Définition d'un HMM dans SHOW

Création d'un fichier modulaire constitué de la description successive des différents états cachés.

```
BEGIN_STATE
state_id: intergenic
   BEGIN_TRANSITIONS
      type: 1
      state: start+_1
      ptrans: 0.00432589
      type: 1
      state: stop-_3*1
      ptrans: 0.00426073
      type: 1
      state: stop-_3*2
      ptrans: 0.00089290
      type: 1
      state: intergenic
      ptrans: 0.99052
   END_TRANSITIONS
   BEGIN_OBSERVATIONS
      seq: genomic_dna
      type: 1
      order: 2
      pobs:
            0.314514 0.183587 0.185804 0.316094

            0.385992 0.177465 0.146873 0.28967
            0.327029 0.217127 0.207901 0.247942
            0.310991 0.158126 0.221846 0.309037
            0.23778 0.185229 0.190936 0.386055

            0.44321 0.181248 0.140779 0.234764
            0.33124 0.25435 0.217354 0.197056
            0.382986 0.161845 0.2005 0.25467
            0.303587 0.204104 0.178921 0.313388
            0.38761 0.202831 0.138278 0.27128
            0.342915 0.224382 0.201373 0.231331
            0.290107 0.172784 0.214286 0.322822
            0.255216 0.193496 0.177495 0.373793
            0.326314 0.198495 0.157725 0.317467
            0.260414 0.258744 0.223338 0.257503
            0.22166 0.186155 0.233092 0.359093
            0.184989 0.211447 0.218147 0.385417
            0.334371 0.134578 0.155746 0.375305
            0.345444 0.155348 0.195448 0.30376
            0.328272 0.126919 0.235736 0.309073
            0.204668 0.155128 0.192557 0.447647
   END_OBSERVATIONS
END_STATE
```

Identifiant de l'état

Description des différentes transitions

- type = 1 autorise l'estimation des transitions
  type = 0 transitions non estimées
- state : transition vers l'état cité (ex start+_1)
- ptrans: probabilité de cette transition

Même identifiant que celui donné dans le fichier séquence (-seq)

Order : ordre de la chaîne de Markov pour la distribution des probabilités des émissions

bases

doublets

pobs : valeurs données par l'utilisateur. Si order vaut k $\sum_{t=1}^{k+1} 4^t$ valeurs.

triplets

Si type vaut 0, valeurs utilisées comme probabilité d'émission, si type vaut 1, valeurs utilisées pour initialiser l'algorithme EM. On peut initialiser cette étape d'estimation avec **pobs: random**

Définition d'un HMM dans SHOW

Si on veut prendre en compte différent modèle de composition des gènes : les mots clefs **label** et **tied_to** dans la description des transitions

```
BEGIN_STATE
state_id: cds1+_3
    BEGIN_TRANSITIONS
       label: trans_cds+_3  # identifier is trans_cds+_3
       type: 1
       state: cds1+_1
       ptrans: 0.99
       type: 1
       state: stop+_1
       ptrans: 0.01
    END_TRANSITIONS
    BEGIN_OBSERVATIONS
       seq: genomic_dna
       type: 1
       order: 2
       pobs:    random
       excepted: TGA TAG TAA
    END_OBSERVATIONS
END_STATE

BEGIN_STATE
state_id: cds2+_3
    BEGIN_TRANSITIONS
       tied_to: trans_cds+_3
       state: cds2+_1     # P(cds2+_3 -> cds2+_1) =  P(cds1+_3 -> cds1+_1)
       state: stop+_1     # P(cds2+_3 -> stop+_1) =  P(cds1+_3 -> stop+_1)
    END_TRANSITIONS
    BEGIN_OBSERVATIONS
       seq: genomic_dna
       type: 1
       order: 2
       pobs:    random
       excepted: TGA TAG TAA
    END_OBSERVATIONS
END_STATE
```

label permet de définir un identifiant qui sera utilisé avec le mot clef tied_to

cds1+_3 et cds2+_3 correspondent à la 3ème position du codon dans deux types de composition des séquences codantes. Les transitions de ces états vers les états suivants cités sont liées, elles sont identiques et estimées simultanément. Ceci assure que les probabilités de transition seront les mêmes pour les deux (ou plus) états.

mot clef **excepted** peu être utilisé pour interdire l'émission de certains mots, comme ici les 3 codons stop.

