TP de modélisation

Maximum de vraisemblance

Recherche des îlots CpG

Dans le génome humain, la méthylation agit sur les dinucléotides CG, en méthylant la cytosine qui peut alors muter en thymine par simple oxydation. Pour cette raison, le taux des dinucléotides CpG [1] est anormalement bas. Néanmoins, il existe des régions dans lesquelles ce processus n'agit pas, et où le taux de CpG est plus élevé qu'alentours, régions nommées " îlots CpG ". Ces régions ont des longueurs de quelques centaines à quelques milliers de bases.

Notre but est d'utiliser une chaîne de Markov d'ordre 1 (dite "modèle M1") pour discriminer des séquences faisant ou non partie d'îlot CpG. L'idée est la suivante : à partir d'un jeu de séquences qui ne contiennent pas d'îlots CpG, on construit un processus de Markov de matrice de transition $\mathcal{M}^-$, qui maximise la vraisemblance sur l'ensemble de ces données. À partir d'un autre de jeu de séquences, ne contenant que des îlots CpG, on calcule un autre modèle de Markov, de matrice de transition $\mathcal{M}^+$, qui maximise la vraisemblance sur ces séquences. Ensuite, une séquence inconnue sera classée îlot CpG si sa vraisemblance est plus grande selon $\mathcal{M}^+$ que selon $\mathcal{M}^-$.

Pour des raisons pratiques, on travaille généralement avec le logarithme de la vraisemblance. (On note que, le logarithme étant une fonction croissante, maximiser la vraisemblance ou la log-vraisemblance revient au même.)

Pour pouvoir comparer les valeurs entre des séquences de longueurs différentes, il faut considérer la log-vraisemblance moyenne par nucléotide, soit pour une séquence $\mathcal{S}$ de longueur $|\mathcal{S}|$:

$$\hat{\mathcal{L}}(\mathcal{S}|\mathcal{M}) = \frac{\mathcal{L}(\mathcal{S}|\mathcal{M})}{|\mathcal{S}|}$$

1 Construction et évaluation des modèles

Allez chercher les données sur :

`ftp://pbil.univ-lyon1.fr/pub/cours/gueguen/lib/TP_Markov_ML/`

Vous disposez des fichiers suivants :

- Pour l'apprentissage : `mus_cpg_app.fa` et `mus_tem_app.fa`
- Pour l'évaluation de la méthode : `mus_cpg_test.fa` et `mus_tem_test.fa`
- Enfin, une séquence inconnue à annoter *de novo* : `mus.fa`

Pour effectuer les comptes et dessins, on utilise la bibliothèque `seqinr` de R [2] :

```
install.packages("seqinr")
```

1. En comptant les mots sur les séquences d'îlots CpG (fichier `mus_cpg_app.fa`), déterminer le processus le plus vraisemblable $\mathcal{M}^+$ sur l'ensemble de ces séquences.
   Voici un exemple de code `R` pour faire cela :

1. On met le "p" pour différencier le dinucléotide de la paire Watson-Crick C-G entre les deux brins de l'ADN.
2. Si vous n'avez pas droit d'écriture dans les répertoires par défaut (probable), essayez :
   `install.packages("seqinr",lib=getwd())`
   `library(seqinr,lib.loc=getwd())`

```
library(seqinr)
# Apprentissage de M-plus
app_cpg_seqs = read.fasta("mus_cpg_app.fa") # les sequences
# On compte les dinucléotides.
app_cpg_ndi = sapply(app_cpg_seqs, count, 2)
app_cpg_ndi_tot = apply(app_cpg_ndi, 1, sum)
# On compte les mononucleotides
app_cpg_nmono = sapply(app_cpg_seqs, count, 1)
app_cpg_nmono_tot = apply(app_cpg_nmono, 1, sum)
# La matrice M-plus
Mp = app_cpg_ndi_tot/rep(app_cpg_nmono_tot, each = 4)
```

2. Faire de même sur les séquences témoins (fichier `mus_tem_app.fa`), pour déterminer le processus le plus vraisemblable $\mathcal{M}^-$ (variable `Mm`).
3. Comparer les deux modèles ainsi obtenus. Sur quel taux de transition les deux modèles diffèrent-ils le plus ? Était-ce attendu ?
4. Calculer les log-vraisemblances de ces processus sur de nouvelles séquences CpG (fichier `mus_cpg_test.fa`) et séquences témoins (fichier `mus_tem_test.fa`), puis les comparer. Exemple de commandes R :

```
test_cpg_seqs = read.fasta("mus_cpg_test.fa")
test_cpg_ndi = sapply(test_cpg_seqs, count, 2)
test_cpg_lnL_p = apply(test_cpg_ndi, 2, function(x) {
      sum(x * log(Mp))/sum(x) # calcul de la vraisemblance sous Mp
    })
test_cpg_lnL_m = apply(test_cpg_ndi, 2, function(x) {
      sum(x * log(Mm))/sum(x) # calcul de la vraisemblance sous Mm
    })
hist(test_cpg_lnL_p - test_cpg_lnL_m, nclass = 50)
abline(v = 0, col = "red")
sum(test_cpg_lnL_p - test_cpg_lnL_m > 0) # nombre de positifs
```

Calculer le nombre de vrais/faux – positifs/négatifs.

prédictions

réalité →	+	-
positifs	VP	FP
négatifs	FN	VN

Quelles sont la sensibilité (taux de vrais positifs) et la spécificité (taux de vrais négatifs) de cette méthode ?

2 Discrimination des îlots CpG

En utilisant la méthode développée à la section précédente, on souhaite identifier d'éventuels d'îlots CpG dans la séquence "inconnue" `mus.fa` (500 000 bp).

1. Dans un premier temps, on découpe cette séquence en 500 fragments de 1000 bp :

```
mus=read.fasta("mus.fa")[[1]] # lecture du fichier
mus1000=list() # liste vide
for(i in 1:500){
    j=i*1000
    mus1000[[i]] = mus[(j-999):j] # remplissage
    }
```

2. En vous inspirant de la question 4, calculez les vraisemblances des 500 fragments sous les modèles $\mathcal{M}^+$ et $\mathcal{M}^-$.

   Quels fragments correspondent à des îlots ? (Vous stockerez leurs index dans la variable `ilots`.)
3. Dessinez ces îlots. Voici un exemple de code `R` pour faire cela :

```
n=length(mus)
plot(
    NULL, # seulement le cadre
    xlim = c(0,n), ylim=c(-1,1),
    main="M+ vs. M- ",
    xlab = "Position",
    ylab="", yaxt="n"
    )
lines(x=c(0,n), y=c(0, 0))
for(i in ilots){
    j=i*1000
    rect(xright=j, xleft=(j-999), ybot=0, ytop=0.1, col="red")
    }
```

3 Évaluation du modèle M1

Dans toute modélisation, il est important de critiquer les modèles que l'on se donne. Dans notre cas, on peut se demander le modèle M1 est vraiment plus intéressant que le modèle M0. Comme on peut déduire M0 de M1 (ou encore l'information que gère M0 est incluse dans celle que gère M1), on s'attend à ce que M1 discrimine mieux que M0.

1. Avec le modèle M0, faire le même travail de discrimination que précédemment. Comparer avec les résultats de M1. Cela était-il prévisible, étant donné ce que l'on cherchait ?
2. Faire de même avec les modèles M2, M3, etc. Qu'est-ce qui statistiquement contraint la profondeur maximale du modèle ?