Breast cancer survival analysis with R

Statistics M1

Raphaël Bonnet (PhD student C3M-U1065, Inserm, UCA)

03 décembre 2019

Introduction notes

The curatedBreastData package contains 34 high-quality GEO gene expression microarray datasets, all with advanced breast cancer represented as S4 ExpressionObjects with attached clinical data (phenoData) for easier data analyses in R.

You will learn to explore this type of objects in R and to extract genetic and clinical feature.

We will overlook the Cox regression model for survival analysis and explore the relevance of most of the clinical covariates toward survival time.

curatedBreastData package: Access & explore the data

These datasets all contain at least one survival and/or treatment response variable, and minimal treamtent information (such as whether they had chemotherapy or not.) Clinical variables were semantically normalized across all datasets to provide a powerful database for investigating genes that are related to clinical variables such as pathological complete response, ER and HER2 IHC pathology tests, pam50 subtyping tests (when available), and tumor stage. This database was original designed as a MySQL database, but has been re-represented as S4 ExpressionObjects for easier data analyses in R.

Loading packages

```
suppressMessages(library(Biobase))
suppressMessages(library(limma))
suppressMessages(library(survival))
suppressMessages(library(survminer))
suppressMessages(library("curatedBreastData"))
suppressMessages(library(survivalROC))
suppressMessages(library(tibble))
suppressMessages(library(purrr))
suppressMessages(library(tidyr))
suppressMessages(library(dplyr))
suppressMessages(library(FactoMineR))
```

Loading data

```
#load up datasets that are in S4 expressionSet format.
#clinical data from master clinicalTable already linked to each sample
#in these ExpressionSets in the phenoData slot.

data(curatedBreastDataExprSetList)
data(clinicalData)
```

First layer: the list of ExpressionSets

using length(), head(), names() on list.eset

```
#renaming for easier handling

list.eset=curatedBreastDataExprSetList
```

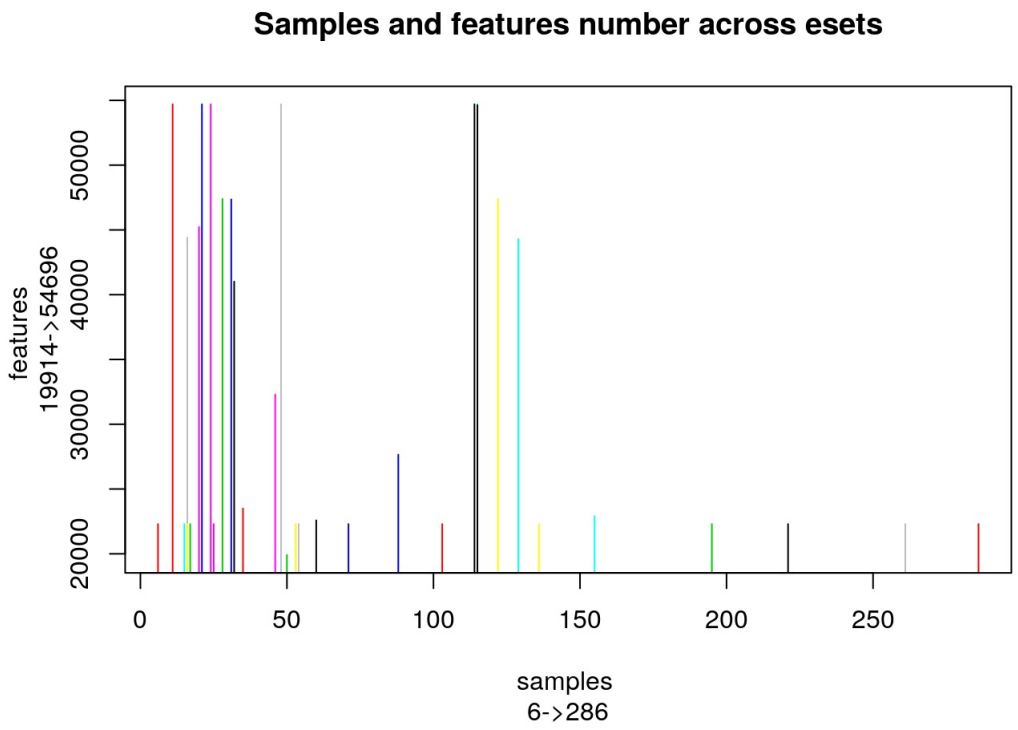
```
# How big is loaded object
length(list.eset)
```

```
## [1] 34
```

```
# what's inside ?
# first 5 elements
head(names(list.eset))
```

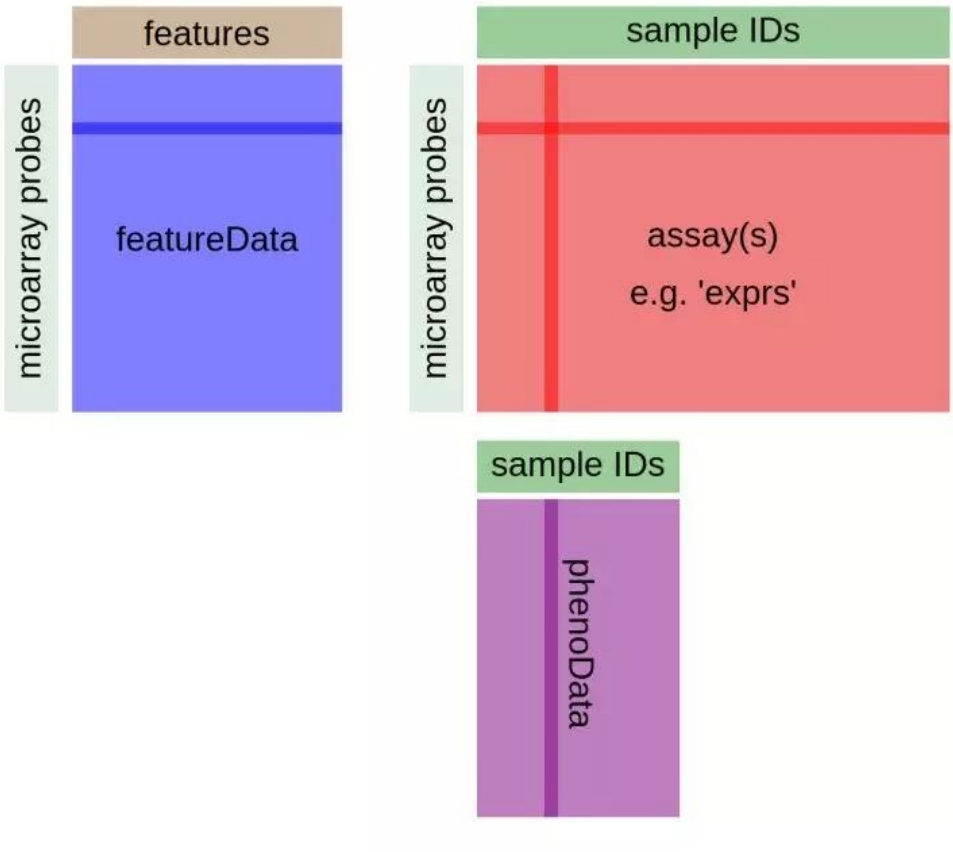
```
## [1] "study_1379_GPL1223_all"  "study_2034_GPL96_all"
## [3] "study_4913_GPL3558_all"  "study_6577_GPL3883_all"
## [5] "study_9893_GPL5049_all"  "study_12071_GPL5186_all"
```

```
#What are the dimensions of each eset (samples and genes)
dims=c()
for (l in seq_along(curatedBreastDataExprSetList))
{
  dims=rbind(dims,dim(curatedBreastDataExprSetList[[l]]))
}
plot(dims[,2],dims[,1],type = 'h',
     xlab = c('samples ', paste(range(dims[,2]),collapse = '->'),sep=''),
     ylab= c('features ',paste(range(dims[,1]),collapse = '->'),sep=''),
     col=c(1:34),
     main="Samples and features number across esets")
```



Exploring the ExpressionSet

-ExpressionSet Structure summary



ExpressionSet is an object of type S4 that can be explored using the accessor “@” (S4 itself) and “$” (elements within S4)

exprs()

