Welcome to the practical session **"big data & predictive models"**

Loic Verlingue

ISFBM module 12

January 23th 2019

Running instructions

- Open R studio
- Open New File -> R Notebook
- Go to the web https://github.com/gustaveroussy/IFSBM-bigdata/edit/master/TP_IFSBM_module12/
- Copy paste the content of TP_notebookR.Rmd in your new R Notebook
- Follow the notebook instructions
- *Enjoye!*

For very advanced R programmers

- You can erase the R code in the Notebook from the section:

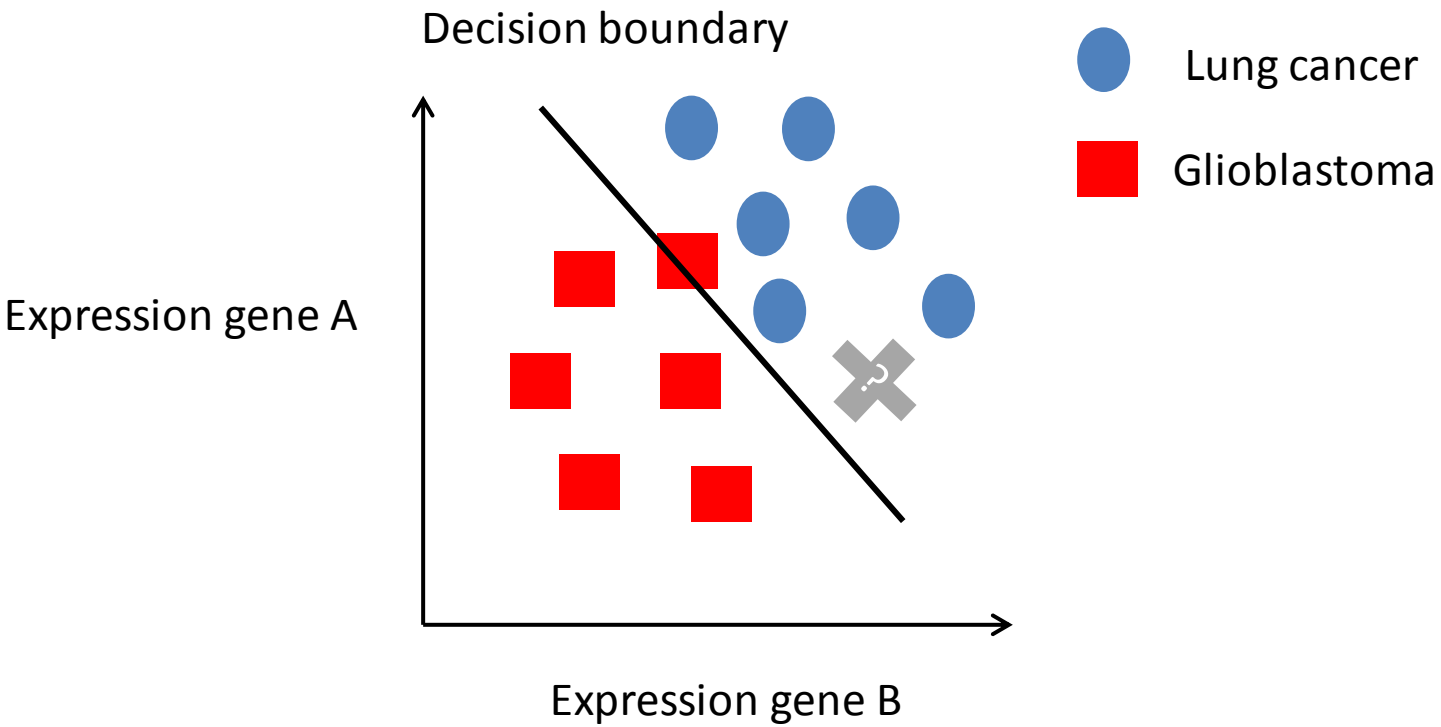
« Description of the data »

To

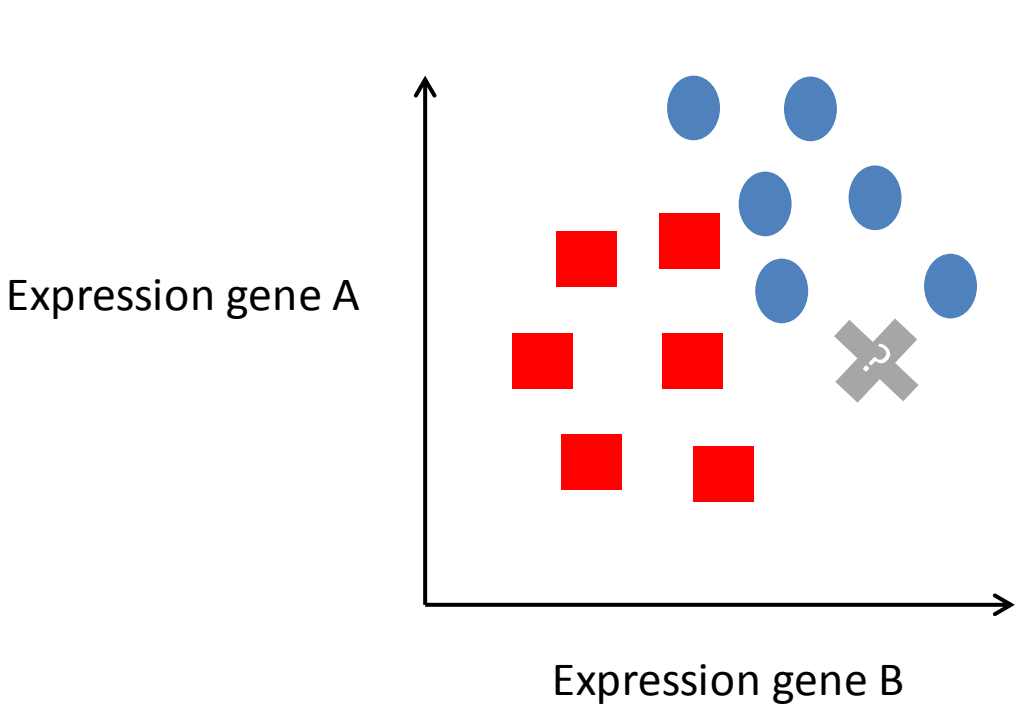
« A glimpse into survival analysis »

