

## Mélanges et point isoélectrique des protéines

P<sup>r</sup> Jean R. LOBRY

---

Étude de la distribution du point isoélectrique dans un protéome.  
Une simulation pour invalider une interprétation trop jolie pour être vraie. D'après un article de G.F. Weiller, G. Caraux et N. Sylvester (2004) *Proteomics*, 4:943-949.

### Table des matières

<b>1 Introduction</b>	<b>1</b>
<b>2 Le point isoélectrique des protéines</b>	<b>2</b>
<b>3 Distribution du point isoélectrique dans un protéome</b>	<b>4</b>
<b>Références</b>	<b>7</b>

### 1 Introduction

Cette fiche s'utilise en complément de la fiche `tdr221` dont elle reprend toutes les notations. On reprend en particulier la définition de la fonction `logvraineg()` qui retourne la valeur du logarithme de la fonction du maximum de vraisemblance dans le cas d'un mélange de deux lois normales (en fait l'opposé de cette valeur parce que l'on cherche à maximiser la vraisemblance et que la fonction d'optimisation utilisée ici, `nlm()`, cherche à minimiser la valeur de la fonction passée en argument).

```
logvraineg <- fonction(param, obs) {  
  p <- param[1]  
  m1 <- param[2]  
  sd1 <- param[3]  
  m2 <- param[4]  
  sd2 <- param[5]  
  -sum(log(p*dnorm(obs, m1, sd1) + (1 - p)*dnorm(obs, m2, sd2)))  
}
```

Ainsi que la fonction `simulmixnor()` pour simuler un mélange de deux lois normales :

```

simulmixnor <- fonction(n, p, m1, sd1, m2, sd2) {
  n1 <- rbinom(1, n, p)
  x1 <- rnorm(n1, m1, sd1)
  x2 <- rnorm(n-n1, m2, sd2)
  c(x1, x2)
}
    
```

Pour ces deux fonctions, le paramètre  $p$  représente la fréquence relative de la première population dans le mélange des deux populations,  $m1$  et  $m2$  la moyenne pour la première et la deuxième population, respectivement,  $sd1$  et  $sd2$  l'écart-type pour la première et la deuxième population, respectivement.

Cette fiche de TD s'inspire de l'article [6] très amusant de Weiller *et al.* (2004). On reprendra leur notations.

## 2 Le point isoélectrique des protéines

Par définition, le  $pI$  d'une protéine est le  $pH$  pour lequel les charges positives compensent les charges négatives. A ce  $pH$ , la somme des charges pour tous les acides aminés est nulle.

- Dans les protéines il y a quatre groupes ionisables qui peuvent chargés positivement. Ce sont les trois acides aminés lysine (K), arginine (R) et histidine (H) ainsi que l'extrémité N-terminale (N-term).
- Les charges négatives peuvent être portées par les quatre acides aminés tyrosine (Y), cystéine (C), aspartate (D) et glutamate (E) ainsi que par l'extrémité C-terminale (C-term).

Notons  $f^+$  la somme de toutes les charges positives et  $f^-$  la somme de toutes les charges négatives d'une protéine donnée. Ces deux valeurs dépendent du  $pH$  et des  $pK$  des groupes ionisables de la protéine. Notons  $I^+ = \{K, R, H, N-term\}$  l'ensemble des groupes chargés positivement et  $I^- = \{Y, C, D, E, C-term\}$  ceux chargés négativement. Ainsi, la charge positive  $f^+$  et la charge négative  $f^-$  d'une protéine sont donnés par :

$$f^+ = \sum_{i \in I^+} n_i f_i^+ \text{ et } f^- = \sum_{i \in I^-} n_i f_i^- \quad (1)$$

où  $f_i^+$  est la charge élémentaire portée par un acide aminé de type  $i$  et  $n_i$  le nombre total d'acide aminés de type  $i$  dans la protéine considérée. Nous avons  $n_{N-term} = n_{C-term} = 1$ . Par définition, la valeur de  $pI$  d'une protéine est la solution de l'équation :

$$f^+(pH) + f^-(pH) = 0 \quad (2)$$

Pour un acide aminé,  $f_i^+$  ou  $f_i^-$  sont donnés par l'équation d'Henderson-Hasselbach :