- Use class(), `eset@assayData$exprs` or exprs() to access gene expression data

```
# lets look inside one object
eset=list.eset[[4]]
```

```
#what types are eset and eset@assayData$exprs

class(eset)
```

```
## [1] "ExpressionSet"
## attr(,"package")
## [1] "Biobase"
```

```
# class is ExpressionSet, this class has a package attribute called Biobase

class(eset@assayData$exprs)
```

```
## [1] "matrix"
```

```
#have a look into exprs(eset)
#USE head()

head(exprs(eset))
```

```
##      152028    152029    152030   152031    152032    152033   152034
## 1  9.953611 10.450180  9.465110 10.14900  9.386305  9.598014 10.69238
## 2  9.934310 10.479716  9.559373 10.16784  9.433689  9.707053 10.78098
## 3 10.712922 10.718653 10.520253 10.48239  8.594126 10.314322 12.03656
```

```
## 4 11.861188  8.788562 10.617397 10.63950 10.232048 10.530069 11.10053
## 5 10.400636  8.926569 10.596201 10.25423 10.015555 10.031665 10.84187
## 6 11.093413 10.096191 11.160600 11.58466 10.371817 10.798198 11.31962
##     152035    152036    152037    152038   152039    152040   152041
## 1 10.19648  9.935644  9.748017  9.636818 10.40652  9.528447 10.30374
## 2 10.32133  9.990808  9.687410  9.616142 10.53395  9.509889 10.31078
## 3 10.28948 10.926632 10.292262 10.413837 10.83664 10.364416 10.67495
## 4 10.78236  9.187585 10.369203 10.594165 11.93863  9.870054 10.60681
## 5 10.29162 10.598828 10.166026 10.266743 11.36796  9.791813 10.69883
## 6 10.33274 10.418934 11.026962 10.737262 10.45390 11.566464 11.15198
##      152042    152043    152044    152045    152046    152047    152048
## 1  9.847928 10.160090  9.712169  9.708846 10.775792 10.201279  9.411080
## 2  9.859449 10.371794  9.724947  9.647046 10.753147 10.246796  9.759292
## 3 10.754558 10.137377 10.343589 11.164833 10.367330 10.095286  8.210500
## 4 10.777232 10.144277 10.236209 11.010238  8.406527  8.693675 10.234862
## 5 10.624990  9.964792 10.322001 10.372585  9.549729  8.963252  8.969184
## 6 11.005129 10.815463 10.314406 11.020372 11.154735 10.620568  8.879108
##      152049    152050    152051    152052    152053    152054   152055
## 1 10.175124 10.044900  9.705605  9.830732  9.705662  9.589174 10.14679
## 2 10.236488 10.035691  9.651791  9.754832  9.843567  9.708849 10.15117
## 3 10.720795 10.259731 10.587721 10.113029 10.641073 10.542908 11.81921
## 4 10.576220  9.970985 10.454030 10.061656 10.638832 10.365497 10.95642
## 5  9.885195 10.016022 10.348919 10.246489 10.123728 10.442086 10.96833
## 6 10.785745 10.950725 10.300724  9.411066 10.414675 11.528840 11.35218
##      152056   152057   152058    152059   152060   152061    152062
## 1  9.725556 10.14810 10.04069 10.780137 10.89906 10.85119  9.239151
## 2  9.737143 10.23746 10.02065 10.762689 11.11427 10.84773  9.209983
## 3 10.414483 10.62086 10.81563 10.565913 11.10292 11.32387 10.536872
## 4 10.911059 11.77844 11.14650 10.146804 10.99419 11.44078 10.449898
## 5  9.818053 11.83735 10.97293  9.121957 10.05579  9.28983 11.151726
## 6 11.225544 10.52527 11.48913 11.464448 11.79581 11.66116 11.165504
##      152063    152064   152065    152066   152067   152068   152069
## 1  9.719518  9.504052 10.16310 10.039550 10.11495 10.10396 10.02540
## 2  9.748349  9.235641 10.21776 10.080735 10.17168 10.17707 10.00874
## 3 10.822976  9.955114 10.23521 10.116603 10.38674 10.36665 10.27000
## 4 10.708419 10.231347 10.28571 10.633948 10.40593 10.49045 10.40066
## 5 10.556962  9.714672 10.29565  9.714579 10.27404 10.16518 10.26646
## 6 10.349264 10.570613 11.81418 11.080938 10.55624 10.51808 10.65991
##      152070    152071   152072    152073    152074    152075   152076
## 1  9.978830 10.157111 10.14763  9.584361  8.954905  9.595038 10.32186
## 2 10.040602 10.030030 10.20870  9.648138  9.006531  9.568741 10.39940
## 3 10.224928  9.648158 10.35237 10.220938 10.220499 10.734123 11.46906
## 4  8.208621 10.130744 10.27391 10.621146 10.431599 10.685090 11.16625
## 5  9.464239 10.039497 10.32968 10.236734 10.168723 10.330937 10.80133
## 6 11.094810 11.023849 10.97240 11.180782 11.234495 10.945355 10.99306
##      152077   152078   152079   152080    152081    152082   152083
## 1 10.032914 10.37205 10.03795 10.18206 10.053715 10.135747 10.82656
## 2  9.868808 10.29791 10.05348 10.07777 10.074486 10.122049 10.78328
## 3 10.411163 10.30217 10.72153 10.27395  9.161728 10.106616 10.08857
## 4 10.661037 10.30094 10.64684 10.67480 10.637117 10.078958 10.00184
## 5 10.331354 10.38031 10.30278 10.07621 10.198461 10.090329 10.17927
## 6 10.661989 11.12309 10.60903 10.00057 10.596726  9.072293 10.10279
##      152084    152085   152086    152087    152088    152089   152090
## 1  9.335922 10.084065  9.98834  9.356629  9.631435  9.994526 10.29860
## 2  9.301662 10.143204 10.03098  9.349417  9.681853  9.996163 10.02355
## 3 11.691866 10.359756 10.21911 10.271318  9.363517 10.330239 11.15216
## 4 10.766909  9.079344 10.13570  9.181872 10.602983 10.491360 10.44181
## 5 10.059003 10.453651 10.51156 10.500885 10.153462  9.826637 10.17788
## 6 10.358924 11.043915 11.14777 10.829639 10.861443  9.933434 10.42977
##      152091    152092    152093    152094    152095    152096    152097
## 1  9.859562 10.219391  9.470575 10.309006  9.508088  9.723065  9.312566
## 2  9.923364 10.271600  9.448279 10.327485  9.664549  9.830061  9.498656
## 3 10.702172  9.938074 10.573667 10.253561 10.037120 10.520478  9.873294
## 4 10.599870 10.336371 11.703810 10.358940 11.027444 10.953380 11.326821
## 5  9.407315  9.741968  9.603771  9.589976  9.450726  9.812560 10.317613
## 6 10.422287 10.820136 11.062122 11.147337 10.375232 10.580872 10.496701
##     152098    152099    152100    152101    152102   152103   152104
## 1 10.11578  9.671229  9.522578  9.446791  9.863755 10.06088 10.47912
## 2 10.15996  9.732988  9.428209  9.579458  9.873145 10.10582 10.33500
## 3 10.41708 10.131242 10.031086 10.570174 10.674857 11.49763 10.14102
## 4 10.35829 10.576357 10.970142 10.975396 11.272439 12.26677 10.45566
## 5  9.34623 10.347334 10.326958 11.019246 10.357598 11.27407 10.40373
## 6 11.10263 10.933903 11.306491 10.046882 10.339136 11.14132 10.41452
##      152105   152106    152107    152108    152109    152110    152111
## 1  9.897442 10.76166  9.344508  9.132551 10.450439  9.739366 10.231334
## 2  9.989518 10.77806  9.280037  9.095388 10.501330  9.773241  9.986521
```

```
## 3 10.431309 10.26640 10.111581 10.336670 10.206476 10.373478 10.467038
## 4 10.469141 10.91839 10.485445 10.577457 10.442977 10.473789 10.831337
## 5  9.399458 10.51989 10.025571  9.662728  9.856320  9.269674 10.176672
## 6 11.417487 10.75083 10.488685  9.806426  9.751825 10.922584 10.962956
##      152112    152113    152114   152115
## 1  9.622760  9.788667  9.471935 10.00772
## 2  9.574614  9.959223  9.490847 10.07672
## 3 10.724741 10.800639 10.566865 10.43157
## 4 10.680999 10.766405 11.508577 10.63825
## 5 10.268092 10.116307  9.701825 10.26254
## 6 11.022092 10.636272 11.028256 11.34397
```

Here we have the top 5 rows of the expression matrix, rows are indices and should be replaced by gene names, and column names correspond to patients id.

fData()

- Use `eset@featureData@data` or fData() to access features (gene) metadata

using class() head() levels()

```
#what types are fData(eset) and fData(eset)$gene_symbol
class(fData(eset))
```

```
## [1] "data.frame"
```

```
class(fData(eset)$gene_symbol)
```

```
## [1] "factor"
```

```
#look up the five first non duplicated gene names
head(unique(fData(eset)$gene_symbol))
```

```
## [1] <NA>     HSPG2    NPRL3    TMEM151B FBN2     MRPL27
## 4243 Levels:  AADAC AAK1 AAR2 AARS2 AARSD1 AASDHPPT AASS ABCA1 ... ZZZ3
```

```
#or
head(levels(fData(eset)$gene_symbol))
```

```
## [1] ""       "AADAC"  "AAK1"   "AAR2"   "AARS2"  "AARSD1"
```

*** Use @ and $ accessors instead of fData() to obtain the same results**

```
#enter your code here
class(eset@featureData@data$gene_symbol)
```

```
## [1] "factor"
```

- eset dimensions

```
#what are the dimensions of eset
dim(eset)
```

```
## Features  Samples
##    27648       88
```

pData()

- Use `eset@phenoData@data` or pData() to access samples pheno data

```
# look inside pData(eset) for these dimentions [c(1:3), c(1:30)]
pData(eset)[c(1:3), c(1:30)]
```

```
##                   datasetName dbUniquePatientID study_ID.x patient_ID
## 152028 study_6577_GPL3883_all             37508       6577     152028
## 152029 study_6577_GPL3883_all             37509       6577     152029
## 152030 study_6577_GPL3883_all             37510       6577     152030
##        GEO_GSMID platform_ID GEO_platform_ID AE_platform_ID
## 152028    152028        3883         GPL3883           <NA>
## 152029    152029        3883         GPL3883           <NA>
## 152030    152030        3883         GPL3883           <NA>
##        coordinating_GSE_series_GSMID original_study_ID site_ID
## 152028                            NA             10109    <NA>
## 152029                            NA             10289    <NA>
## 152030                            NA             10335    <NA>
##        site_ID_preprocessed microarray_outlier biopsy_preTreat
## 152028                 <NA>                  0               1
## 152029                 <NA>                  0               1
## 152030                 <NA>                  0               1
##        biopsy_postTrt_days pCR_postTrt_days
## 152028                   0              NA
## 152029                   0              NA
## 152030                   0              NA
##        tumor_size_cm_preTrt_preSurgery
## 152028                             2.1
## 152029                             2.5
## 152030                             1.8
##        tumor_size_cm_secondAxis_preTrt_preSurgery
## 152028                                         NA
## 152029                                         NA
## 152030                                         NA
##        tumor_size_cm_preTrt_preSurgeryMin tumor_size_cm_postTrt
## 152028                                 NA                    NA
## 152029                                 NA                    NA
## 152030                                 NA                    NA
##        treatment_protocol_number clinical_AJCC_stage
## 152028                         1                 II
## 152029                         1                 II
## 152030                         1                 II
##        clinical_AJCC_stageRangeMin clinical_AJCC_stageRangeMax
## 152028                        <NA>                       <NA>
## 152029                        <NA>                       <NA>
## 152030                        <NA>                       <NA>
##        preTrt_lymph_node_status postTrt_lymph_node_status
## 152028                 positive                      <NA>
## 152029                 positive                      <NA>
## 152030                 positive                      <NA>
##        preTrt_totalLymphNodes preTrt_numPosLymphNodes
## 152028                     NA                       1
## 152029                     NA                       4
## 152030                     NA                       7
##        preTrt_numPosLymphNodesRemoved postTrt_totalLymphNodes
## 152028                             NA                     NA
## 152029                             NA                     NA
## 152030                             NA                     NA
```

Here you have all the informations regarding the processing and the phenotypic data of each samplkes/patients.

Here are some interesting features

```
paste(names(pData(eset)),sapply(pData(eset), class),sep = " - ")[c(1:10,143:153)]
```

```
##  [1] "datasetName - factor"
##  [2] "dbUniquePatientID - numeric"
##  [3] "study_ID.x - integer"
##  [4] "patient_ID - integer"
##  [5] "GEO_GSMID - integer"
##  [6] "platform_ID - integer"
##  [7] "GEO_platform_ID - character"
##  [8] "AE_platform_ID - character"
##  [9] "coordinating_GSE_series_GSMID - integer"
## [10] "original_study_ID - character"
## [11] "radiotherapy - factor"
## [12] "chemotherapy - factor"
## [13] "hormone_therapy - factor"
## [14] "no_treatment - factor"
## [15] "methotrexate - factor"
## [16] "cetuximab - factor"
## [17] "carboplatin - factor"
## [18] "other - factor"
## [19] "taxaneGeneral - factor"
## [20] "neoadjuvant_or_adjuvant - factor"
## [21] "study_specific_protocol_number - factor"
```

Usually, only the 10~15 first info slots are availables upon data collection. It's very rare to have access to clinical data, here the clinical data has been collected by the package developpers.