- And try to do the code by your own with help of the original Notebook

Classification



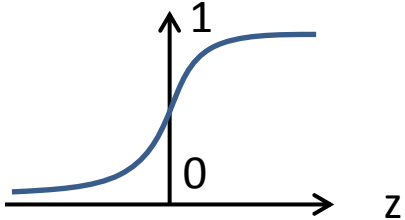
Logistic regression in 2D



Lung cancer=0

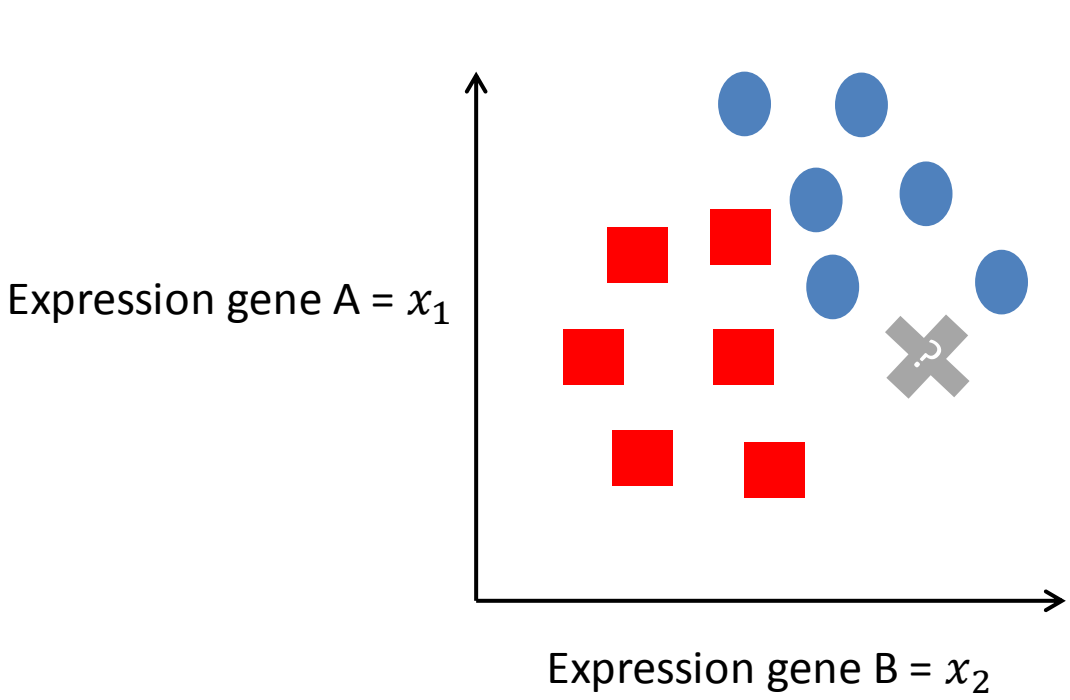
Glioblastoma=1

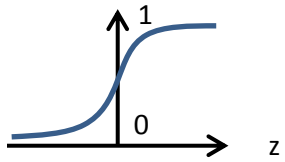
Sigmoid function = logistic function



$$g(z) = \frac{1}{1 + e^{-z}}$$

Logistic regression in 2D



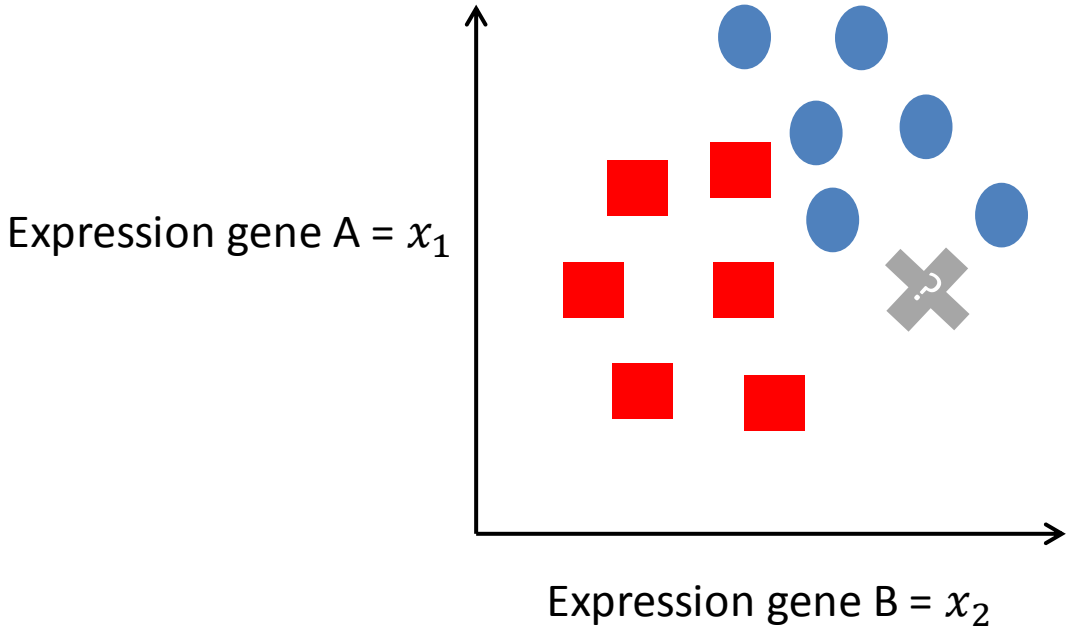


$$z = \theta_1 x_1 + \theta_2 x_2 + b$$

$$h(x) = g(z) = g(\theta_1 x_1 + \theta_2 x_2 + b)$$

→ How do you find θ_1, θ_2 & b?

Logistic regression in 2D



Define a cost function

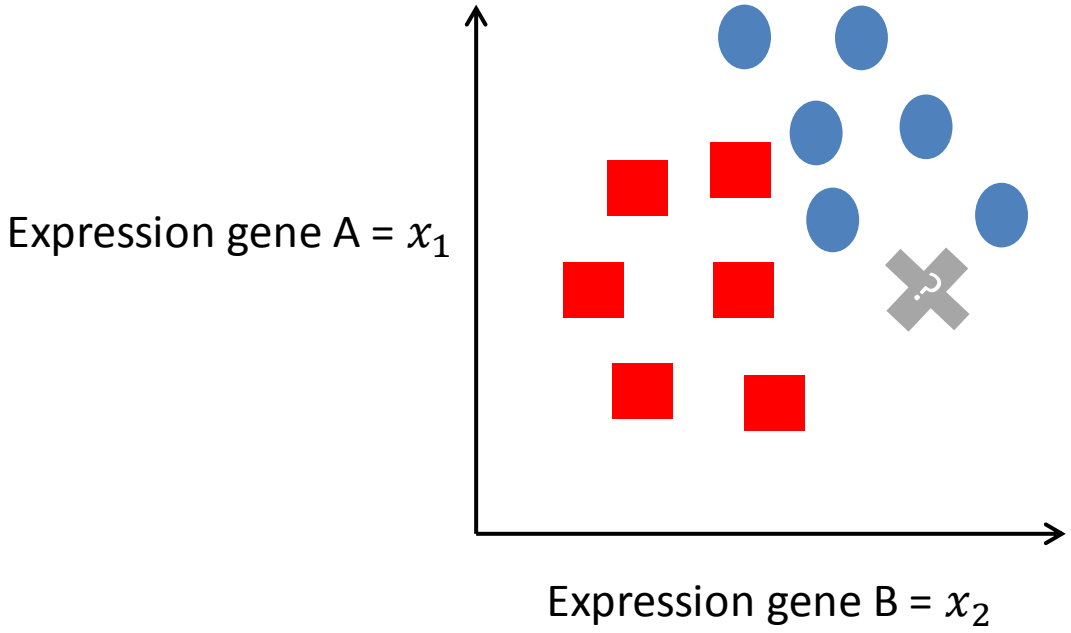
y^i is your ground truth

$h(x^i)$ is your prediction

$\rightarrow cost(\ h(x^i), y^i\ )$

- If y^i = 0 (lung adenocarcinoma) you want $h(x^i)$ to be ~0
- If y^i = 1 (glioblastoma) you want $h(x^i)$ to be ~1

