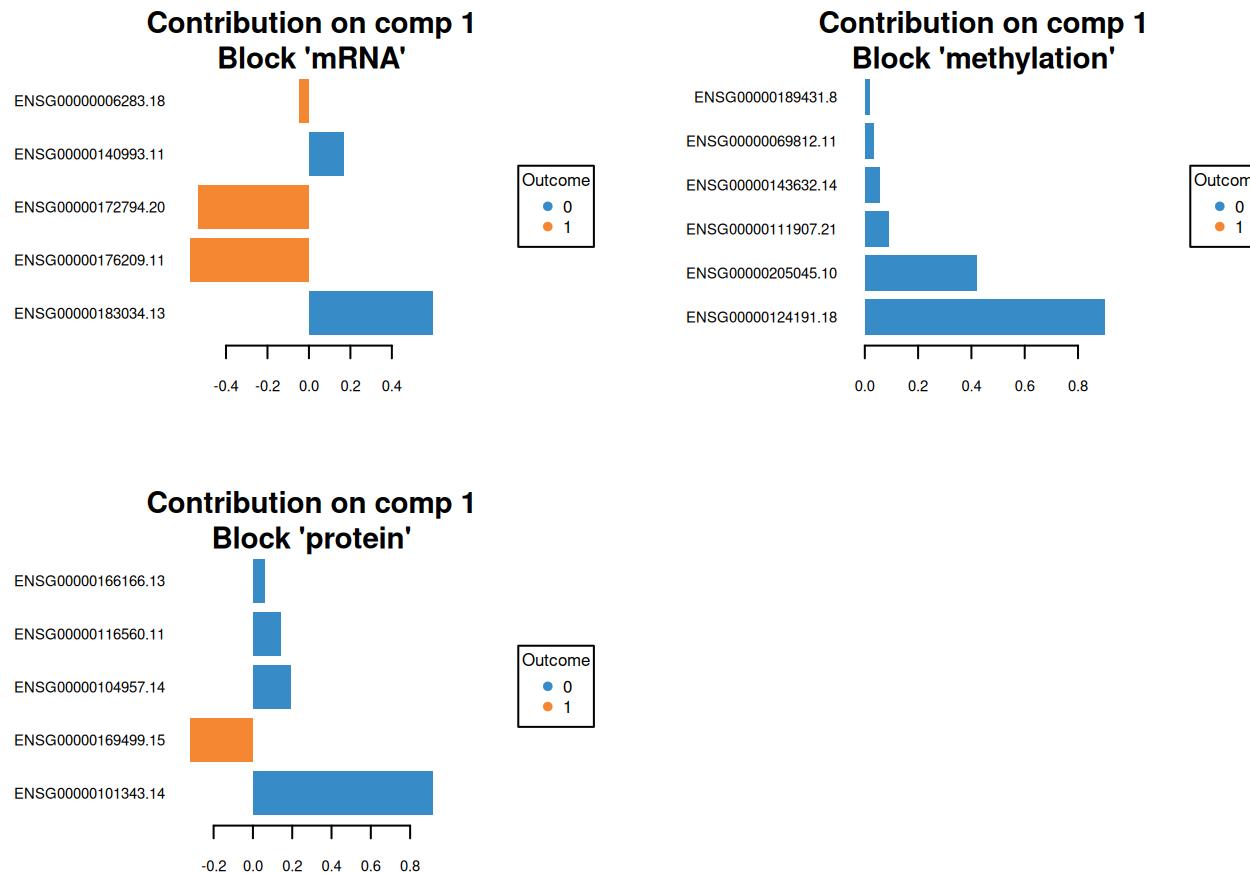
	predicted.as.0	predicted.as.1
0	2	1
1	6	10

```
get.BER(confusion.mat.gbm)
```

```
[1] 0.3541667
```

`plotLoadings()` visualises the loading weights of each selected variable on each component and each data set. The colour indicates the class in which the variable has the maximum level of expression (`contrib = 'max'`) or minimum (`contrib = 'min'`), on average (`method = 'mean'`) or using the median (`method = 'median'`)

```
plotLoadings(diablo.gbm, comp = 1, contrib = 'max', method = 'median')
```



```
set.seed(123)
```

```
perf.diabolo.gbm <- perf(diabolo.gbm, validation = 'Mfold', folds = 10,
                           nrepeat = 10, dist = 'centroids.dist')
```

```
# Performance with Majority vote
perf.diabolo.gbm$MajorityVote.error.rate
```