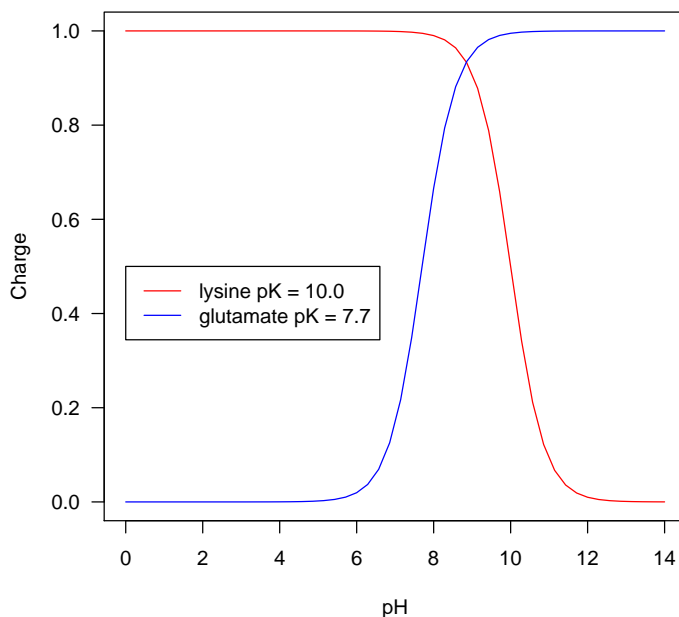
$$f_i^+(pH) = \frac{1}{1 + 10^{pH-pK_i}} \text{ et } f_i^-(pH) = f_i^+(pH) - 1 = \frac{10^{pH-pK_i}}{1 + 10^{pH-pK_i}} \quad (3)$$

avec  $0 \leq f_i^+(pH) \leq 1$  et  $0 \leq f_i^-(pH) \leq 1$ . Toutes les fonctions  $f_i^+$  ont la même allure et peuvent être déduites les unes des autres par simple translation. Ceci est également vrai pour les fonctions  $f_i^-$ . Ce sont des fonctions sigmoïdes du pH avec un point d'inflexion pour  $pH = pK_i$ . C'est un point de symétrie tel que  $f_i^+(pK_i) = \frac{1}{2}$  ou  $f_i^-(pK_i) = \frac{1}{2}$ . Les fonctions  $f_i^+$  décroissent de 1 à 0 avec le pH et réciproquement les fonctions  $f_i^-$  augmentent de 0 à 1 avec le pH. Les courbes ont une zone de variation rapide environ une unité de pH autour de leur pK (c'est la zone de tampon bien connue où une large variation de la charge ne modifie que peu le pH).

A titre d'illustration, représentons les courbes dans le cas de la lysine (pK = 10.0) et du glutamate (pK = 7.7), ces valeurs de pK étant extraites de [2].

```
fp <- fonction(pH, pK) {1/(1+10^(pH-pK))}
fm <- fonction(pH, pK) {10^(pH-pK)/(1+10^(pH-pK))}
pH <- seq(from = 0, to = 14, length = 50)
plot(x = pH, y = sapply(pH, fp, pK = 10.0),
     xlab = "pH", ylab = "Charge", las = 1, type = "l", lty = 1, col = "red",
     main = "Évolution de la charge en fonction du pH")
lines(pH, sapply(pH, fm, pK = 7.7), col = "blue")
legend(0, 0.5, legend=c("lysine pK = 10.0", "glutamate pK = 7.7"), col = c("red", "blue"), lty = 1)
```

