Hypoxia Classifier Tutorial

Laura Puente-Santamaria

Step 1: Load decision trees and gene expression matrix.

```
load("Tree_colection.RData")
load("GeneExpression_example.RData")

head(gene_expression)
```

```
##            [,1]
## A1BG      1.000
## A1BG-AS1  5.635
## A1CF      2.000
## A2M      45.972
## A2M-AS1   5.000
## A2ML1     2.243
```

This gene expression matrix is the salmon output corresponding to GSM2390150, an RNA-seq of HUVEC cells grown in hypoxia for 8h.

Step 2: Ranking percentile.

```
rank_percentile <- matrix( 100*rank( gene_expression ) / length( gene_expression ) )

rownames( rank_percentile ) <- rownames( gene_expression )
colnames( rank_percentile ) <- "Sample 1"

head(rank_percentile)
```

```
##          Sample 1
## A1BG     39.94070
## A1BG-AS1 50.75651
## A1CF     44.06828
## A2M      65.71069
## A2M-AS1  50.03055
## A2ML1    45.44474
```

We rank genes from the least to the most expressed, with 100 being the most expressed gene in the sample, and 0 the least. The trees generated with `rpart` take as input a data frame with a row for samples and columns for variables:

```
rank_percentile <- data.frame( t( rank_percentile ) )

rank_percentile[,1:5]
```

```
##             A1BG A1BG.AS1     A1CF      A2M  A2M.AS1
## Sample 1 39.9407 50.75651 44.06828 65.71069 50.03055
```

Step 3: Classify the sample.

```
decisionTree <- fullTreeCollection[[ 125 ]]

prediction <- predict( decisionTree, rank_percentile )

prediction
```

```
##                    N         H
## Sample 1 0.07352941 0.9264706
```

The resulting matrix, `prediction`, has two columns with the probabilities of the sample to be normoxic (first column) or hypoxic (second column).