Logistic regression in 2D



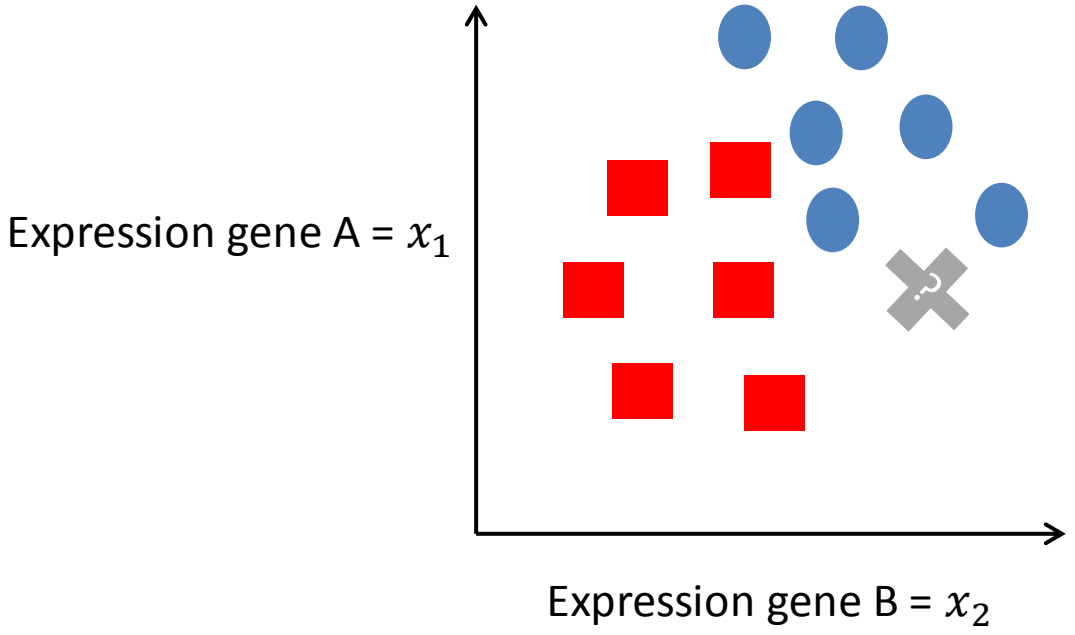
For example

$$cost(\ h(x^i), y^i\ )$$

$$-\log\left(1 - h(x^i)\right) \quad if \quad y^i = 0$$

$$-\log\left(h(x^i)\right) \quad if \quad y^i = 1$$

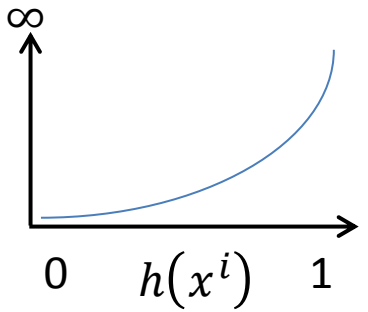
Logistic regression in 2D



For example

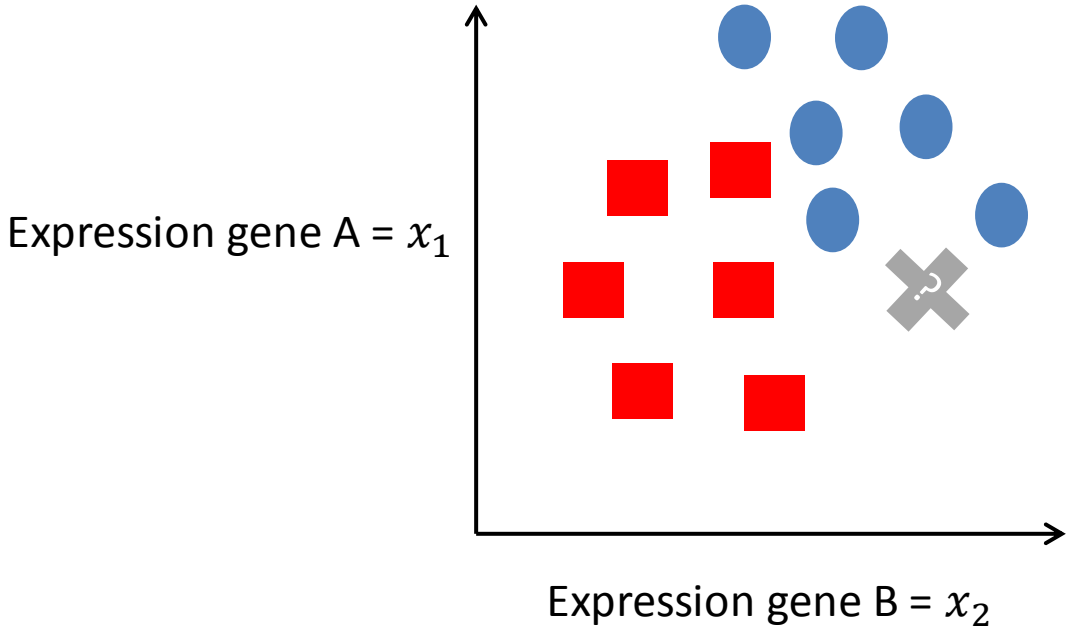
$$cost(\ h(x^i), y^i\ )$$

$$if\quad y^i = 0$$



$$cost = -\log\left(1 - h(x^i)\right)$$

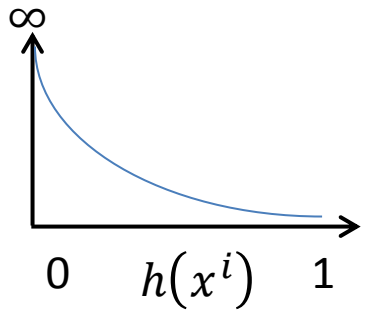
Logistic regression in 2D



For example

$$cost(\, h(x^i), y^i \,)$$

$$if \quad y^i = 1$$



$$cost = -\log\left(h(x^i)\right)$$

Logistic regression in 2D

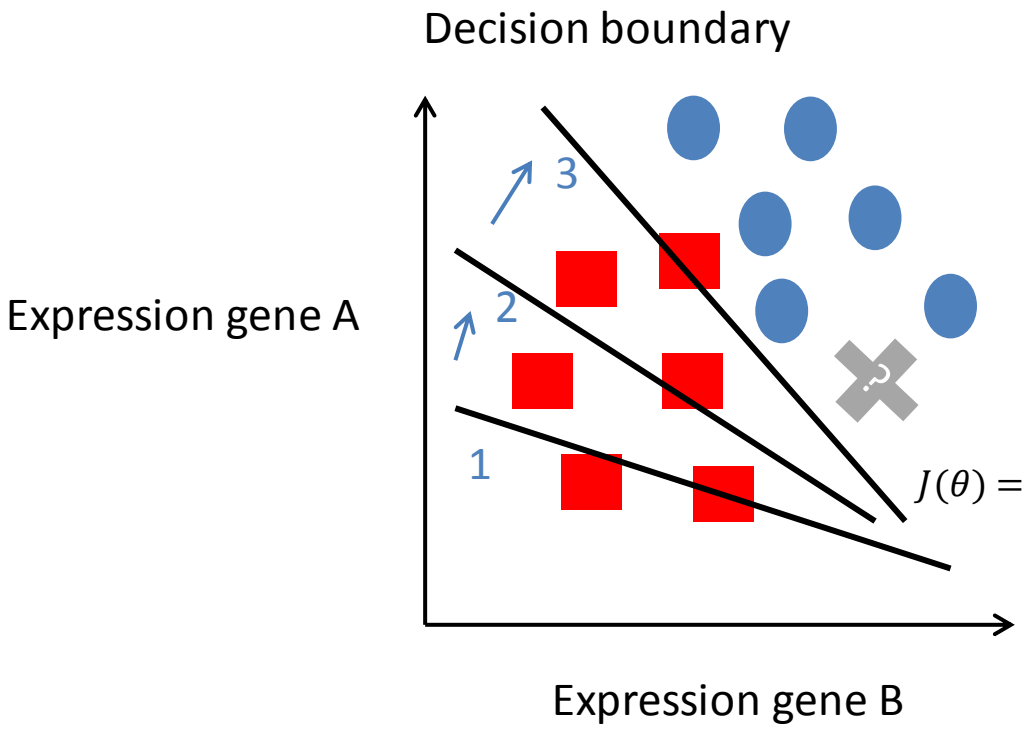
Do it on all your data

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} cost\left(h\left(x^i\right), y^i \right)$$

$$= -\frac{1}{m}\sum_{i=1}^{m} y^{\,i} * \log\left(h\left(x^i\right)\right) + \left(1 - y^i\right) * \log\left(1 - h\left(x^i\right)\right)$$

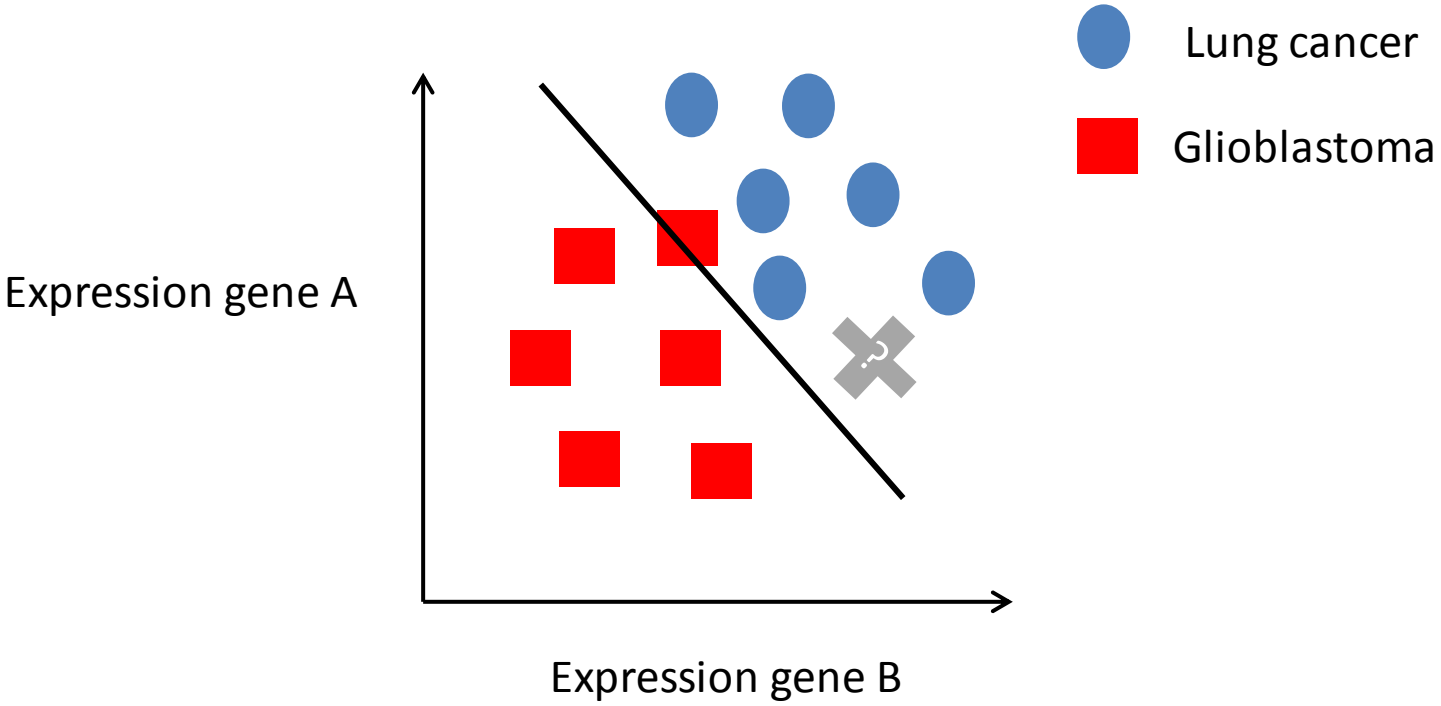
And then minimize it!

Minimization with gradient descent

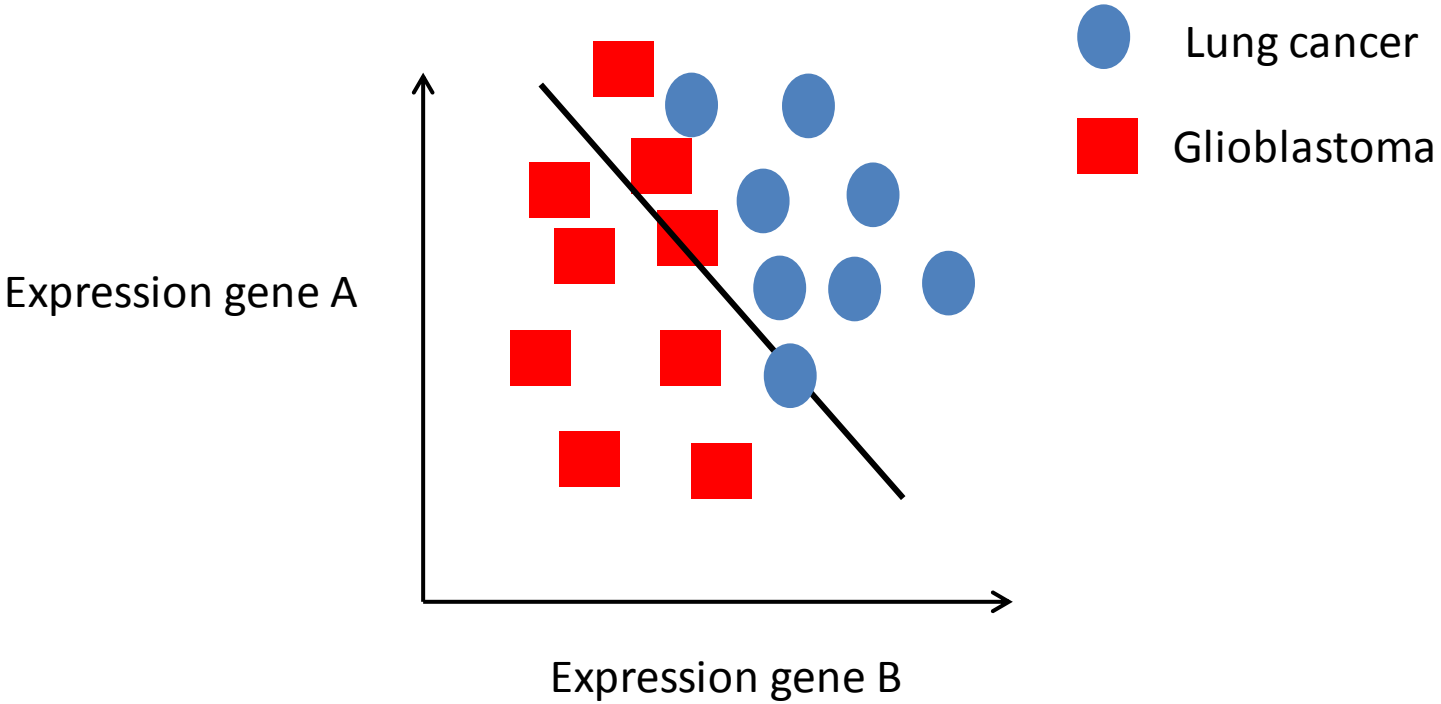


$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^{i} * \log\left(h\left(x^{i}\right)\right) + \left(1 - y^{i}\right) * \log(1 - h\left(x^{i}\right)$$

