# Prédiction des intéractions Enhancers-Genes

*Océane Cassan*

*27 mars 2019*

## Contexte

Ce script utilise un jeu de données regroupant des paires enhancers promoters, labélisées suivant leur intéraction ou non dans un type cellulaire précis. Parmis les paires fournies par FANTOM, qui sont données d'une manière globale sur tous les types cellulaires, nous définissons des paires intéragissant dans notre type cellulaire comme les paires FANTOM comprenant un enhancer exprimé et un promoter exprimé dans la matrice d'expression du type cellulaire. Les paires n'intéragissant pas sont définies comme les paires avec un enhancer non exprimé, et un promoter actif. Dans ce contexte, notre problématique se ramène à prédire l'activité ou non d'un enhancer suivant le type cellulaire.

Le script make_datasets.R contient plusieurs fonctions. create_dataset() génère une matrice contenant, pour chaque paire du jeu de données équilibré, les variables numériques passées en argument. Il peut s'agir de :

- La composition nucléotidique
- scores PWM
- Variables du graphe d'exploration de dexter

## Construction du jeu de données d'apprentissage et test

Ici, on choisit les nucléotides comme variables prédictives.

```
source("make_datasets.R")
ct = "CNhs11827"

data <- create_dataset(ct, variables = c("nucl"))
```

```
## [1] 2882   28
```

```
names(data)
```

```
##  [1] "pairs"       "E_AT"        "E_ApA"       "E_ApC"       "E_ApT"
##  [6] "E_CpA"       "E_CpC"       "E_CpG"       "E_CpT"       "E_GC"
## [11] "E_GpC"       "E_TpA"       "E_TpC"       "P_AT"        "P_ApA"
## [16] "P_ApC"       "P_ApT"       "P_CpA"       "P_CpC"       "P_CpG"
## [21] "P_CpT"       "P_GC"        "P_GpC"       "P_TpA"       "P_TpC"
## [26] "chr"         "distance"    "interaction"
```

On sépare en jeu de train et de test, en échantillonnant dans tous les chromosomes. Nous ne voulons pas que des variables soient partagées entre des paires du jeu d'apprentissage et du jeu de test. Ainsi, pour créer le jeu de test, nous piochons successivement dans tous les chromosomes un enhancer et ses enhancers chevauchants.

```
sets <- train_test_split(data)
data_train = sets[[1]]
data_test = sets[[2]]
```

### Construire la régression

Tout d'abord, les jeu de données sont construits

```
source("make_models.R")

model <- train_lasso_model(data_train)
glm <- model[[1]]
variables <- model[[2]]

test_lasso_model(data_test=data_test, glm=glm, data_train=data_train, variables = variables, vars = "nu
```