Définition d'un HMM dans SHOW

Utilisation des mots clefs **label** et **tied_to** dans la description des observations

```
BEGIN_STATE
state_id: start+_1
    BEGIN_TRANSITIONS
       type: 0
       state: start+_2
       ptrans: 1
    END_TRANSITIONS
    BEGIN_OBSERVATIONS
       seq: genomic_dna
       label: start+_1_obs
       type: 1
       order: 0
       pobs:
            0.3 0.3 0 0.4 # a, g or t.

    END_OBSERVATIONS
END_STATE

BEGIN_STATE
state_id: start-_1
    BEGIN_TRANSITIONS
       type: 0
       state: intergenic
       ptrans: 1
    END_TRANSITIONS
    BEGIN_OBSERVATIONS
       seq: genomic_dna
       tied_to: start+_1_obs
       type: 3
    END_OBSERVATIONS
END_STATE
```

Utilisation de deux autres valeurs de type qui ne doivent être utilisées que quand les observations sont liées :

type : 2 estimations des probabilités identiques à celles l'observation de référence

type 3 : estimations complémentaires à celles de référence. Ne peut être utilisé que pour un ordre de Markov de 0.

Ex : start+_1 (1ère position du codon start sur brin direct) est la référence identifiée par label :start+_1_obs.
start-_1 (1ère position du codon start brin complémentaire), les probabilités d'émission des caractères de cet état sont liées à celles de l'état start+_1 et ceci par le type 3.
Dans start+_1 :
pobs(*A*) = 0.3, pobs(*G*) = 0.3, pobs(*C*) = 0 et pobs(T) = 0.4
Dans start-_1 on aura par déduction :
pobs(T) = 0.3, pobs(*C*) = 0.3, pobs(*G*) = 0 et pobs(*A*) = 0.4

Modélisation des états BEGIN et END dans SHOW

Deux modélisations distinctes :

- La première permet de travailler en fonction de la longueur de la séquence. Dans ce cas, la longueur de la séquence n'est pas modélisée et la séquence commence et finit dans n'importe quel état caché. Modélisation par défaut.

- La deuxième permet de modéliser la longueur de la séquence. Réalisé en imposant un état dont l'identifiant est **bound**.

L'état **bound** :

- correspond au début et à la fin de la séquence. La séquence démarre à la sortie de l'état bound et finit quand elle atteind l'état bound.
- état silencieux, pas d'émission de caractère
- pour les transitions idem les autres états

Exemple :

```
BEGIN_STATE
state_id: bound
      BEGIN_TRANSITIONS
            type: 1
            state: motif_1_1
            ptrans:0.5
            type: 1
            state: motif_1_2
            ptrans:0.5
      END_TRANSITIONS
END_STATE
```

.....

```
BEGIN_STATE
state_id: motif_1_10
      BEGIN_TRANSITIONS
            type: 0
            state: bound
            ptrans:1
      END_TRANSITIONS
      BEGIN_OBSERVATIONS
            seq: genomic_dna
            type: 1
            order: 0
            pobs: random
      END_OBSERVATIONS
END_STATE
```

Construction du fichier séquence (SHOW)

Les exécutables de SHOW peuvent travailler soit sur une seule séquence, soit sur un ensemble de séquences. Les séquences à analyser sont référencer dans le fichier –**seq <file>**. Ce fichier doit suivre la structure suivante :

```
seq_identifier: genomic_dna
seq_type: dna
seq_files:
          Contig1.dna
          Contig2.dna
```

seq_identifier fait référence à un identifiant choisi pour la (les) séquence(s). Il doit être le même que celui donné dans le fichier modèle (-model) pour le mot clef **seq** au niveau de la description des observations.

seq_type correspond à la nature des séquences (pour le moment seulement adn)

seq_files correspond aux noms des fichiers contenant les séquences à analyser (format Fasta et GenBank accepté). Si plusieurs séquences, soit un seul fichier Fasta avec toutes les séquences, soit un fichier par séquence.

Mise en pratique de SHOW

Estimation des paramètres : **show_emfit** (EM alogorithme/Baum-Welch algorithme

Fichiers d'entrée :

- -model <file> : le fichier contenant la description du modèle
- -seq <file> : le fichier contenant la description des séquences à analyser
- -em <file> : le fichier contenant les informations pour initialiser et faire tourner l'algorithme

Fichier –em <file> :

```
estep-segment: 2000
estep_overlap: 100
nb_sel: 3
niter_sel: 100
eps_sel: 0.01
niter: 1000
epsi: 0.001
```

Taille des segments utilisée pour la mémoire sauvant les approximations lors de l'étape forward-backward de l'algorithme

seulement quand pobs: random

niter et **epsi** : critère d'arrêt. L'algorithme stop quand l'augmentation de la vraisemblance entre deux itérations est plus petite que la valeur de epsi ou quand le nombre maximal d'itérations défini par niter est atteint.
nb_sel correspond au nombre de point de départ aléatoire de l 'algorithme EM, **niter_sel** se rapporte au nombre maximal d'itérations réalisé par point de départ. **eps_sel** correspond au critère d'arrêt de l'algorithme pour chaque point de départ.