- Let's have a look at another unprocessed dataset.

```
gse.list=GEOquery::getGEO('GSE39591')
```

*** Use the class function on gse**

*** Create a new variable for gse$GSE39591_series_matrix.txt.gz and check dims**

*** Use pData [c(1:3), c(1:30)] and fData to overlook the phenotypic and feature data.**

```
class(gse.list)
```

```
## [1] "list"
```

```
gse=gse.list$GSE39591_series_matrix.txt.gz
class(gse)
```

```
## [1] "ExpressionSet"
## attr(,"package")
## [1] "Biobase"
```

```
dim(gse)
```

```
## Features  Samples
##    35557        7
```

```
pData(gse)[c(1:3), c(1:30)] #sample info
```

```
##              title geo_accession                status submission_date
## GSM972657   wt #92     GSM972657 Public on Feb 03 2014     Jul 23 2012
## GSM972658 ST3 #130     GSM972658 Public on Feb 03 2014     Jul 23 2012
## GSM972659 ST3 #116     GSM972659 Public on Feb 03 2014     Jul 23 2012
##           last_update_date type channel_count source_name_ch1 organism_ch1
## GSM972657      Feb 04 2014  RNA             1          wt #92 Mus musculus
## GSM972658      Feb 04 2014  RNA             1        ST3 #130 Mus musculus
## GSM972659      Feb 04 2014  RNA             1        ST3 #116 Mus musculus
##                               characteristics_ch1 characteristics_ch1.1
## GSM972657                    cell type: thymocyte  rna amount (ug): 0.3
## GSM972658 tissue: stage 3 invasive lymphoma tumor  rna amount (ug): 0.3
## GSM972659 tissue: stage 3 invasive lymphoma tumor  rna amount (ug): 0.3
##                       characteristics_ch1.2          characteristics_ch1.3
## GSM972657                      genotype: wt disease state: T cell lymphoma
## GSM972658 genotype: T cell specific PTEN KO disease state: T cell lymphoma
## GSM972659 genotype: T cell specific PTEN KO disease state: T cell lymphoma
##           characteristics_ch1.4
## GSM972657         rin number: 3
## GSM972658         rin number: 3
## GSM972659         rin number: 3
##                                  characteristics_ch1.5 molecule_ch1
## GSM972657 genetic background: mixed with FVB and C57BL6    total RNA
## GSM972658 genetic background: mixed with FVB and C57BL6    total RNA
## GSM972659 genetic background: mixed with FVB and C57BL6    total RNA
##           extract_protocol_ch1 label_ch1 label_protocol_ch1 taxid_ch1
## GSM972657               Trizol    biotin    Affy-FS450-0007     10090
## GSM972658               Trizol    biotin    Affy-FS450-0007     10090
## GSM972659               Trizol    biotin    Affy-FS450-0007     10090
##                                             hyb_protocol
## GSM972657 Affy-169 : 300.0 ng at 45 degree_C during 17 hours
## GSM972658 Affy-169 : 300.0 ng at 45 degree_C during 17 hours
## GSM972659 Affy-169 : 300.0 ng at 45 degree_C during 17 hours
##                                                                 scan_protocol
## GSM972657 Arrays were scanned using the GeneChip Scanner 3000 7G (Affymetrix).
## GSM972658 Arrays were scanned using the GeneChip Scanner 3000 7G (Affymetrix).
## GSM972659 Arrays were scanned using the GeneChip Scanner 3000 7G (Affymetrix).
##           description                      description.1
## GSM972657      wt #92 @5205580069786212140840371816037 2
## GSM972658    ST3 #130 @5205580069786212140840371816041 2
## GSM972659    ST3 #116 @5205580067295708280840351266313 4
##                                     data_processing platform_id
## GSM972657 Affyemtrix expression console data processing     GPL6246
## GSM972658 Affyemtrix expression console data processing     GPL6246
## GSM972659 Affyemtrix expression console data processing     GPL6246
##               contact_name
## GSM972657 Kevin,,Lebrigand
## GSM972658 Kevin,,Lebrigand
## GSM972659 Kevin,,Lebrigand
##                                                  contact_laboratory
## GSM972657 Functional Genomics Platform of Nice-Sophia-Antipolis, France.
## GSM972658 Functional Genomics Platform of Nice-Sophia-Antipolis, France.
## GSM972659 Functional Genomics Platform of Nice-Sophia-Antipolis, France.
##           contact_institute        contact_address
## GSM972657         IPMC/CNRS 660 route des lucioles
## GSM972658         IPMC/CNRS 660 route des lucioles
## GSM972659         IPMC/CNRS 660 route des lucioles
```

```
head(fData(gse)) #gene info
```

```
##                ID GB_LIST SPOT_ID seqname RANGE_GB RANGE_STRAND
## 10338001 10338001         control     ---                   ---
## 10338002 10338002         control     ---                   ---
## 10338003 10338003         control     ---                   ---
## 10338004 10338004         control     ---                   ---
## 10338005 10338005         control     ---                   ---
## 10338006 10338006         control     ---                   ---
##          RANGE_START RANGE_STOP total_probes gene_assignment
## 10338001         ---        ---          622             ---
## 10338002         ---        ---          454             ---
## 10338003         ---        ---          723             ---
## 10338004         ---        ---           20             ---
## 10338005         ---        ---          477             ---
## 10338006         ---        ---          468             ---
##                                                                                mrna_assignment
## 10338001                                                                                   ---
## 10338002                                                                                   ---
## 10338003                                                                                   ---
## 10338004 --- // --- // AFFX-BioB-3_at, bac_spike // --- // --- // --- // --- // --- // --- // ---
## 10338005                                                                                   ---
## 10338006                                                                                   ---
##                           category
## 10338001             control->affx
## 10338002 control->bgp->antigenomic
## 10338003             control->affx
## 10338004             control->affx
## 10338005 control->bgp->antigenomic
## 10338006 control->bgp->antigenomic
```

We find that the collected data is under the form of a list and that it contains an Expression set as shown before.

Overview of the gene expression: & heatmap & MDS plots

Back to breast cancer data. For the rest of the analysis we will focus on the 4th dataset in the eset list because there are some interesting clinical data associated to it.

```
##
## Analyzing dataset 1 or dataset named study_6577_GPL3883_all
```

```
##
## Note: this function assumes your missing values
##   are proper NAs, not "null",etc.
```

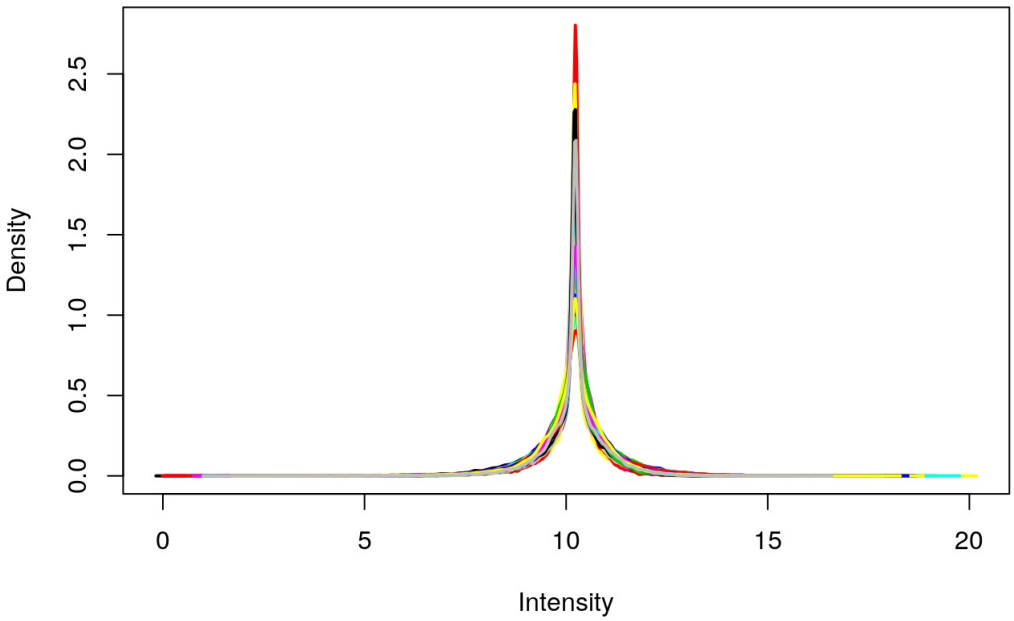
```
## It's best to impute NA values before running this function
## otherwise it may set averages to NA if there is 1 NA present.
## This function just removes any genes whose key is NA.
```

```
##
## You may get a warning here because key (usually gene) names are
##   duplicated so it can't use them as row names.
##   That's OK, because we are immediately collapsing them into one feature.
```

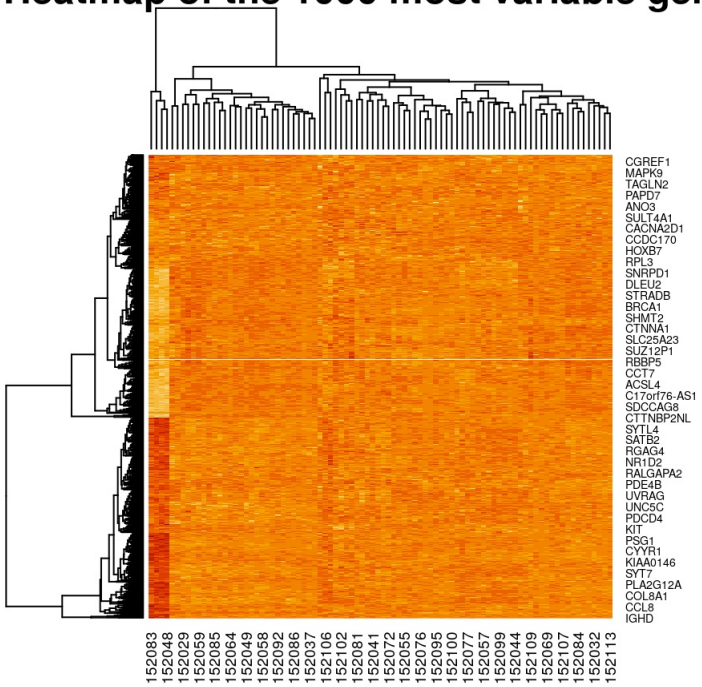
```
##
## Starting with  88patients.
```

```
## found no multiple samples from the same patient(s)
```

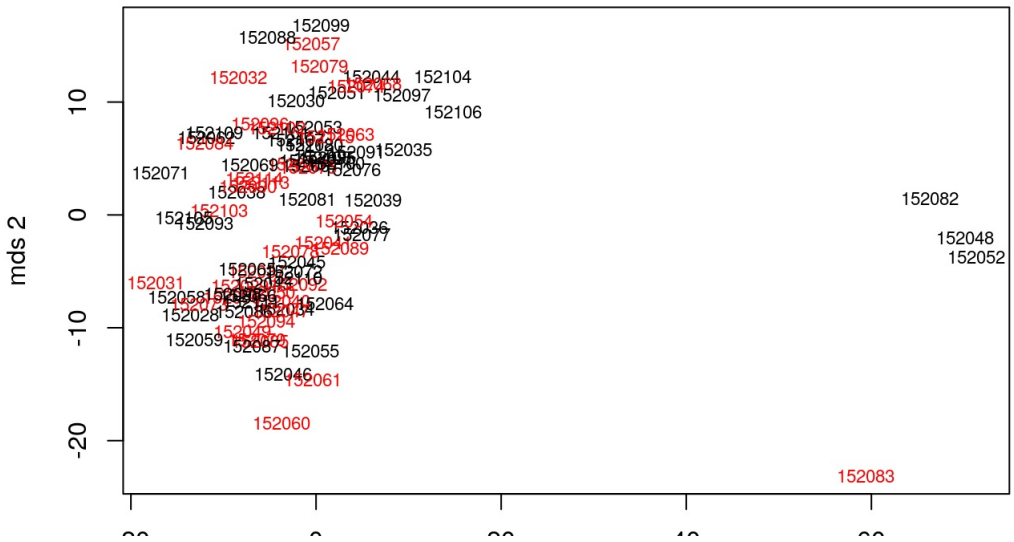
Features density across samples

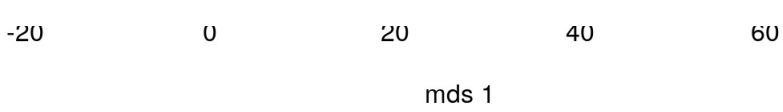


Heatmap of the 1000 most variable genes



Metric MDS





Survival of the patient is overlayed in color (black: remission, red: relapse)

These vizualisation gives us some insights on the quality of the expression set we are analysing.

- We can have a look to the distribution of the feature (genes) expressions.
- We can overlook the expression of the top 1000 most variable genes in a heatmap to see wether or not some patterns already emerge.
- We can see here that the 1000 most variable genes fail to explain the survival of the patients.