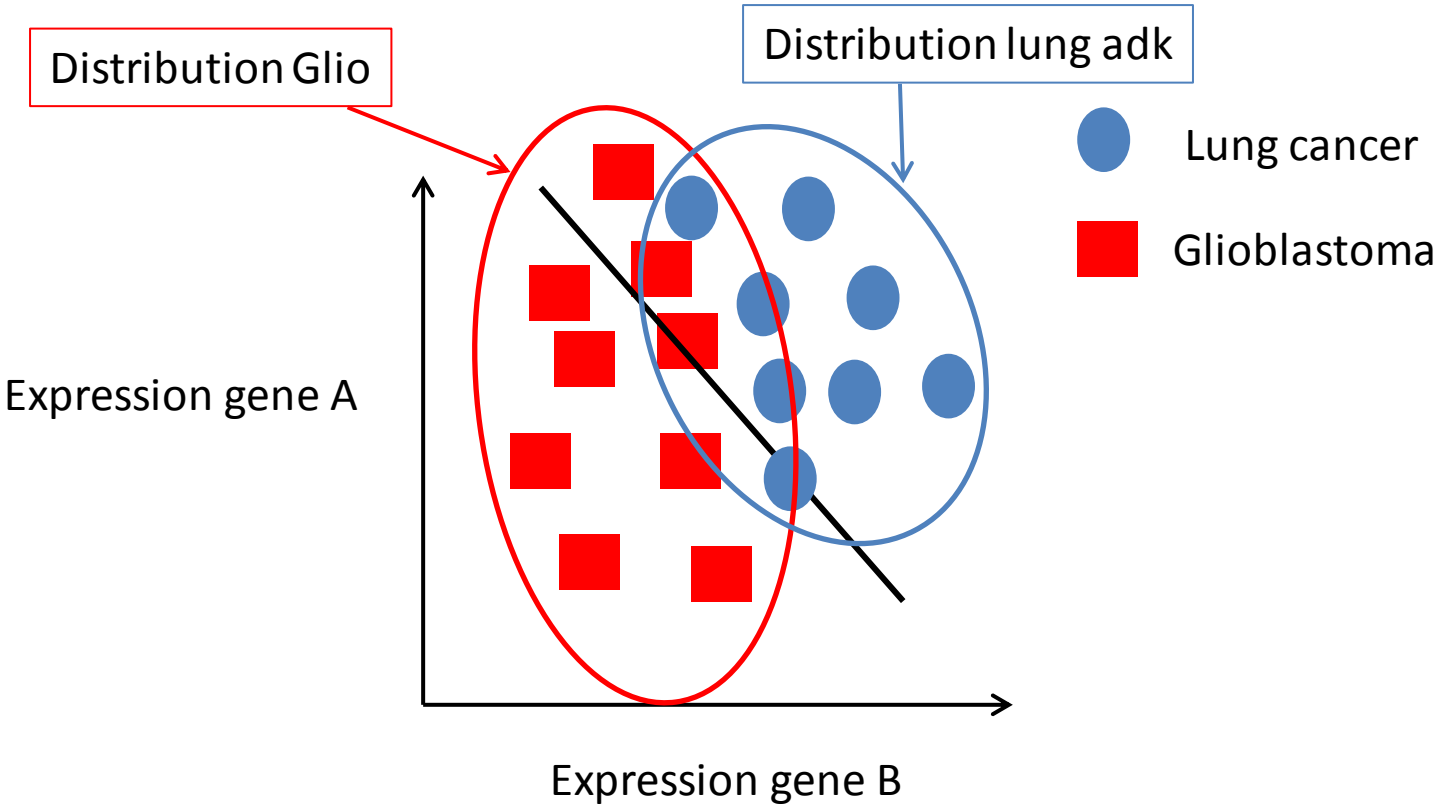
Are you write? Probably yes !



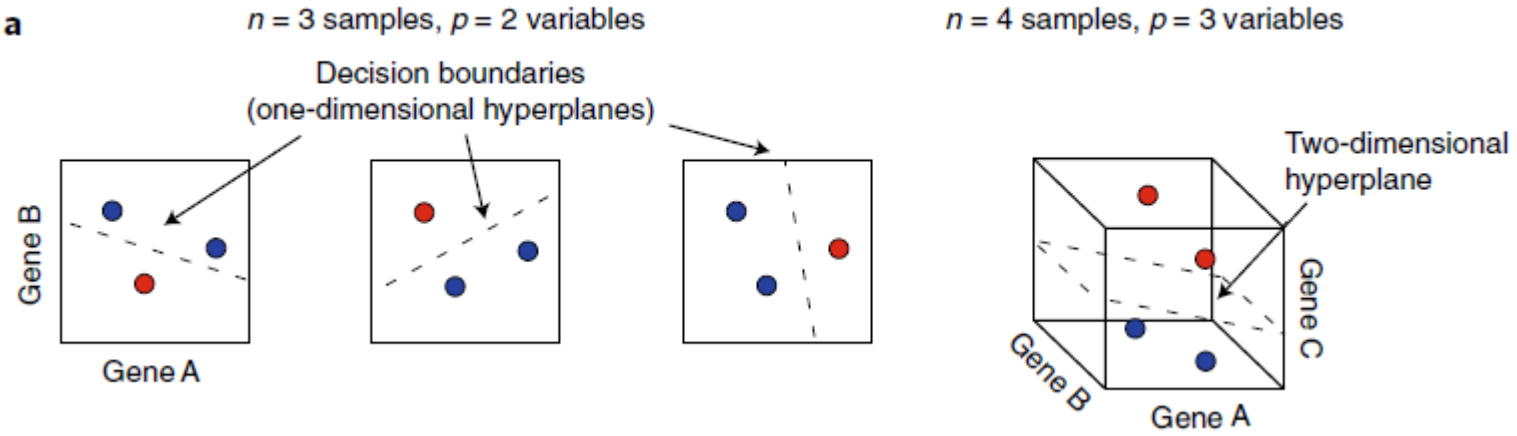
Are you write? No !



Are you write? No !



Generalisation and overfitting



Teschendorff AE. Avoiding common pitfalls in machine learning omic data science. Nat Mater. 2018 Nov 26. doi: 10.1038/s41563-018-0241-z. PubMed PMID: 30478452.

The curse of dimensionality!

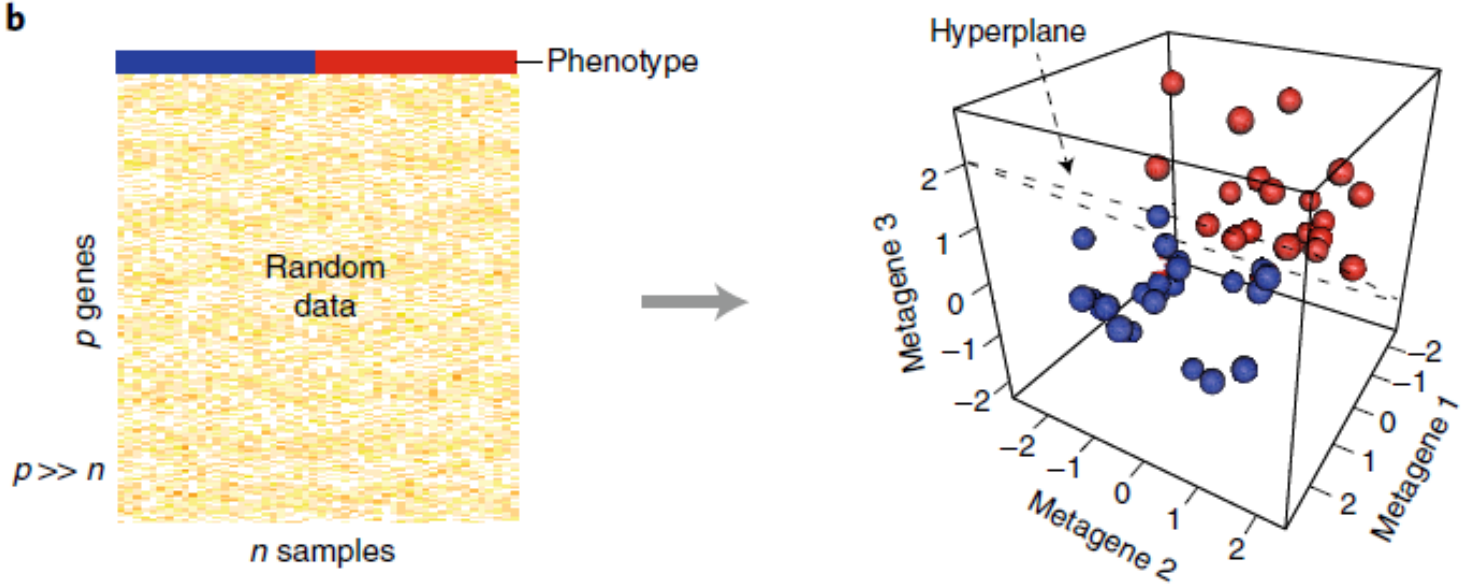
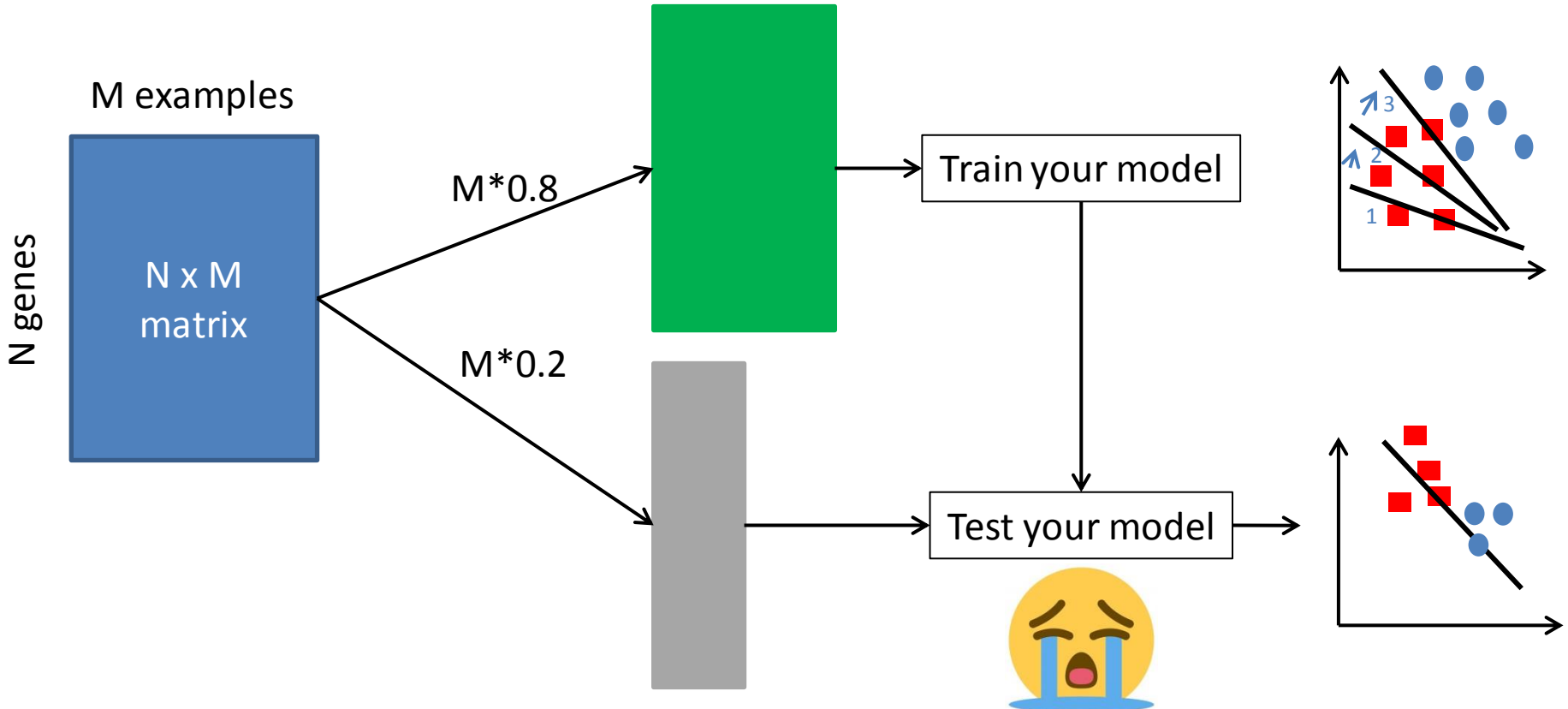