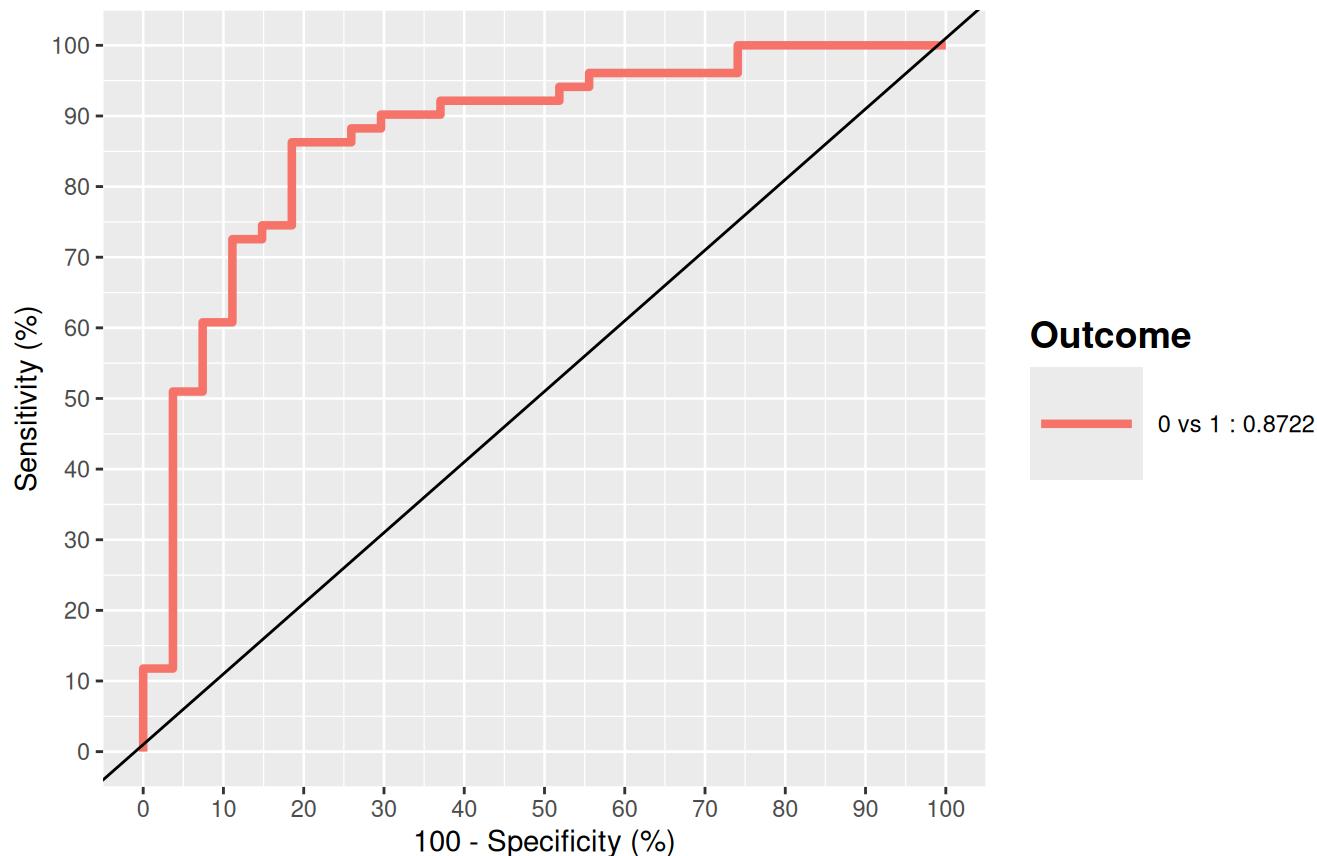
```
$centroids.dist
      comp1     comp2
0 0.6222222 0.5851852
1 0.3882353 0.3450980
Overall.ER 0.4692308 0.4282051
Overall.BER 0.5052288 0.4651416
```

```
# Performance with Weighted vote
perf.diabolo.gbm$WeightedVote.error.rate
```

```
$centroids.dist
      comp1     comp2
0 0.6222222 0.5851852
1 0.3882353 0.3450980
Overall.ER 0.4692308 0.4282051
Overall.BER 0.5052288 0.4651416
```

```
auc.diabolo.gbm <- auroc(diabolo.gbm, roc.block = "methylation", roc.comp = 2,
                            print = TRUE)
```

**ROC Curve**  
**Block: methylation, Using Comp(s): 1, 2**



\$mRNA  
\$mRNA\$comp1  
AUC p-value  
0 vs 1 0.7887 2.981e-05

\$mRNA\$comp2  
AUC p-value  
0 vs 1 0.8802 3.835e-08

\$methylation  
\$methylation\$comp1

```
      AUC      p-value
0 vs 1 0.764 0.0001346
```

```
$methylation$comp2
      AUC      p-value
0 vs 1 0.8722 7.335e-08
```

```
$protein
$protein$comp1
      AUC      p-value
0 vs 1 0.8134 5.842e-06
```

```
$protein$comp2
      AUC      p-value
0 vs 1 0.915 1.944e-09
```

```
data.test.gbm <- list(protein = glioblastoma_data$data.test$protein,
                      methylation = glioblastoma_data$data.test$methylation)

predict.diablo.gbm <- predict(diablo.gbm, newdata = data.test.gbm)
```

Warning in predict.block.spls(diablo.gbm, newdata = data.test.gbm): Some blocks are missing in 'newdata'; the prediction is based on the following blocks only: methylation, protein

```
confusion.mat.gbm <- get.confusion_matrix(truth = glioblastoma_data$data.test$class,
                                              predicted = predict.diablo.gbm$WeightedVote$centroids.dist[,2])
confusion.mat.gbm
```

```
predicted.as.0 predicted.as.1
0              2                  1
1              6                 10
```

```
get.BER(confusion.mat.gbm)
```

[1] 0.3541667