Sorting out the clinical table

```
clinical=pData(eset)

clinical=data.frame(clinical$study_ID,
           clinical$OS,
           clinical$OS_months_or_MIN_months_of_OS,
           clinical$RFS,
           clinical$RFS_months_or_MIN_months_of_RFS,
           clinical$metastasis,
           clinical$age,
           clinical$ER_preTrt,
           clinical$ER_fmolmg_preTrt,
           clinical$tumor_size_cm_preTrt_preSurgery,
           clinical$tumor_stage_preTrt,
            clinical$preTrt_lymph_node_status,
           clinical$ERBB2_CPN_amplified,
           clinical$Erbeta_preTrt,
           clinical$PTEN_mutation,
           clinical$PTEN_pos
           )

paste(names(clinical),sapply(clinical, class),sep = " - ")
```

```
##  [1] "clinical.study_ID - integer"
##  [2] "clinical.OS - integer"
##  [3] "clinical.OS_months_or_MIN_months_of_OS - numeric"
##  [4] "clinical.RFS - integer"
##  [5] "clinical.RFS_months_or_MIN_months_of_RFS - numeric"
##  [6] "clinical.metastasis - integer"
##  [7] "clinical.age - numeric"
##  [8] "clinical.ER_preTrt - integer"
##  [9] "clinical.ER_fmolmg_preTrt - integer"
## [10] "clinical.tumor_size_cm_preTrt_preSurgery - numeric"
## [11] "clinical.tumor_stage_preTrt - factor"
## [12] "clinical.preTrt_lymph_node_status - factor"
## [13] "clinical.ERBB2_CPN_amplified - integer"
## [14] "clinical.Erbeta_preTrt - integer"
## [15] "clinical.PTEN_mutation - integer"
## [16] "clinical.PTEN_pos - integer"
```

*** Use the head function to display the content of the variable named “clinical”**

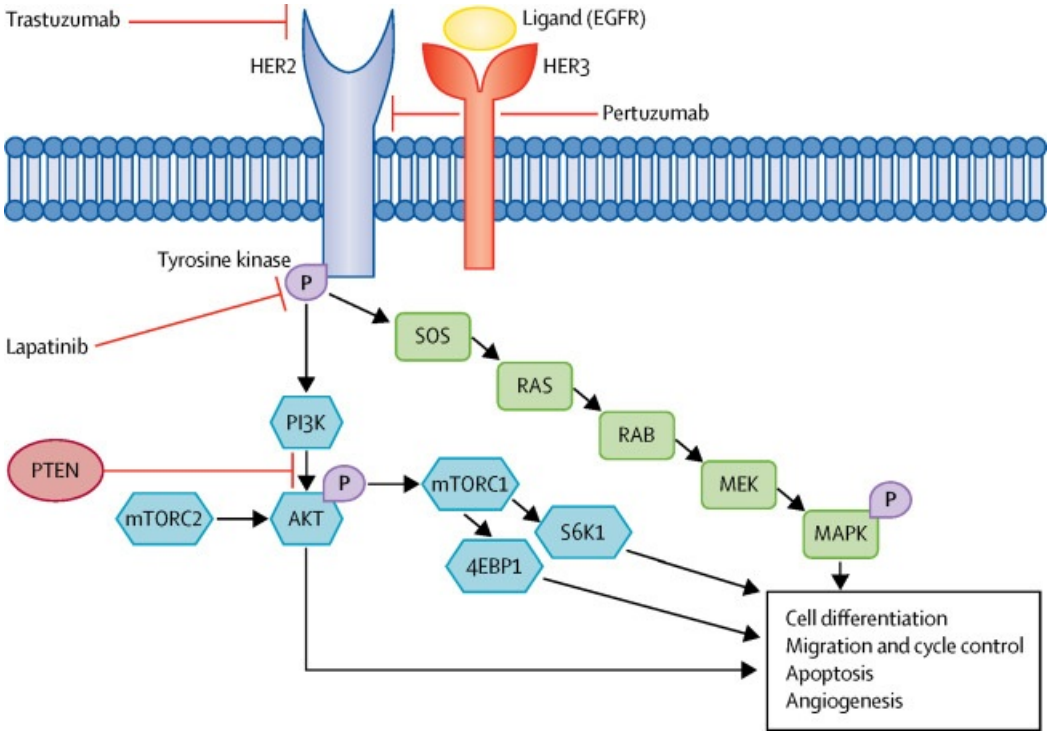
```
head(clinical)
```

```
##   clinical.study_ID clinical.OS clinical.OS_months_or_MIN_months_of_OS
## 1              6577           0                                59.60548
## 2              6577           1                                66.77260
## 3              6577           0                                61.21644
## 4              6577           1                                32.41644
## 5              6577           1                                17.81918
## 6              6577           1                                39.68219
##   clinical.RFS clinical.RFS_months_or_MIN_months_of_RFS
## 1            1                                59.60548
## 2            1                                63.05753
## 3            1                                61.21644
## 4            1                                19.52877
## 5            0                                13.08493
## 6            0                                20.74521
##   clinical.metastasis clinical.age clinical.ER_preTrt
## 1                   0           68                  1
## 2                   0           49                  1
## 3                   0           59                  0
## 4                   0           64                  1
## 5                   1           72                  0
## 6                   1           66                  0
##   clinical.ER_fmolmg_preTrt clinical.tumor_size_cm_preTrt_preSurgery
## 1                       330                                     2.1
## 2                       350                                     2.5
## 3                         1                                     1.8
## 4                       240                                     3.5
## 5                         3                                     3.0
## 6                        13                                     3.0
##   clinical.tumor_stage_preTrt clinical.preTrt_lymph_node_status
## 1                          T2                         positive
## 2                          T2                         positive
## 3                          T1                         positive
## 4                          T2                         positive
## 5                          T2                               N0
## 6                          T2                         positive
##   clinical.ERBB2_CPN_amplified clinical.Erbeta_preTrt
## 1                            0                      0
## 2                            0                      0
## 3                            0                      1
## 4                            0                      1
## 5                            1                      0
## 6                            0                      0
##   clinical.PTEN_mutation clinical.PTEN_pos
## 1                      0                 0
## 2                      0                 0
## 3                      0                 1
## 4                      0                 1
## 5                      0                 1
## 6                      0                 1
```

Gene expression impact on survival

HER2-positive breast cancer (Lancet, 2016) IF: 47.831 (2016) -> 59.102 (2019)

Anti-HER2 treatment for HER2-positive breast cancer has changed the natural biology of this disease. This Series article reviews the main achievements so far in the treatment of both metastatic and early HER2-positive breast cancer. The success of neoadjuvant therapy in HER2-positive early breast cancer is especially acknowledged, as pertuzumab has been approved on the basis of a higher proportion of patients achieving a pathological complete response with pertuzumab and trastuzumab than with trastuzumab alone in a neoadjuvant study. Event-free survival after the confirmatory adjuvant trial completed recruitment was numerically better with pertuzumab plus trastuzumab than with trastuzumab alone. With survival rates of almost 5 years in women with metastatic HER2-positive breast cancer and 75% of patients achieving a pathological complete response, new treatments in the past decade have clearly improved the prognosis of HER2-positive breast cancer. Despite these achievements, however, the persisting high toll of deaths resulting from HER2-positive breast cancer calls for continued, intensive clinical research of newer therapies and combinations.



Reminder to implement the model in R

To fit a cox model to data, we use the following code:

fit=survfit(Surv(os_time,os) ~ covariate)

survfit() is the command to compute an estimate of the survival curve of the model using the necessary data

Surv(os_time,os) containing the time to censor (here os_time) and censor (here os), returns the vector c(time,time+,etc)

os_time is a numeric vector associated to every patients containing the time from diagnosis (t0) to event (death, relapse, cencoring)

os is a boolean resulting from the question ‘does an event occured?’

covariate set to 1 for the whole cohort or corresponds to a clinical feature (e.g: treatment)

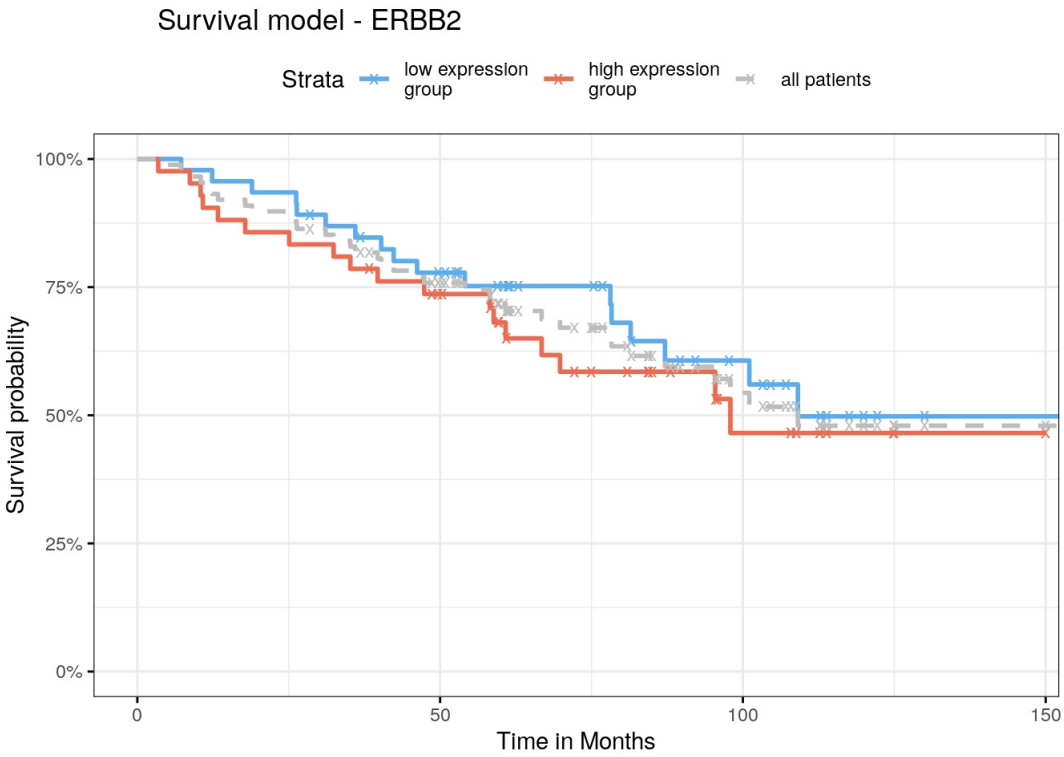
fit_table=coxph(Surv(os_time,os) ~ covariate)

fit_table() is the function to fit the model and to access statistics

- Using the mean of gene expression of all patients as a cut-off, it is possible to discretize (0: low expression, 1: high expression) the gene expression to create two groups of patients.