Évolution de la charge en fonction du pH



L'équation 2 n'a pas de solution simple. Mais il existe dans le paquet `seqinr` de R une fonction (`computePI()`) permettant de calculer facilement le  $pI$  d'une protéine :

```
library(seqinr)
proteine <- read.fasta(file = system.file("sequences/seqAA.fasta",
     package = "seqinr"), seqtype = "AA")[[1]]
proteine
[1] "M" "P" "R" "L" "F" "S" "Y" "L" "L" "G" "V" "W" "L" "L" "L" "S" "Q" "L" "P" "R"
[21] "E" "I" "P" "G" "Q" "S" "T" "N" "D" "F" "I" "K" "A" "C" "G" "R" "E" "L" "V" "R"
```

```
[41] "L" "W" "V" "E" "I" "C" "G" "S" "V" "S" "W" "G" "R" "T" "A" "L" "S" "L" "E" "E"
[61] "P" "Q" "L" "E" "T" "G" "P" "P" "A" "E" "T" "M" "P" "S" "S" "I" "T" "K" "D" "A"
[81] "E" "I" "L" "K" "M" "M" "L" "E" "F" "V" "P" "N" "L" "P" "Q" "E" "L" "K" "A" "T"
[101] "L" "S" "E" "R" "Q" "P" "S" "L" "R" "E" "L" "Q" "Q" "S" "A" "S" "K" "D" "S" "N"
[121] "L" "N" "F" "E" "E" "F" "K" "K" "I" "I" "L" "N" "R" "Q" "N" "E" "A" "E" "D" "K"
[141] "S" "L" "L" "E" "L" "K" "N" "L" "G" "L" "D" "K" "H" "S" "R" "K" "K" "R" "L" "F"
[161] "R" "M" "T" "L" "S" "E" "K" "C" "C" "Q" "V" "G" "C" "I" "R" "K" "D" "I" "A" "R"
[181] "L" "C" "*"
attr(,"name")
[1] "A06852"
attr(,"Annot")
[1] ">A06852          183 residues"
attr(,"class")
[1] "SeqFastaAA"
computePI(proteine)
[1] 8.534902
```

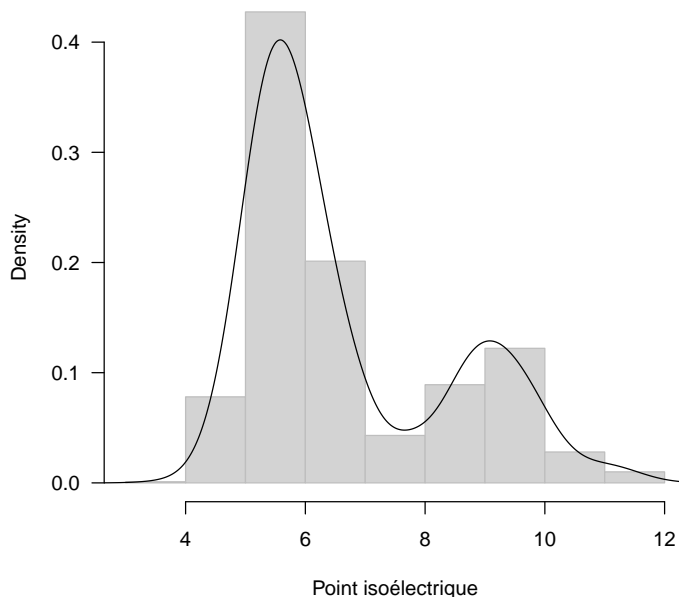
Cette protéine a donc un  $pI$  de 8.53.

### 3 Distribution du point isoélectrique dans un protéome

On s'intéresse à un sous ensemble du protéome de *Escherichia coli* constitué de 999 protéines [3].

```
load(url("http://pbil.univ-lyon1.fr/R/donnees/pIs.Rdata"))
hist(pIs, col = "lightgrey",
     main = "Distribution des points isoélectriques\n pour 999 protéines de Escherichia coli",
     proba = TRUE, las = 1, border = "grey", xlab = "Point isoélectrique")
lines(density(pIs))
```

**Distribution des points isoélectriques  
pour 999 protéines de Escherichia coli**



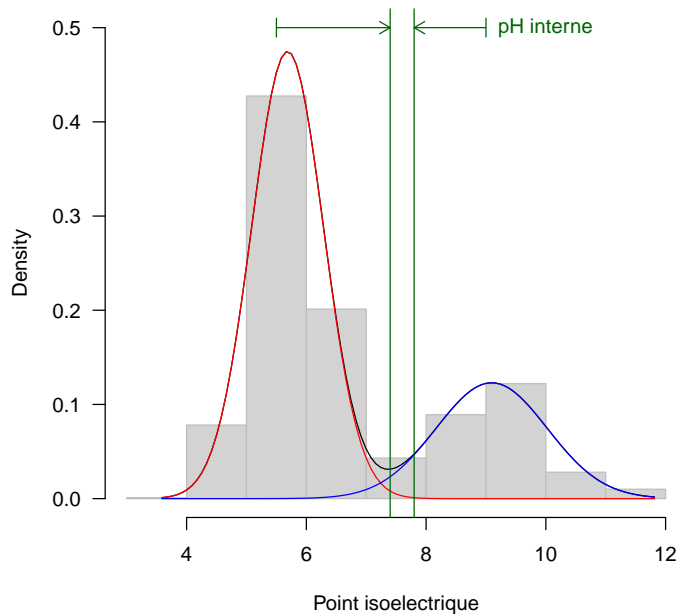
La distribution semble être bimodale. Essayons d'ajuster un mélange de lois normales :

```
estimate <- nlm(f = logvraineg, p = c(0.7, 5.5, 1, 9, 1), obs = pIs)$estimate
estimate
[1] 0.7136124 5.6855310 0.5996581 9.0953438 0.9286524
```