Fig. 1 | The curse of dimensionality and overfitting. a, Low-dimensional examples designed to

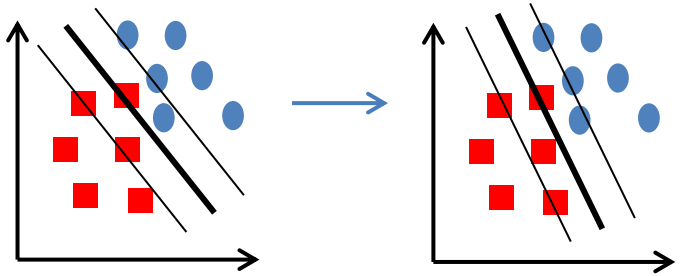
Teschendorff AE. Avoiding common pitfalls in machine learning omic data science. Nat Mater. 2018 Nov 26. doi: 10.1038/s41563-018-0241-z. PubMed PMID: 30478452.

How do you evaluate generalization?



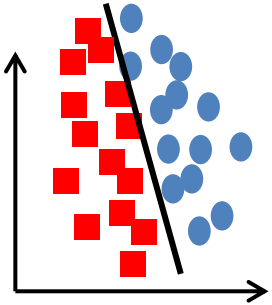
Are there strategies not to  ?

On the modeling / learning:



Eg: SVM, regularisation / penalisation..

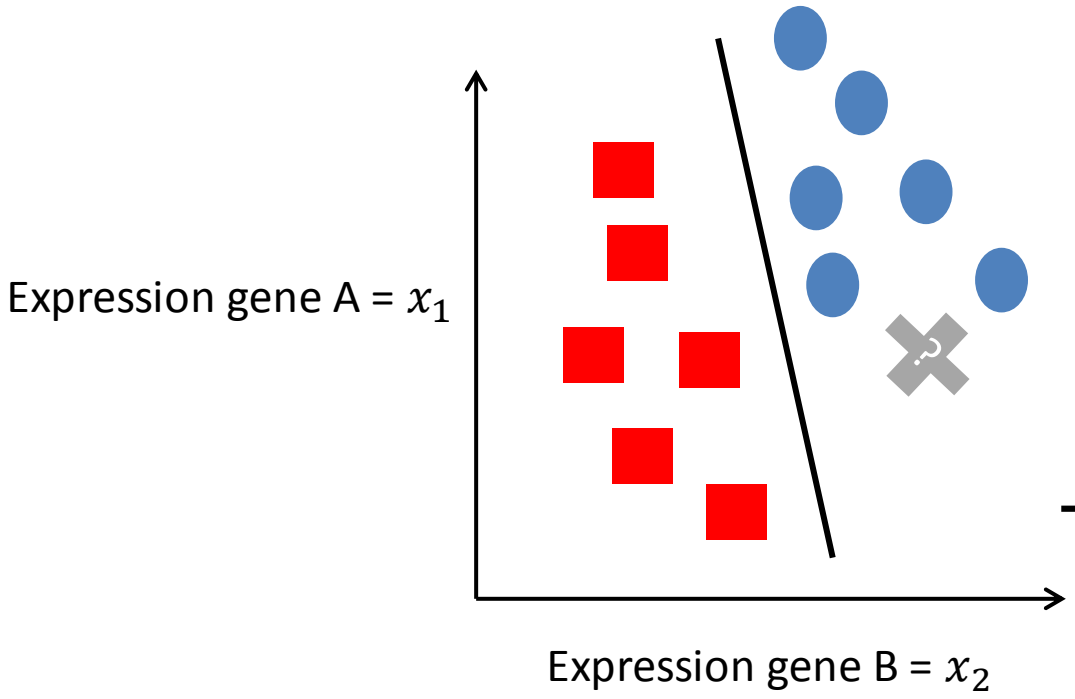
On the data:



Eg: data augmentation (real or artificial), work with distribution…

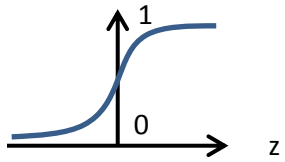
Back to TPBigData.Rmd

Intepret things biologically



Lung cancer

Glioblastoma

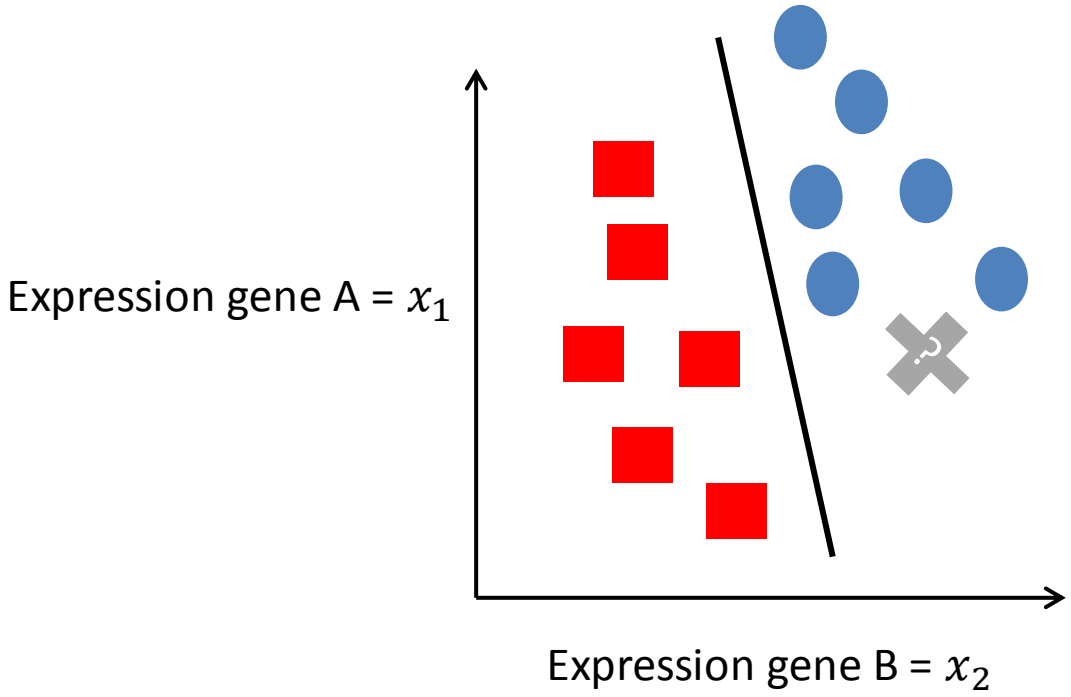


$$z = \theta_1 x_1 + \theta_2 x_2 + b$$

$$h(x) = g(z) = g(\theta_1 x_1 + \theta_2 x_2 + b)$$

→ you have found θ_1, θ_2 & b

Intepret things biologically



Gene A is not very discriminant

- so $abs(\theta_1)$ should be small

Gene B seems discriminant

- so $abs(\theta_2)$ should be larger

Back to TPBigData.Rmd

Lasso regularization or L1N

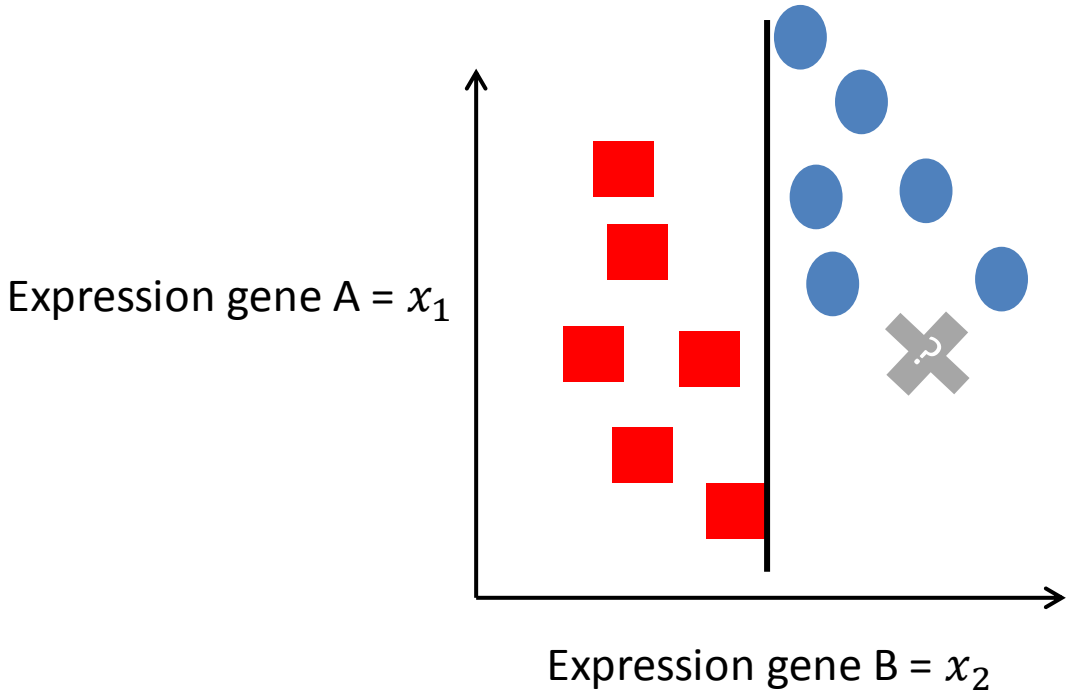
$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} cost\left(h\left(x^i\right), y^i \right) + \boldsymbol{\lambda \sum_{j=1}^{n} \theta_j}$$

→ minimizes $\theta_1, \theta_2 \ \ldots \ \theta_n$

→ And even set some θ very close to zeros

$$z = \theta_1 x_1 + \theta_2 x_2 + b$$
$$h(x) = g(z) = g(\theta_1 x_1 + \theta_2 x_2 + b)$$

With Lasso regularization

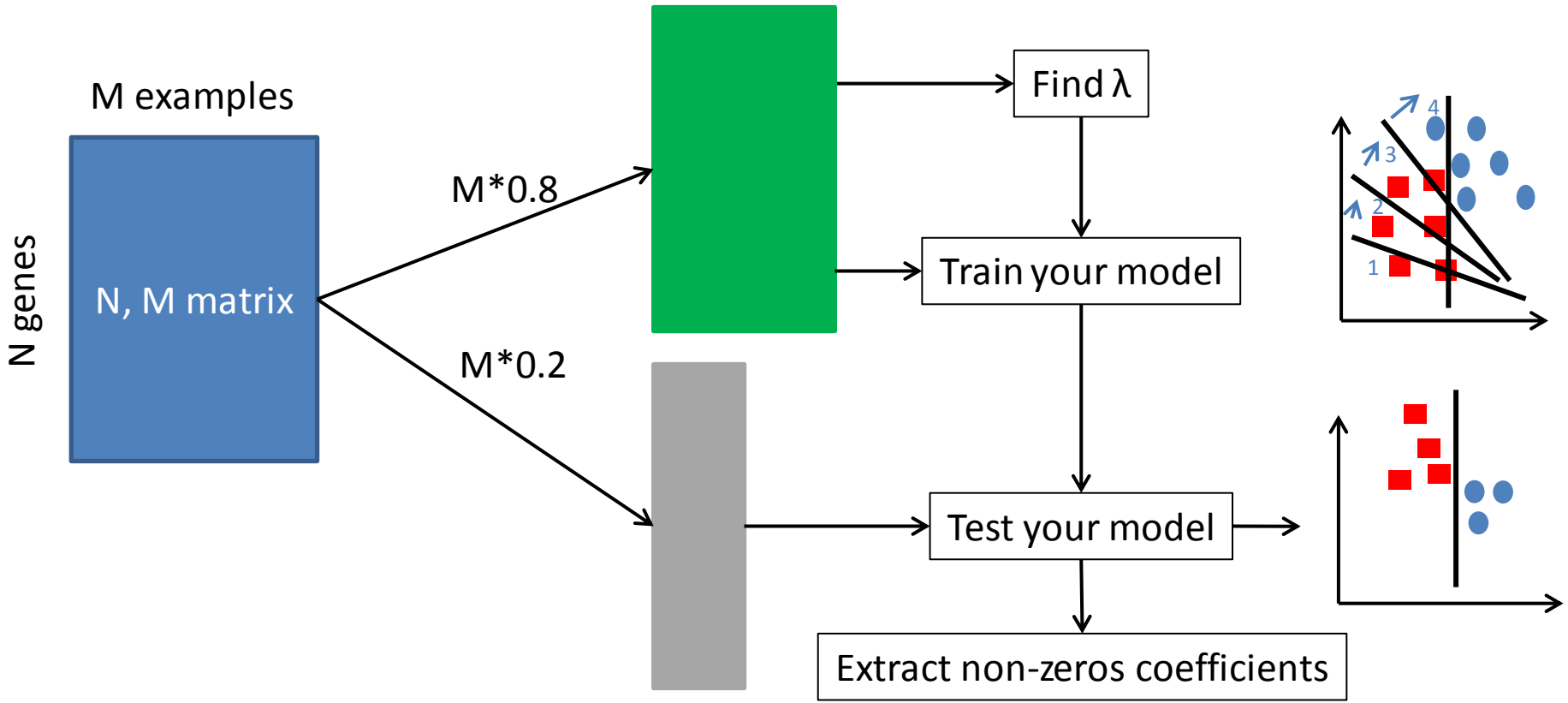


- θ_1 is set to ~0
- $abs(\theta_2)$ is kept large

But you have one more parameter to search = lambda!

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} cost\left(h\left(x^i\right), y^i \right) + \boldsymbol{\lambda} \sum_{j=1}^{n} \theta_j$$

Train and evaluate with Lasso



Find λ ? Do a K fold cross-validation!