- ERBB2 - receptor tyrosine kinases

```
g=which(fData(eset)$gene_symbol=="ERBB2IP")
subset=apply(exprs(eset)[g,],MARGIN = 2,FUN = max)
erbb2=ifelse(subset>mean(subset),1,0)

gene=erbb2

#fit0: every patients survival probability (1 group)
fit0=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~1)

#fit: two groups based on discretized gene expression (~ erbb2)
fit=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS) ~ gene)

#fit table
fit_table=coxph(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~gene)

info=summary(fit_table)
pval=info$sctest[3]
coef=info$coefficients[1]
HR=info$coefficients[2]
ggsurvplot(list('risk'=fit,'null.model'=fit0),data=clinical,
           pval = TRUE,palette =c("steelblue2", "coral2",'Grey'),legend.labs =
               c("low expression\ngroup","high expression\ngroup","all patients"),linetype = c(1,1,2),censor.shap
e="x", censor.size = 3,
           title = "\tSurvival model - ERBB2",xlab = "Time in Months",pval.method =   TRUE,
           ggtheme = theme_bw(), ncensor.plot = F, combine = T,surv.scale=c("percent"), xlim = c(0, 145))
```

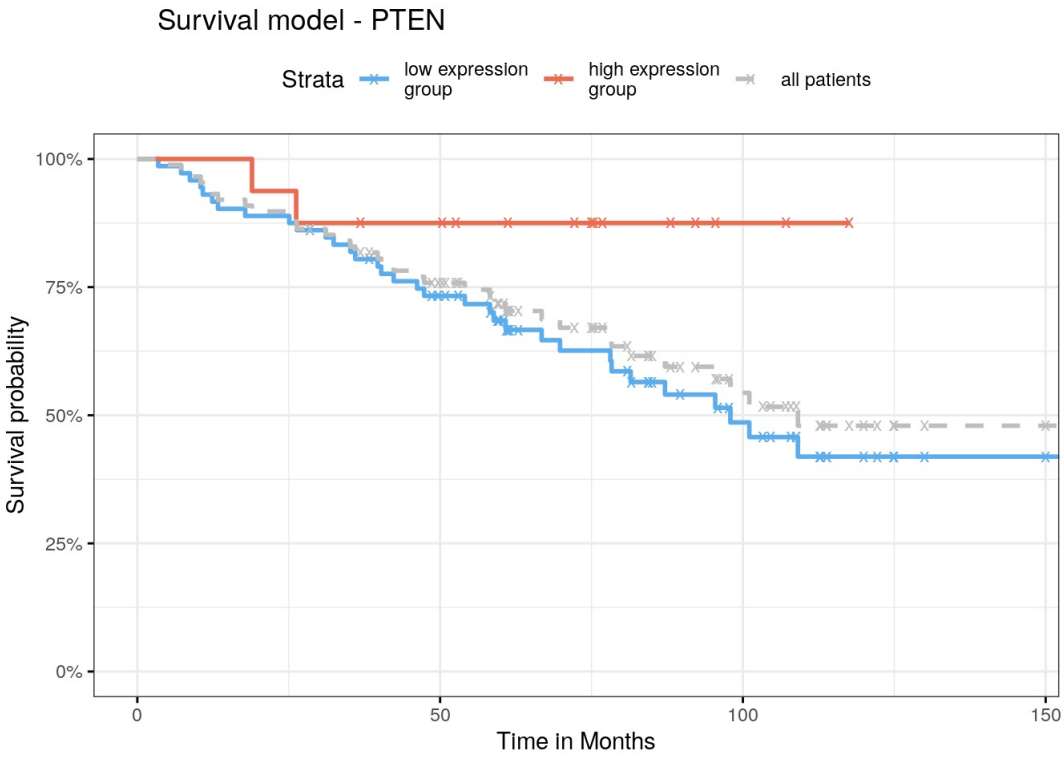



```
print(pval);
```

```
##    pvalue
## 0.4509473
```

- PTEN - tumor suppressor

```
g=which(fData(eset)$gene_symbol=="PTEN")
subset=apply(exprs(eset)[g,],MARGIN = 2,FUN = max)
PTEN=ifelse(subset>mean(subset),1,0)

gene=PTEN

#fit0: every patients survival probability (1 group)
fit0=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~1)

#fit: two groups based on discretized gene expression (~PTEN)
fit=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS) ~ gene)

#fit table
fit_table=coxph(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~gene)

info=summary(fit_table)
pval=info$sctest[3]
coef=info$coefficients[1]
HR=info$coefficients[2]
ggsurvplot(list('risk'=fit,'null.model'=fit0),data=clinical,
           pval = TRUE,palette =c("steelblue2", "coral2",'Grey'),legend.labs =
               c("low expression\ngroup","high expression\ngroup","all patients"),
           linetype = c(1,1,2),censor.shape="x", censor.size = 3,
           title = "\tSurvival model - PTEN",xlab = "Time in Months",pval.method =   TRUE,
           ggtheme = theme_bw(), ncensor.plot = F, combine = T,surv.scale=c("percent"), xlim = c(0, 145))
```




```
print(pval);
```

```
##     pvalue
## 0.04623205
```

Pvalue is significant, so we can have a look at the Hazard ratio between the two groups.

```
print(HR);
```

```
## [1] 0.2597474
```

Hazard ratio

- **3 cases**:
  - HR = 1
  - HR < 1
  - HR > 1

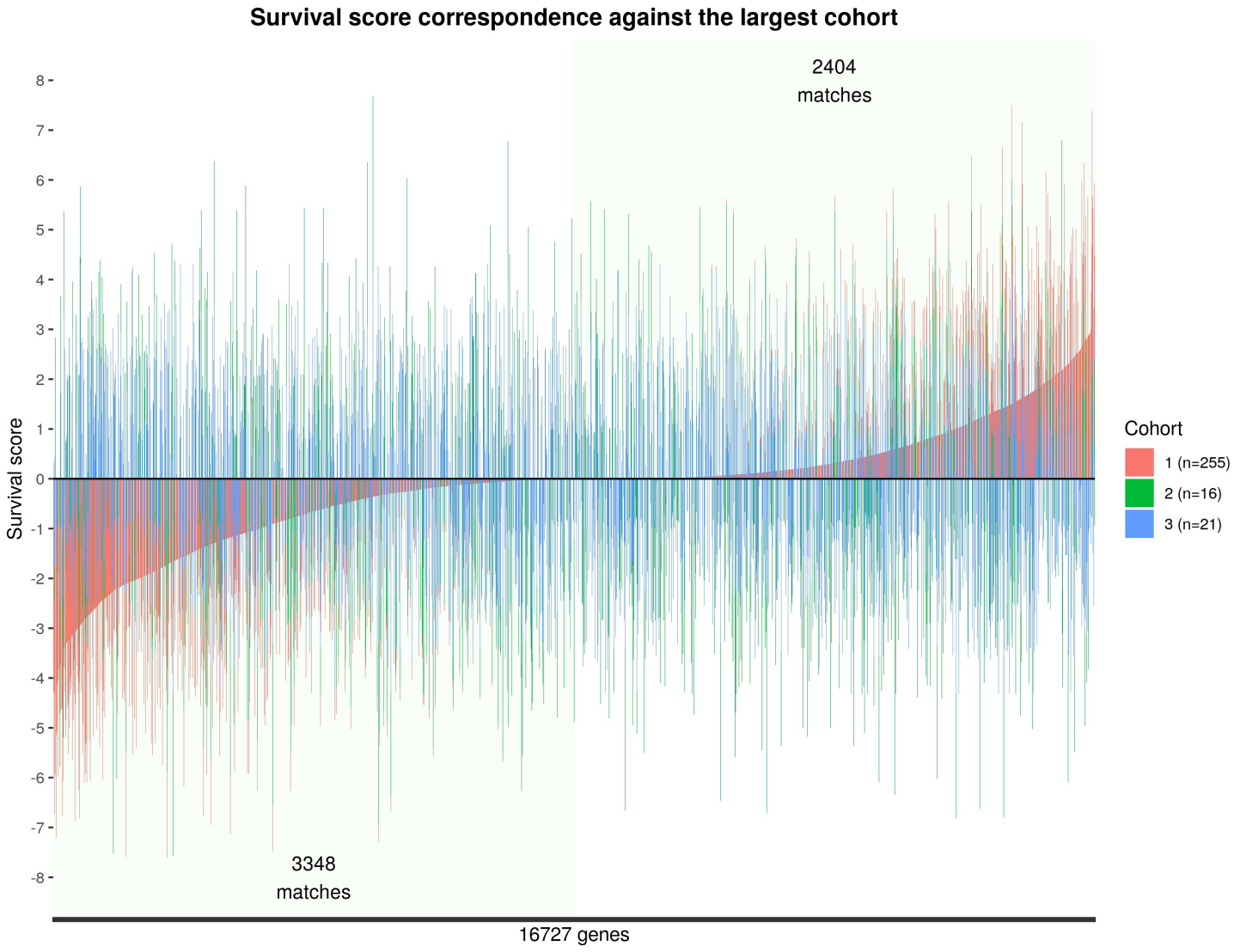
As its name suggest, it’s a ratio between too hazard (risk) values.

- If the ratio is 1 the risk values of the two groups are the same.
  (e.g: gr1=risk of 8, gr2 risk of 8 - HR = 8/8 = 1)
- If the ratio is below 1 the risk value of the second group is higher than the risk of the first group.
  (e.g: gr1=risk of 1, gr2 risk of 8 - HR = 1/8 = 0.125).
  The ratio HR<1 can be read 1/HR (hazard rate) to have a readable ratio.
  (e.g: 1/0.125 = 8 times more Hazard in one of the two groups)
- If the ratio is above 1 the risk value of the first group is higher than the risk value in the second group.
  (e.g: gr1=risk of 1, gr2 risk of 8 - HR = 8/1 = 8)

Better survival for high expression group for PTEN expression is consistent with its tumor suppressor function.

Be very carful when anylising expression data as many other confounding factors can be responsible for the heterogeneity from one cohort to another.

Here is an example of cohort heterogeneity of survival scores across all genes



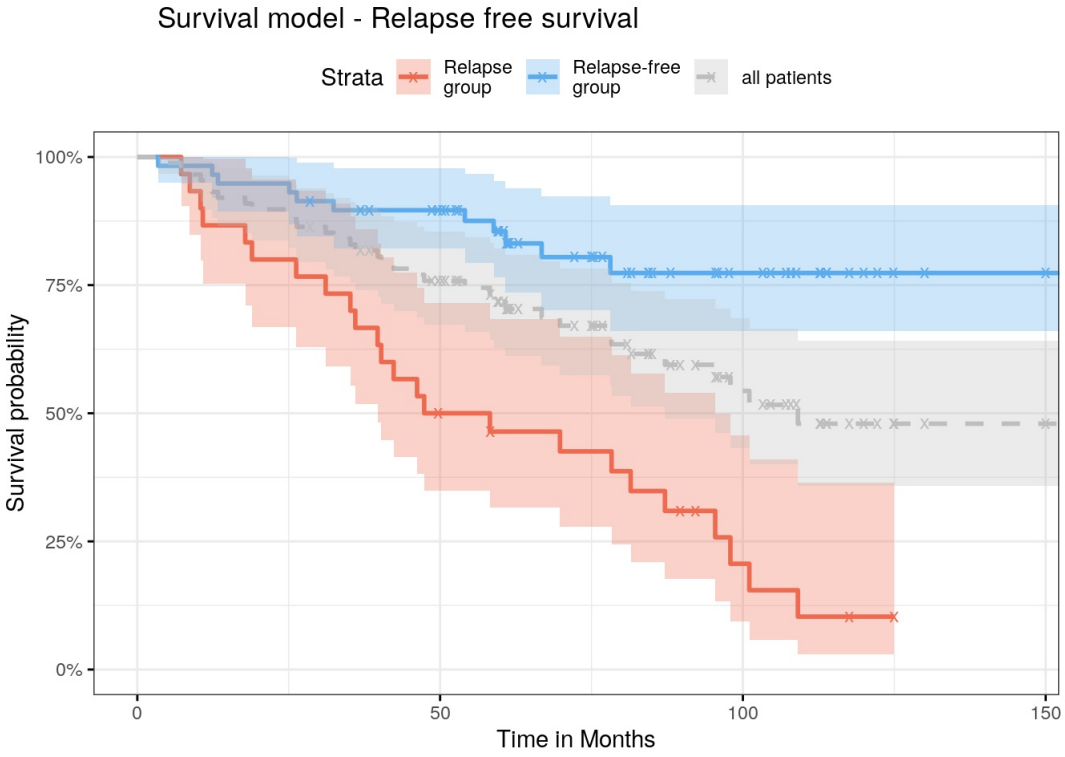
Less than 35% ((3348 + 2404)/ 16727) of genes have the same impact on survival in 3 independant cohorts.