Pour ce jeu de données, la première population représente donc 71.36 % de la population totale, avec un point isoélectrique voisin de 5.69, alors que le point isoélectrique de la deuxième population est voisin de 9.1. Graphiquement :

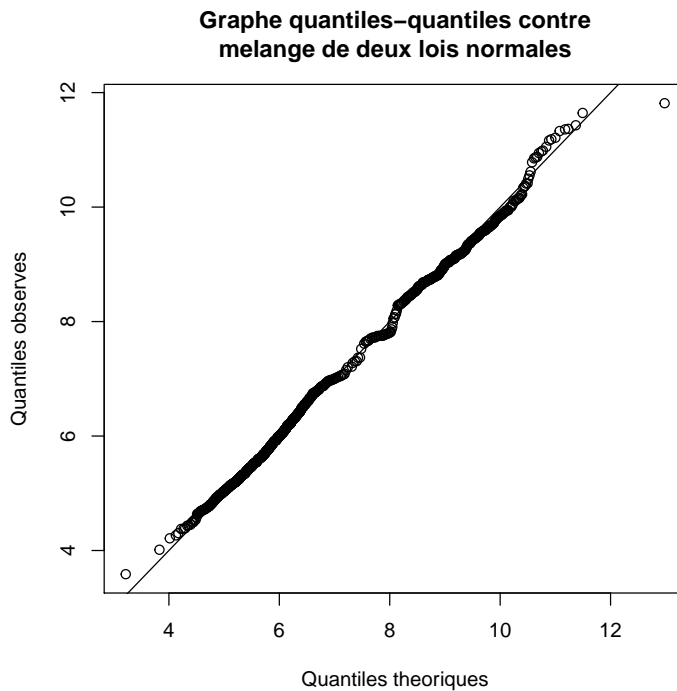
```
x <- seq( min(pIs), max(pIs), length = 100)
y1 <- estimate[1]*dnorm(x, estimate[2], estimate[3])
y2 <- (1 - estimate[1])*dnorm(x, estimate[4], estimate[5])
hist(pIs, col = "lightgrey",
     main = "Distribution des points isoelectriques\n pour 999 protéines de Escherichia coli",
     proba = TRUE, ylim = c(0,0.5), las = 1, border = "grey", xlab = "Point isoelectrique")
lines(x, y1 + y2)
lines(x, y1, col = "red")
lines(x, y2, col = "blue")
abline(v = c(7.4, 7.8), lty = 1, col = "darkgreen")
arrows(x0 = 5.5, y0 = 0.5, x1 = 7.4, y1 = 0.5, length = 0.1, code = 2, col = "darkgreen")
segments(x0 = 5.5, y0 = 0.49, x1 = 5.5, y1 = 0.51, col = "darkgreen")
arrows(x0 = 7.8, y0 = 0.5, x1 = 9, y1 = 0.5, length = 0.1, code = 1, col = "darkgreen")
segments(x0 = 9, y0 = 0.49, x1 = 9, y1 = 0.51, col = "darkgreen")
text(x = 9, y = 0.5, pos = 4, col = "darkgreen", "pH interne")
```

**Distribution des points isoelectriques  
pour 999 protéines de Escherichia coli**



La description de la distribution des points isoélectriques semble plus satisfaisante avec un mélange de deux lois normales qu'avec une seule loi normale. Si on y regarde de plus près avec un graphe quantiles-quantiles :

```
e <- estimate
theo <- simlumnor(10000, e[1], e[2], e[3], e[4], e[5])
qqplot(theo, pIs,
       main="Graphe quantiles-quantiles contre\nmélange de deux lois normales",
       xlab="Quantiles theoriques",ylab="Quantiles observes")
lines(c(0,10000), c(0,10000))
```