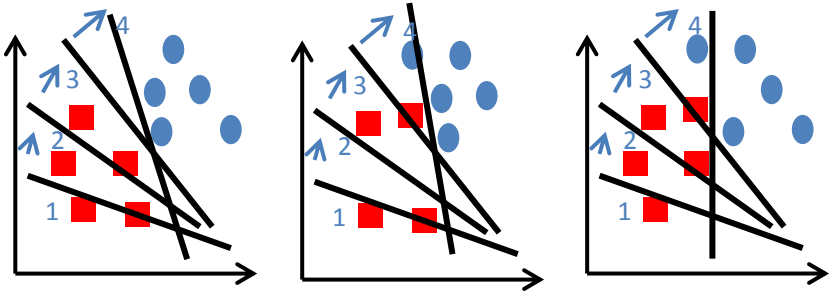
N,
M *0.8

Find λ using
K fold cross-validation

New 80% / 20% split

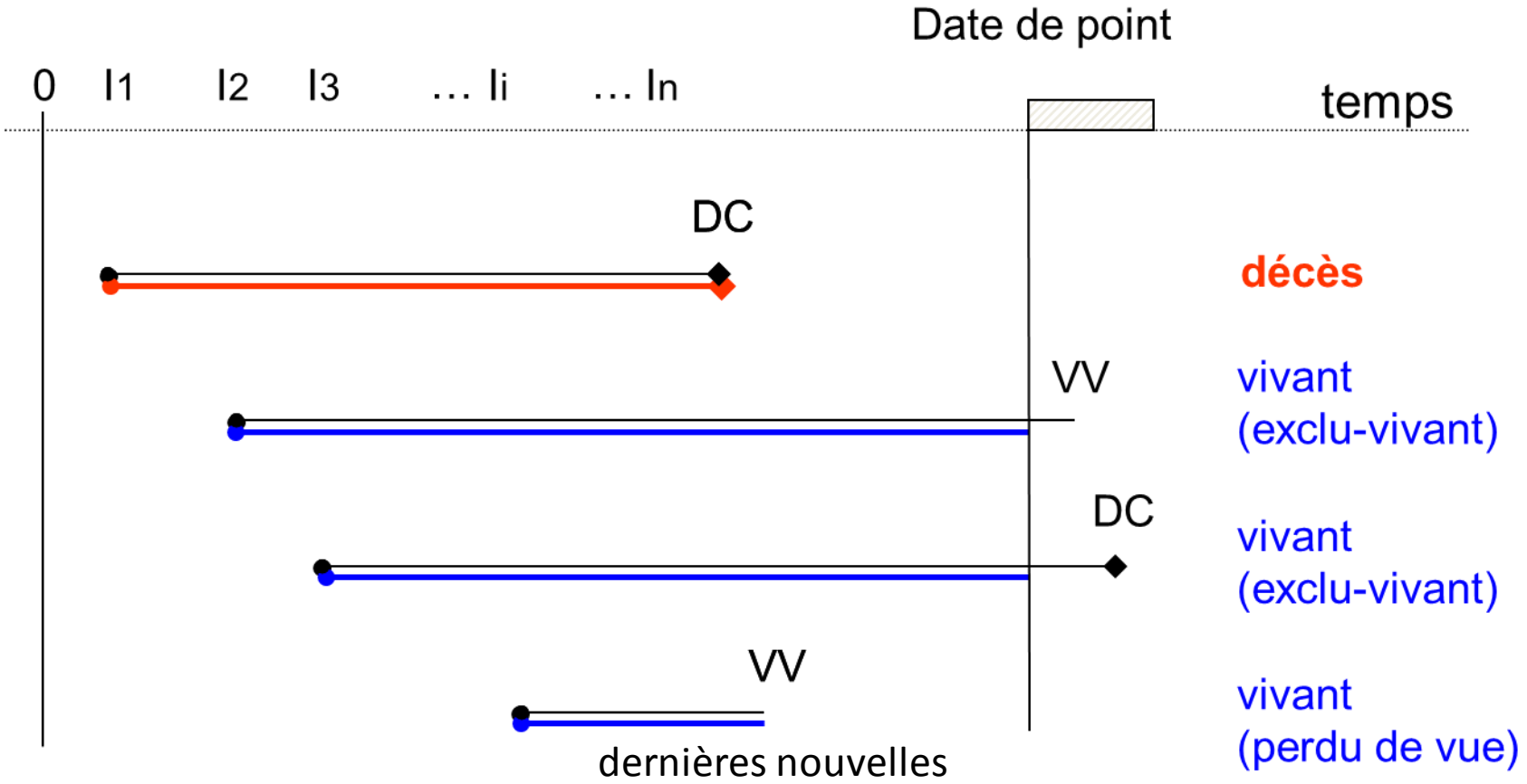
λ = 0.12 λ = 1.4 λ = 2.1 λ = 4.5 …

Keep best λ from
cross-validation set

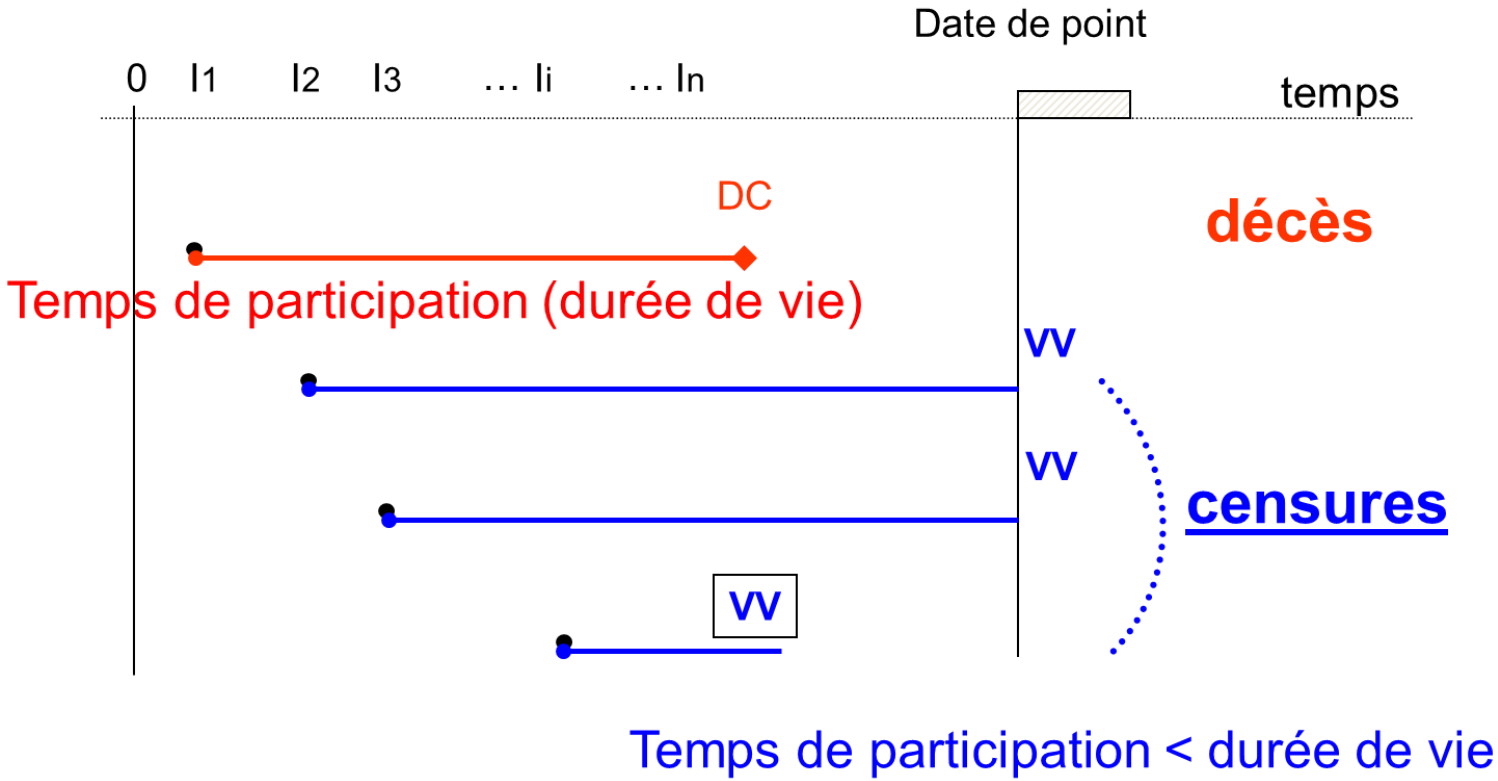


Back to TPBigData.Rmd

La survie : une donnée censurée

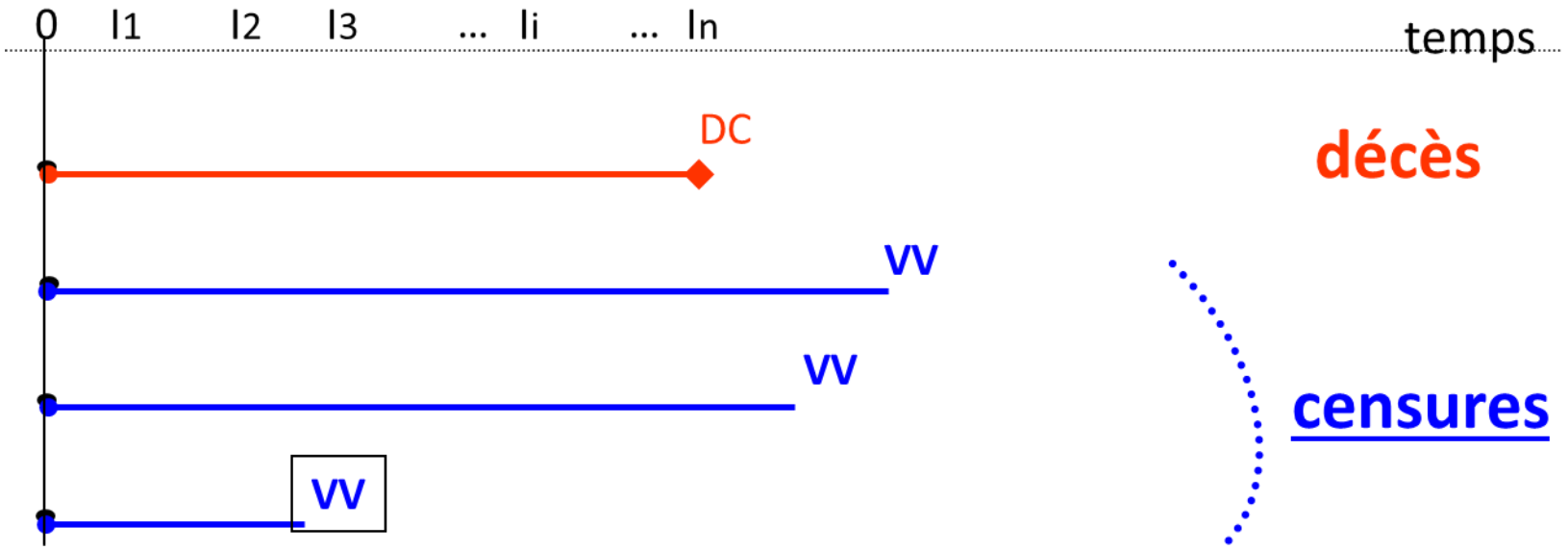


La survie : une donnée censurée



Temps de participation < durée de vie

La survie : une donnée censurée



Modeling Survival

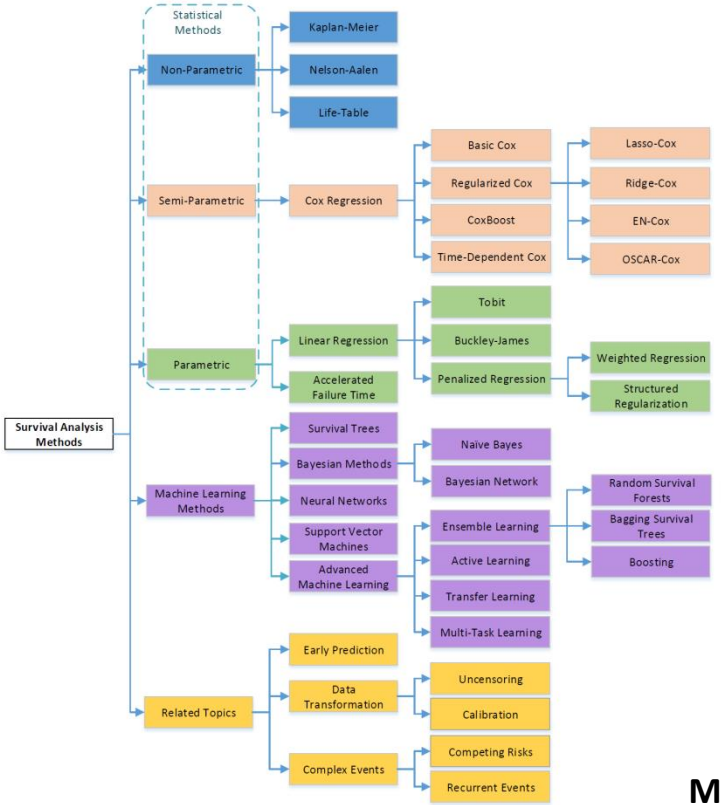
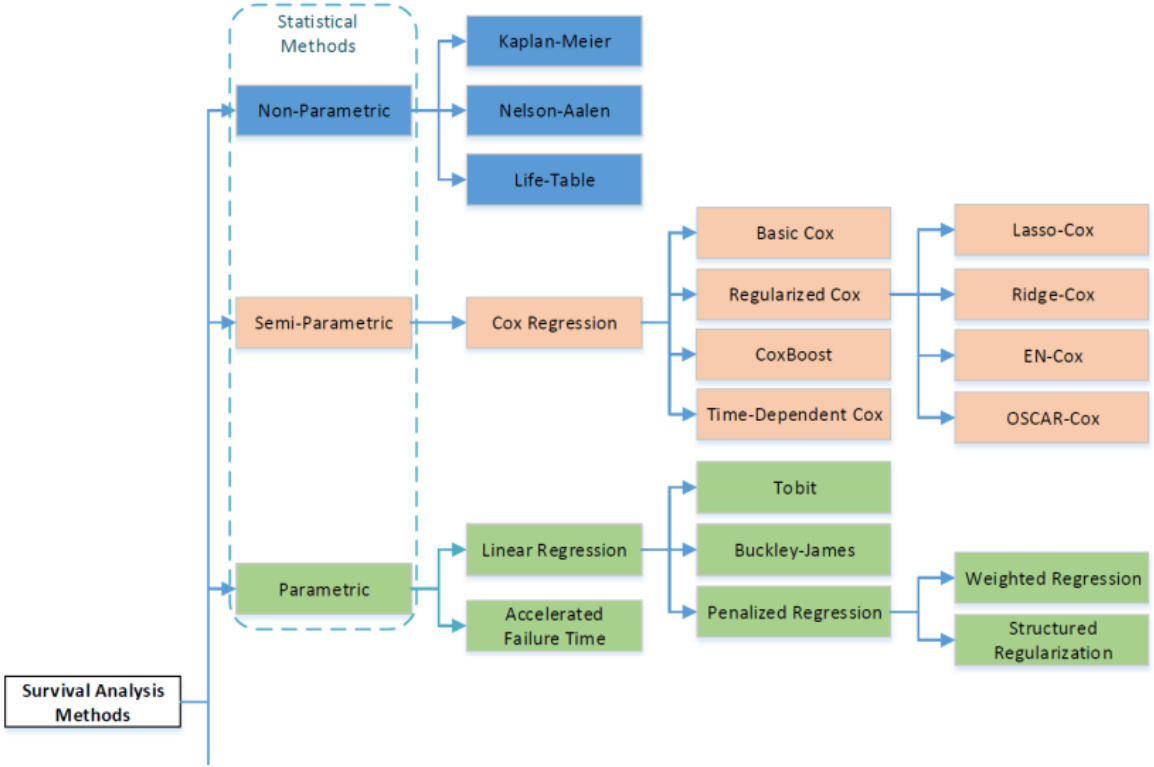


Fig. 3: Taxonomy of the methods developed for survival analysis.

Machine Learning for Survival Analysis: A Survey, PING WANG et al. ArXiv, 2017

Modeling Survival



Modeling Survival

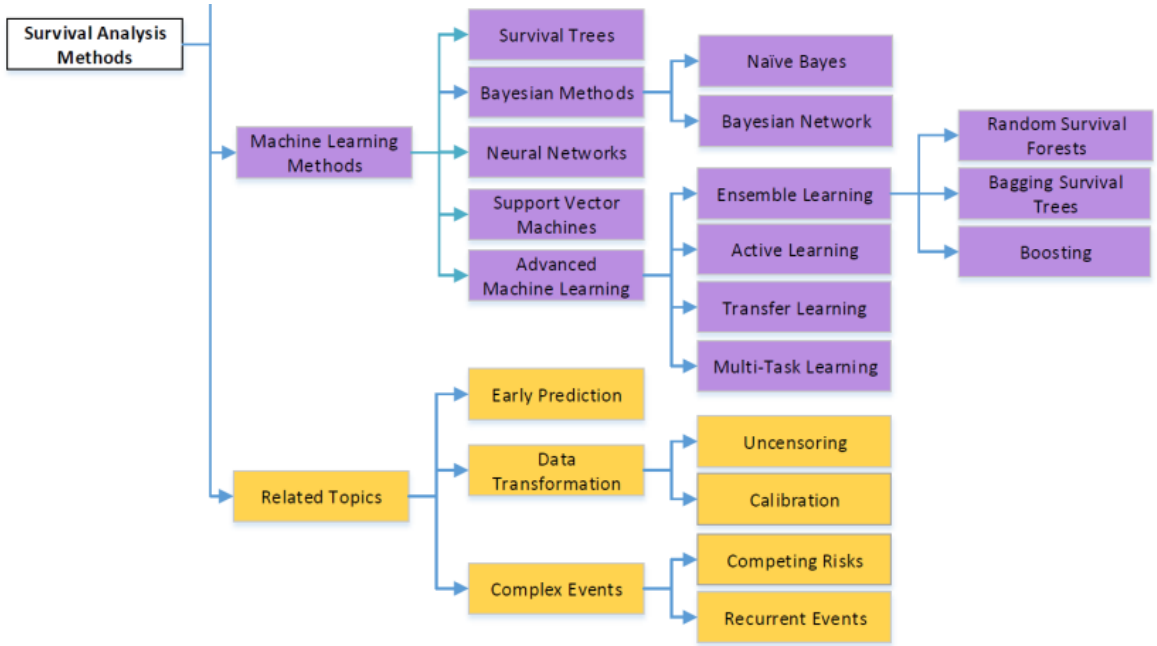


Fig. 3: Taxonomy of the methods developed for survival analysis.

Cox = a semi-parametric model

The knowledge of the underlying distribution of time to event of interest is not required, but the attributes are assumed to have an exponential influence on the outcome.