In silico hypothesis must be confronted to a biological experiment

Relapse impact on survival

```
#fit0: every patients survival probability (1 group)
fit0=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~1)

#fit: two groups based on clinical data  (~RFS)
fit=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS) ~ clinical$clinical.RFS)

#fit table
fit_table=coxph(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~clinical$clinical.RFS)
```

```
info=summary(fit_table)
pval=info$sctest[3]
coef=info$coefficients[1]
HR=info$coefficients[2]
ggsurvplot(list('risk'=fit,'null.model'=fit0),data=clinical,
           pval = TRUE,palette =c("coral2","steelblue2",'Grey'),legend.labs =
               c("Relapse\ngroup","Relapse-free\ngroup","all patients"),linetype = c(1,1,2),censor.shape="x", cen
sor.size = 3,
           title = "\tSurvival model - Relapse free survival",xlab = "Time in Months",pval.method =   TRUE,
           ggtheme = theme_bw(), ncensor.plot = F, combine = T,surv.scale=c("percent"), conf.int = TRUE, xlim = c
(0, 145))
```




```
print(pval);print(HR)
```

```
##       pvalue
## 1.533912e-07
```

```
## [1] 0.1811849
```

How good is your model

Confusion Matrix and ROC Curve

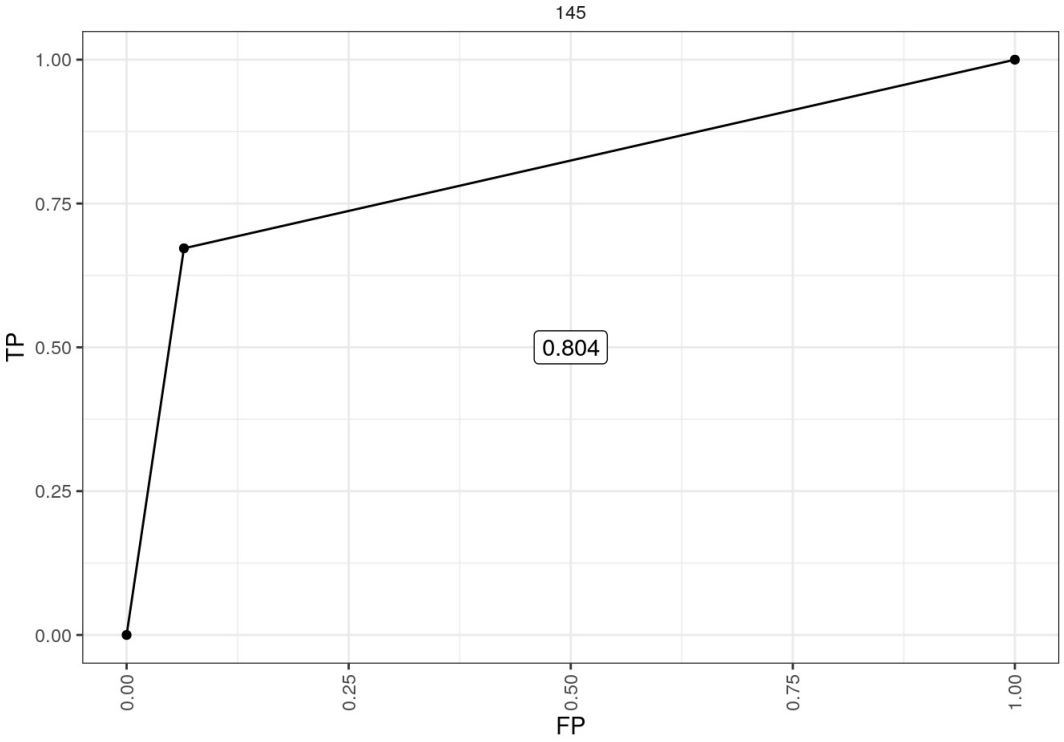
		Predicted Class	
		No	Yes
Observed Class	No	TN	FP
	Yes	FN	TP

TN	True Negative
FP	False Positive
FN	False Negative
TP	True Positive

Model Performance

Accuracy	= (TN+TP)/(TN+FP+FN+TP)
Precision	= TP/(FP+TP)
Sensitivity	= TP/(TP+FN)
Specificity	= TN/(TN+FP)

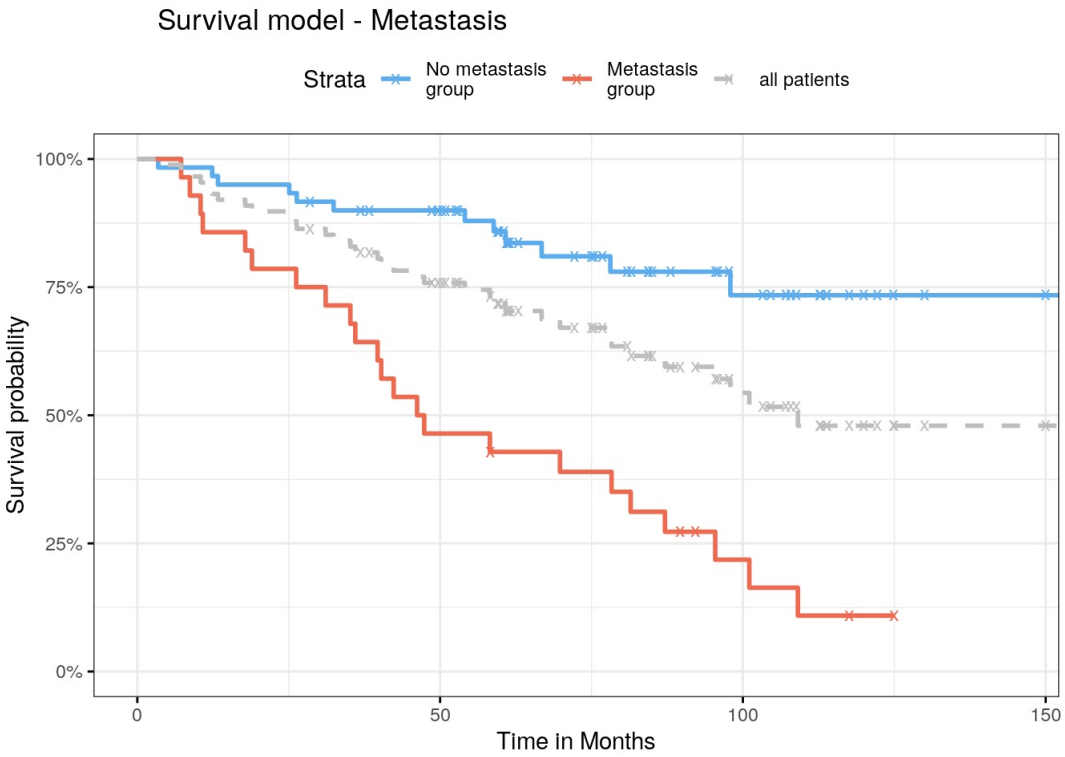
```
fit_table=readRDS('fit_table.RDS')

#predict function
lp.pred <- predict(fit_table,
                   type="lp",
                   data=data.frame(clinical))


## Define a helper function to evaluate at various t
survivalROC_helper <- function(t) {
    survivalROC(Stime        = clinical$clinical.OS_months_or_MIN_months_of_OS,
                status       = clinical$clinical.OS,
                marker       = lp.pred,
                predict.time = t,
                method       = "NNE",
                span = 0.25 * nrow(clinical)^(-0.20))
}
## Evaluate at day 145

survivalROC_data <- tibble(t = 145) %>%
    mutate(survivalROC = map(t, survivalROC_helper),
           ## Extract scalar AUC
           auc = map_dbl(survivalROC, magrittr::extract2, "AUC"),
           ## Put cut off dependent values in a data_frame
           df_survivalROC = map(survivalROC, function(obj) {
               as_tibble(obj[c("cut.values","TP","FP")])
           })) %>%
    dplyr::select(-survivalROC) %>%
    unnest(cols = c(df_survivalROC)) %>%
    arrange(t, FP, TP)

## Plot
survivalROC_data %>%
    ggplot(mapping = aes(x = FP, y = TP)) +
    geom_point() +
    geom_line() +
    geom_label(data = survivalROC_data %>% dplyr::select(t,auc) %>% unique,
               mapping = aes(label = sprintf("%.3f", auc)), x = 0.5, y = 0.5) +
    facet_wrap( ~ t) +
    theme_bw() +
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5),
          legend.key = element_blank(),
          plot.title = element_text(hjust = 0.5),
          strip.background = element_blank())
```



A receiver operating characteristic curve, or ROC curve, is a graphical plot that illustrates the diagnostic ability of a **binary** classifier system as its discrimination threshold is varied. It represent the FPR (false positive rate) against the TPR (true positive rate)

Metastasis impact on survival

```
#fit0: every patients survival probability (1 group)
fit0=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~1)

#fit: two groups based on clinical data (~metastasis)
fit=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS) ~ clinical$clinical.metast
asis)

#fit table
fit_table=coxph(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~clinical$clinical.meta
stasis)

info=summary(fit_table)
pval=info$sctest[3]
coef=info$coefficients[1]
HR=info$coefficients[2]
ggsurvplot(list('risk'=fit,'null.model'=fit0),data=clinical,
           pval = TRUE,palette =c("steelblue2", "coral2",'Grey'),legend.labs =
               c("No metastasis\ngroup","Metastasis\ngroup","all patients"),
          linetype = c(1,1,2),censor.shape="x", censor.size = 3,
           title = "\tSurvival model - Metastasis",xlab = "Time in Months",pval.method =   TRUE,
           ggtheme = theme_bw(), ncensor.plot = F, combine = T,surv.scale=c("percent"), xlim = c(0, 145) )
```




```
print(pval);print(HR)
```

```
##       pvalue
## 8.649197e-08
```

```
## [1] 5.50195
```

Age impact on survival

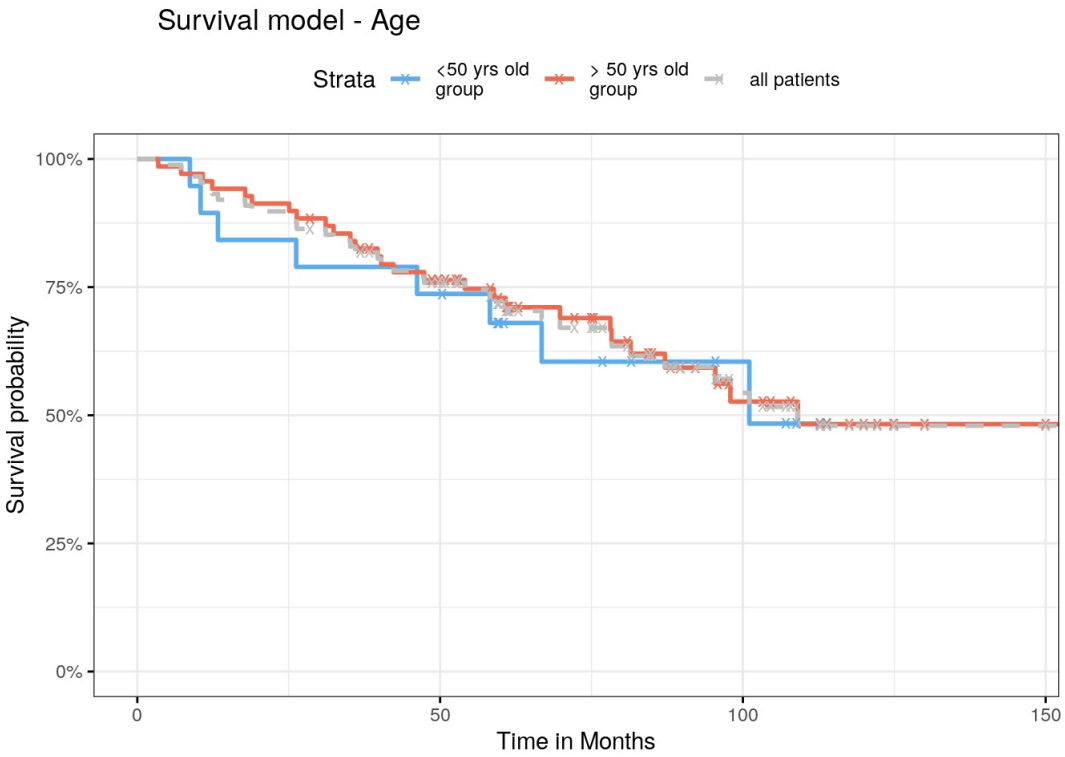
- Complete the code to evaluate the age impact on survival