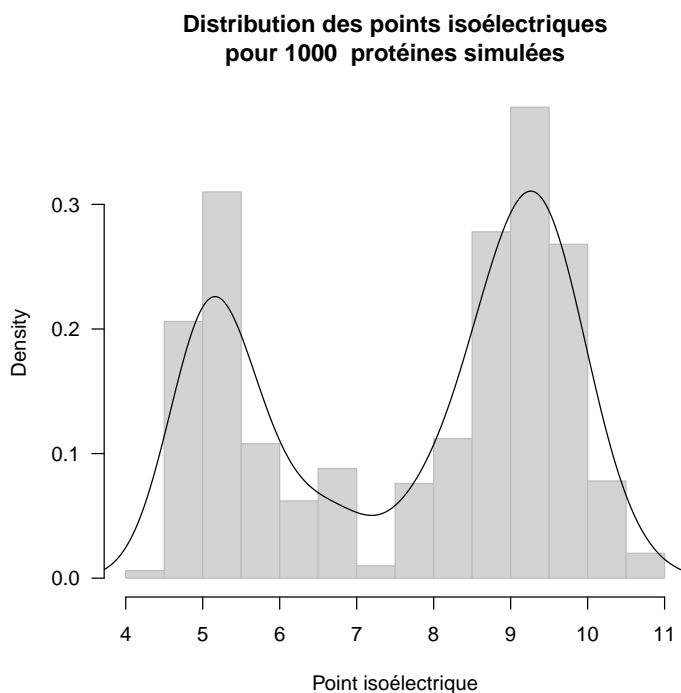


on voit que cette description est assez bonne dans l'ensemble. L'aspect multimodal de la distribution des pIs d'un protéome est une observation assez générale : « Isoelectric point (pI) values have long been a standard measure for distinguishing between proteins. This article analyzes distributions of pI values estimated computationally for all predicted ORFs in a selection of fully sequenced genomes. Histograms of pI values confirm the bimodality that has been observed previously for bacterial and archaeal genomes and reveal a trimodality in eukaryotic genomes.<sup>1</sup> »

Comme les protéines sont en général peu solubles au voisinage de leur point isoélectrique [1], et que le pH du cytoplasme est voisin de 7 [5], il est raisonnable de penser qu'il existe une pression de sélection pour éviter que le point isoélectrique des protéines soit voisin de 7. C'est une explication très raisonnable, mais complètement fautive [6]. Une simple simulation permet de s'en convaincre :

```
n <- 1000
x <- numeric(n)
for(i in 1:n) {x[i] <- computePI(sample(proteine, replace = TRUE))}
hist(x, col = "lightgrey",
     main = paste("Distribution des points isoélectriques\npour", n, " protéines simulées"),
     proba = TRUE, las = 1, border = "grey", xlab = "Point isoélectrique")
lines(density(x))
```

1. Extrait du résumé de [4]



Les distributions multimodales observées sont une conséquence directe des propriétés physico-chimiques des acide aminés (*i.e.* des valeurs des pK des acides aminés des protéines). Il n'y a pas de sélection naturelle qui opère ici.

## Références

- [1] T. Arakawa and S.N. Timasheff. Theory of protein solubility. *Methods in Enzymology*, 114 :49–77, 1985.
- [2] B. Bjellqvist, G.J. Hughes, C. Pasquali, N. Paquet, F. Ravier, J.-C. Sanchez, S. Frutiger, and D.F. Hochstrasser. The focusing positions of polypeptides in immobilized ph gradients can be predicted from their amino acid sequences. *Electrophoresis*, 14 :1023–1031, 1993.
- [3] J.R. Lobry and C. Gautier. Hydrophobicity, expressivity and aromaticity are the major trends of amino-acid usage in 999 *Escherichia coli* chromosome-encoded genes. *Nucleic Acids Research*, 22 :3174–3180, 1994.
- [4] R. Schwartz, C.S. Ting, and King J. Whole proteome pI values correlate with subcellular localizations of proteins for organisms within the three domains of life. *Genome Research*, 11 :703–709, 2001.
- [5] J.L. Slonczewski, B.P. Rosen, J.R. Alger, and R.M. Macnab. pH homeostasis in *Escherichia coli* : Measurement by  $^{31}\text{P}$  nuclear magnetic resonance of methylphosphonate and phosphate. *Proceedings of the National Academy of Sciences of the United States of America*, 78 :6271–6275, 1981.

- [6] G.F. Weiller, G. Caraux, and N. Sylvester. The modal distribution of proteins isoelectric points reflects amino acid properties rather than sequence evolution. *Proteomics*, 4 :943–949, 2004.