Individual hazard function:

$$h(t, X_i) = h_0(t)exp(X_i\beta)$$

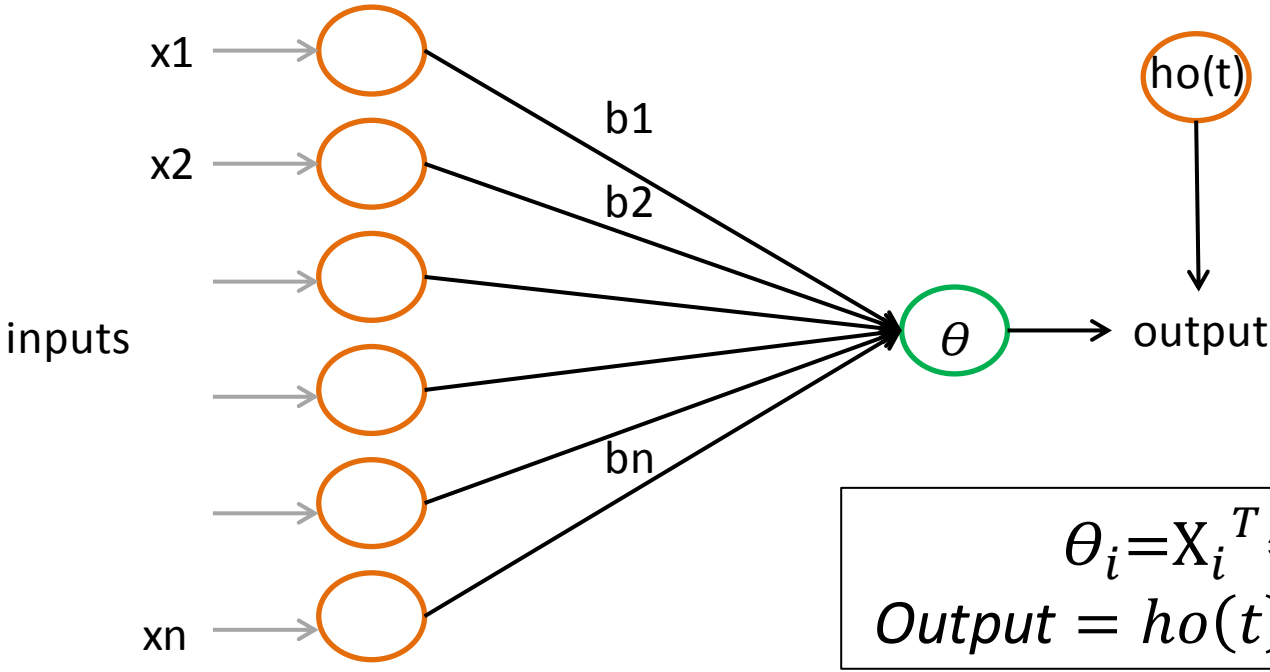
Negative partial log likelyhood to minimize:

$$LL(\beta) = -\sum_{j=1}^{N} \delta_j \{X_j\beta - log[\sum_{i \in R_j} exp(X_i\beta)]\}.$$

N = number of subjects
X = variables
β = parameter to estimate
δ = censoring status
R = the set of risk subjects at Time to event of interest

Machine Learning for Survival Analysis: A Survey, PING WANG et al. ArXiv, 2017

Cox model



$$\theta_i = \mathrm{X}_i^{\,T} * \beta$$

$$Output = ho(t)_i * exp^{\boldsymbol{\theta}_i}$$

C index

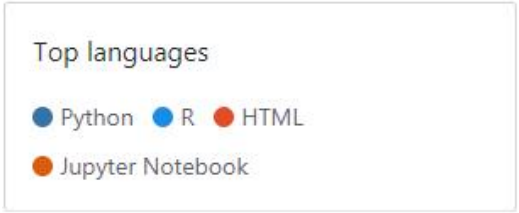
$$c = Pr(\hat{y}_1 > \hat{y}_2 \mid y_1 > y_2)$$

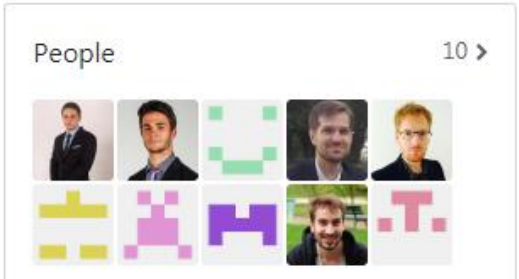
Data Science at DITEP

Info https://github.com/DITEP

Medical team

Data Science team





- 1 Senior Bioinformatician
  Leo Colmet Daage
- Students from CentraleSuppelec
- 1 Student from Telecom Paristech
- For 2019: MSc, PhDc

loicverlingue@yahoo.fr