- Break the cohort into two groups (< and >=50 years old)

- What can you say about the results of the regression model ?

```
age=ifelse(clinical$clinical.age>=50,1,0)

#fit0: every patients survival probability (1 group)
fit0=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~1)

#fit: two groups based on clinical data (~age)
fit=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS) ~ age)

#fit table
fit_table=coxph(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~age)
```

```
info=summary(fit_table)
pval=info$sctest[3]
coef=info$coefficients[1]
HR=info$coefficients[2]
ggsurvplot(list('risk'=fit,'null.model'=fit0),data=clinical,
         pval = TRUE,palette =c("steelblue2", "coral2",'Grey'),legend.labs =
             c("<50 yrs old\ngroup","> 50 yrs old\ngroup","all patients"),
        linetype = c(1,1,2),censor.shape="x", censor.size = 3,
         title = "\tSurvival model - Age",xlab = "Time in Months",pval.method =   TRUE,
         ggtheme = theme_bw(), ncensor.plot = F, combine = T,surv.scale=c("percent"),xlim = c(0, 145) )
```




```
pval
```

```
##    pvalue
## 0.7824474
```

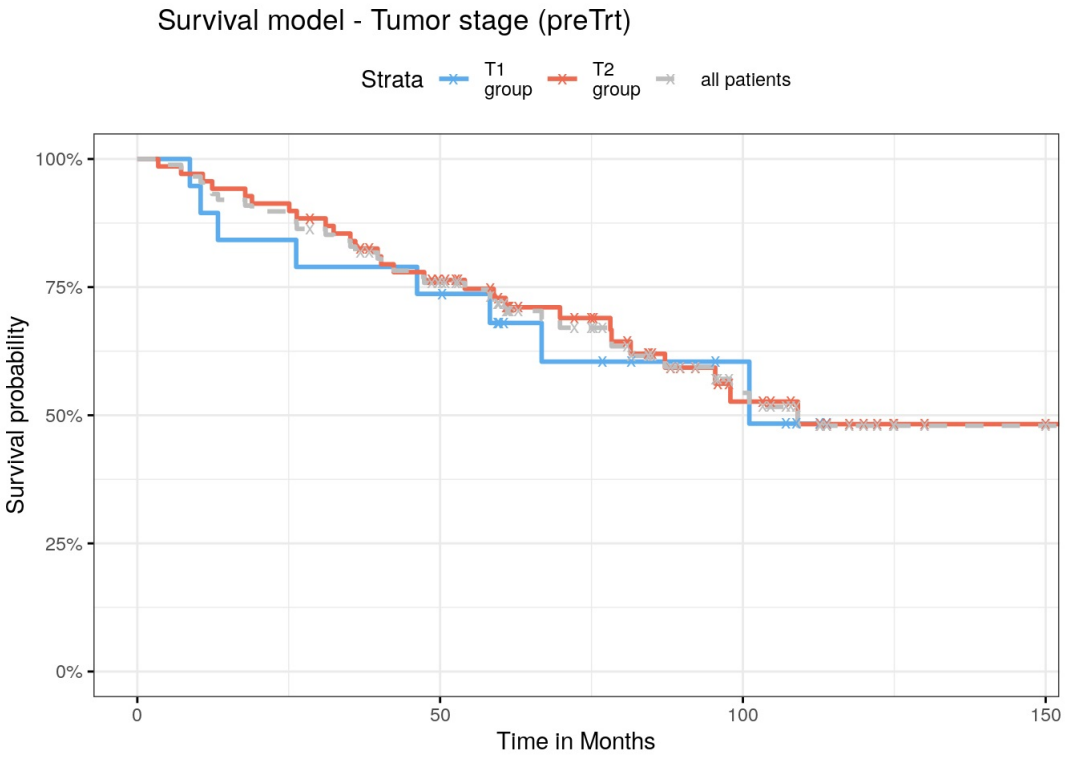
With this cut off the model is not discriminant in term of survival between the two groups.

Cut off can be adapted to the informations found in the litterature.

Tumor stage impact on survival 'clinical.tumor_stage_preTrt'

```
#fit0: every patients survival probability (1 group)
fit0=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~1)

#fit: two groups based on clinical data (~tumor stage)
fit=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS) ~ clinical$clinical.tumor_
stage_preTrt)

#fit table
fit_table=coxph(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~clinical$clinical.tumo
r_stage_preTrt)

info=summary(fit_table)
pval=info$sctest[3]
coef=info$coefficients[1]
HR=info$coefficients[2]
```

```
pval=info$sctest[3]
coef=info$coefficients[1]
HR=info$coefficients[2]*10
if (pval<0.05){print(pval);print(HR)}
ggsurvplot(list('risk'=fit,'null.model'=fit0),data=clinical,
           pval = TRUE,palette =c("steelblue2", "coral2",'Grey'),legend.labs =
               c("T1\ngroup","T2\ngroup","all patients"),
          linetype = c(1,1,2),censor.shape="x", censor.size = 3,
           title = "\tSurvival model - Tumor stage (preTrt)",xlab = "Time in Months",pval.method =TRUE,
           ggtheme = theme_bw(), ncensor.plot = F, combine = T,surv.scale=c("percent"), xlim = c(0, 145) )
```




```
pval
```

```
##    pvalue
## 0.7824474
```

Mutations impact on survival

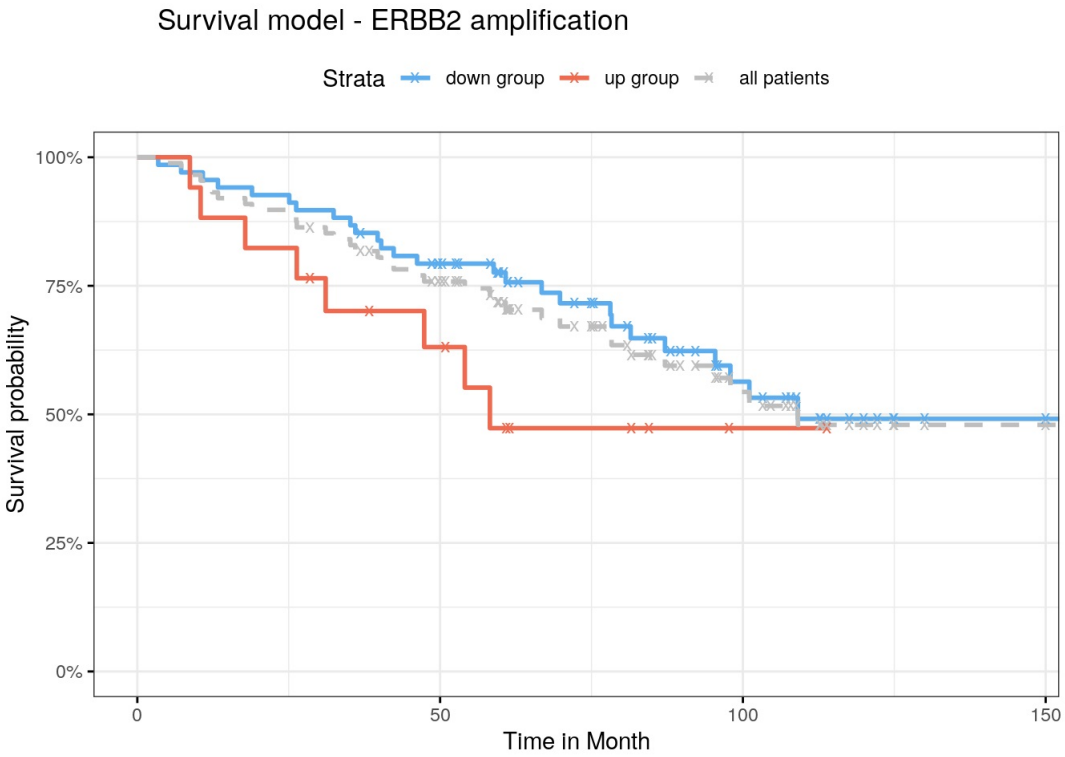
- ERBB2 amplification

```
#fit0: every patients survival probability (1 group)
fit0=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~1)

#fit: two groups based on clinical data (~ERBB2 amplification)
fit=survfit(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS) ~ clinical$clinical.ERBB2_
CPN_amplified)

#fit table
fit_table=coxph(Surv(clinical$clinical.OS_months_or_MIN_months_of_OS,clinical$clinical.OS)~clinical$clinical.ERBB
2_CPN_amplified)

info=summary(fit_table)
pval=info$sctest[3]
coef=info$coefficients[1]
HR=info$coefficients[2]
```

```
pval=info$sctest[3]
coef=info$coefficients[1]
HR=info$coefficients[2]
print(pval)
```

```
##    pvalue
## 0.1250815
```

```
if (pval<0.05){print(pval);print(HR)}
ggsurvplot(list('risk'=fit,'null.model'=fit0),data=clinical,
           pval = TRUE,palette =c("steelblue2", "coral2",'Grey'),legend.labs =
               c("down group","up group","all patients"),linetype = c(1,1,2),censor.shape="x", censor.size = 3,
           title = "\tSurvival model - ERBB2 amplification",xlab = "Time in Month",pval.method =   TRUE,
           ggtheme = theme_bw(), ncensor.plot = F, combine = T,surv.scale=c("percent"), xlim = c(0, 145) )
```




```
pval
```

```
##    pvalue
## 0.1250815
```

Survival across studies

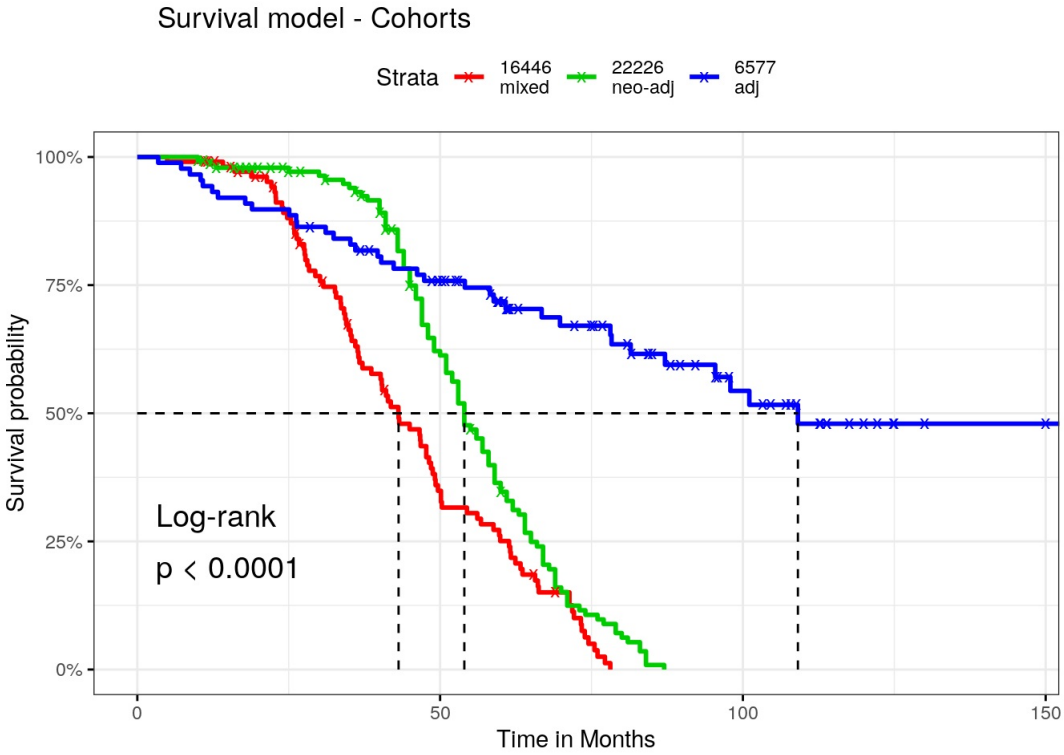
```
studies=c("6577","16446","22226")
clinical=clinicalData$clinicalTable[which(clinicalData$clinicalTable$study_ID%in%studies),]

fit <- survfit(Surv(clinical$OS_months_or_MIN_months_of_OS,clinical$OS) ~ clinical$study_ID,
          data = data.frame(clinical))
fit_table <- coxph(Surv(clinical$OS_months_or_MIN_months_of_OS,clinical$OS) ~ clinical$study_ID )
info=summary(fit_table)
print(info)
```

```
## Call:
## coxph(formula = Surv(clinical$OS_months_or_MIN_months_of_OS,
##     clinical$OS) ~ clinical$study_ID)
##
##   n= 340, number of events= 245
##    (11 observations deleted due to missingness)
##
##                         coef exp(coef)  se(coef)     z Pr(>|z|)
## clinical$study_ID 8.290e-05 1.000e+00 1.117e-05 7.424 1.13e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                   exp(coef) exp(-coef) lower .95 upper .95
## clinical$study_ID         1     0.9999         1         1
##
## Concordance= 0.541  (se = 0.022 )
## Likelihood ratio test= 62.97  on 1 df,   p=2e-15
## Wald test            = 55.12  on 1 df,   p=1e-13
## Score (logrank) test = 59.27  on 1 df,   p=1e-14
```

```
pval=info$sctest[3]
coef=info$coefficients[1]
HR=info$coefficients[2]*10
if (pval<0.05){print(pval);print(HR)}
```

```
##       pvalue
## 1.375211e-14
## [1] 10.00083
```

```
ggsurvplot(fit,data=clinical,
           pval = TRUE,censor.shape="x", censor.size = 3,palette =c(2,3,4),legend.labs =
               c("16446\nmixed","22226\nneo-adj","6577\nadj"),
           title = "\tSurvival model - Cohorts",xlab = "Time in Months",pval.method =   TRUE,
           ggtheme = theme_bw(), ncensor.plot = F, combine = T,surv.scale=c("percent"), xlim = c(0, 145),surv.med
ian.line = "hv" )
```



Survival according to treatments

```
treatments=clinical[,c(117:119,124,128,136:137)]
study=ifelse(as.integer(clinical$study_ID)==22226,1,0)
treatments=cbind(treatments,study)
neoadj_or_adj=clinical[,c(138)]
treatment_class=rowSums(sapply(treatments,as.numeric)-1)+1
```

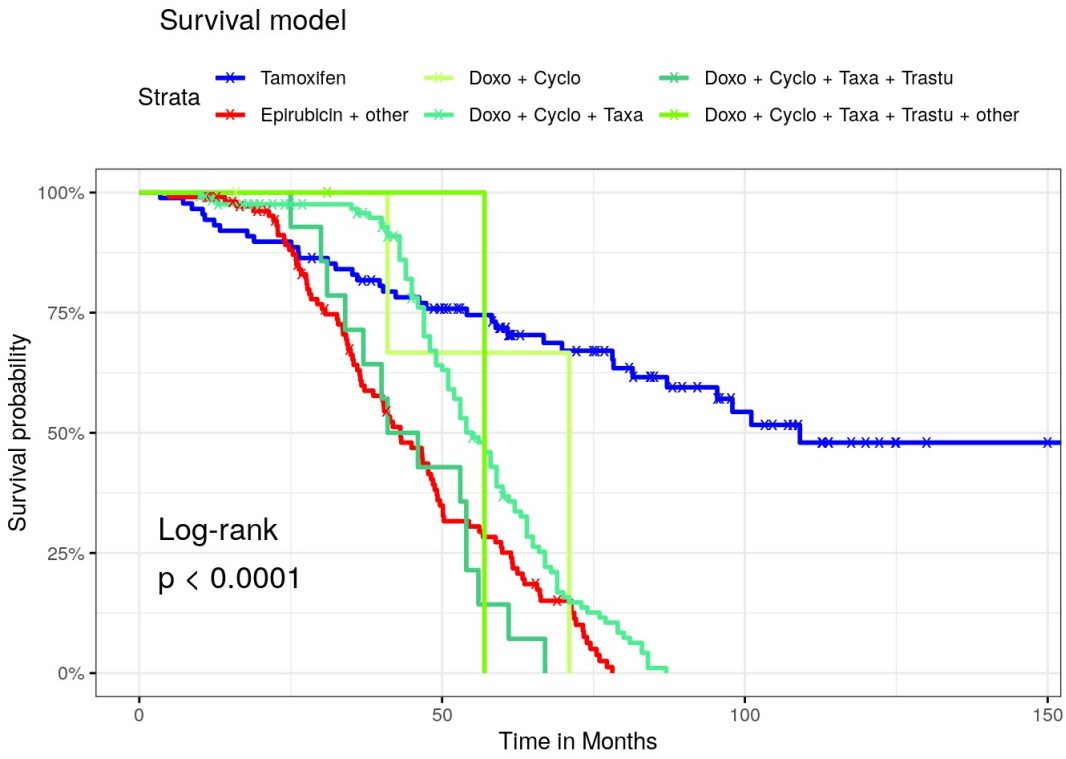
```
fit <- survfit(Surv(clinical$OS_months_or_MIN_months_of_OS,clinical$OS) ~ treatment_class ,
            data = data.frame(clinical))
fit_table <- coxph(Surv(clinical$OS_months_or_MIN_months_of_OS,clinical$OS) ~ treatment_class)
info=summary(fit_table)
print(info)
```

```
## Call:
## coxph(formula = Surv(clinical$OS_months_or_MIN_months_of_OS,
##     clinical$OS) ~ treatment_class)
##
##   n= 340, number of events= 245
##    (11 observations deleted due to missingness)
##
##                     coef exp(coef) se(coef)     z Pr(>|z|)
## treatment_class 0.27530   1.31693  0.04754 5.791 6.99e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                 exp(coef) exp(-coef) lower .95 upper .95
## treatment_class     1.317     0.7593       1.2     1.446
##
## Concordance= 0.549  (se = 0.023 )
## Likelihood ratio test= 33.91  on 1 df,   p=6e-09
## Wald test            = 33.54  on 1 df,   p=7e-09
## Score (logrank) test = 34.72  on 1 df,   p=4e-09
```

```
pval=info$sctest[3]
coef=info$coefficients[1]
HR=info$coefficients[2]*10
if (pval<0.05){print(pval);print(HR)}
```

```
##       pvalue
## 3.806042e-09
## [1] 13.16928
```

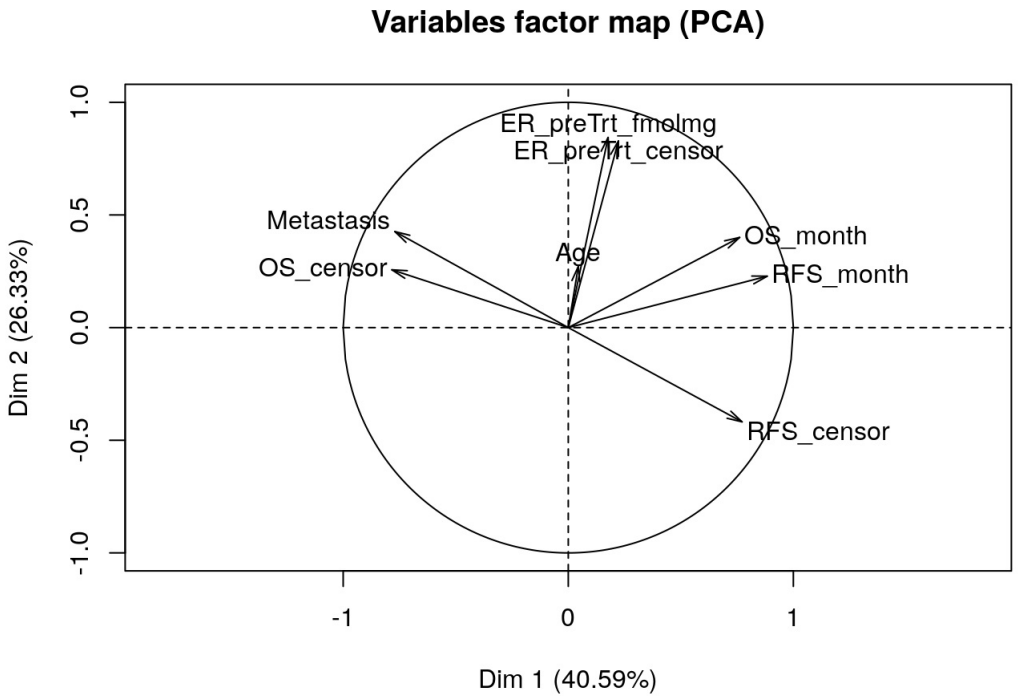
```
 ggsurvplot(fit,data=clinical,
             pval = TRUE,censor.shape="x", censor.size = 3,legend.labs =
                 c("Tamoxifen","Epirubicin + other", "Doxo + Cyclo","Doxo + Cyclo + Taxa","Doxo + Cyclo + Taxa +
Trastu"
,"Doxo + Cyclo + Taxa + Trastu + other"), palette =c(4,2,"darkolivegreen1","seagreen2","seagreen3","lawngreen"),
             title = "\tSurvival model",xlab = "Time in Months",pval.method =   TRUE,
             ggtheme = theme_bw(), ncensor.plot = F, combine = T,surv.scale=c("percent"), xlim = c(0, 145) )
```



Conclusion

- Clinical data are essential to perform a survival analysis and are often missing due to confidentiality.
- Gene expression often fail to robustly predict outcome when used alone without prior-knowledge or multiomics analysis.
- Cox reggression model is widely used and provide the user with a fast and comprehensive view of both:
    - Samples survival

- Effect of covariates on survival



- Avoid hasty conclusion as pre-tretament covariates could be correlated with treatment.

(e.g. Pretreatment ER levels (fmol/mg) can influence the treatement protocol selection and thus the impact on survival of this ER levels is minimised by a more specific treatment)