Création d'un HMM pour identifier les promoteurs de *B. subtilis*

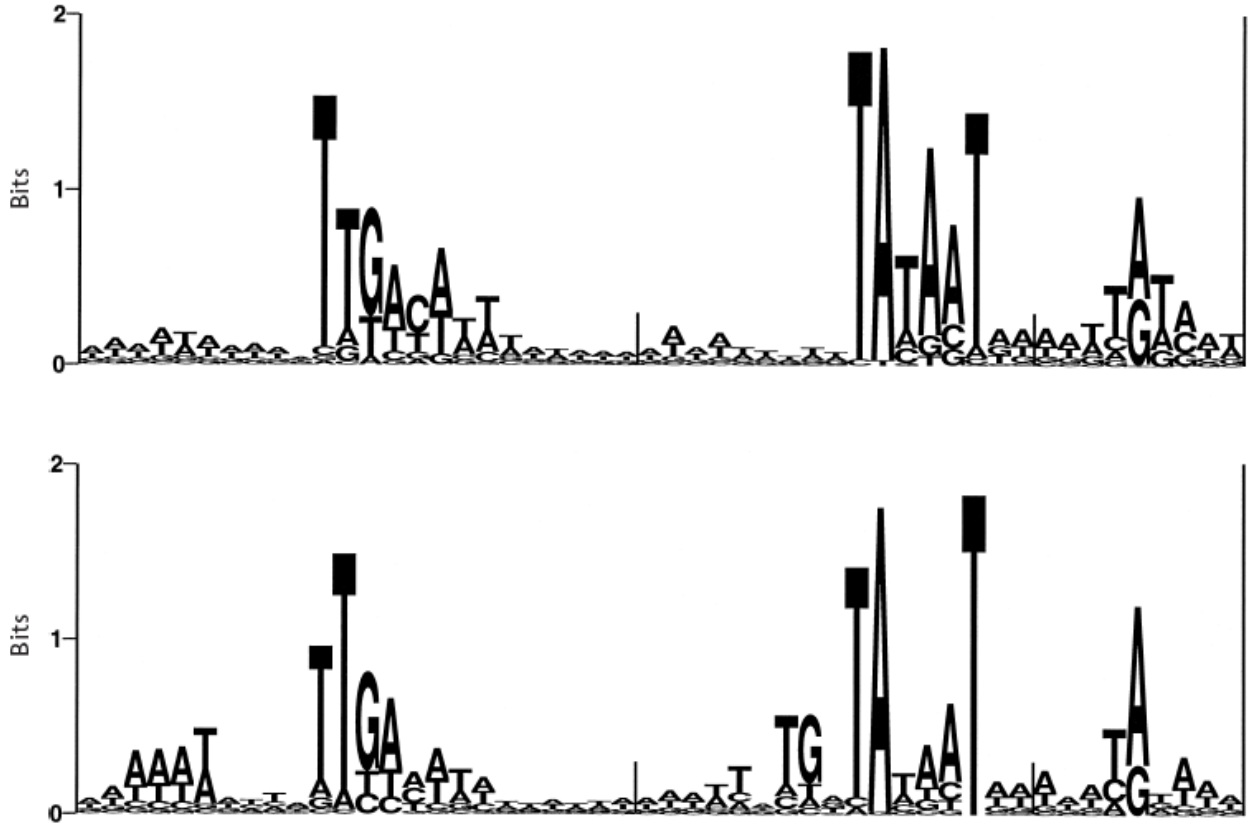

Fig. 4. Logos of the predicted σ^A-binding sites in *B. subtilis*. The logos shown are merged from six individual logos (the merging positions are shown in the figure by horizontal bars), each containing either the −10 or the −35 signal from either the normal (top logo) and the extended (bottom logo) type of σ^A recognition sites. Each of the −10 and the −35 signals represents approximately 500 signals predicted by the HMM. Each of the +1 logos is generated from 350 predicted signals. The logos are constructed by aligning the six types of signals on the first base of the reported signal. For the +1 signal, this base is represented by the highest peak in that area, with an A on the top in both types of binding site. The Shannon information content is shown on the *y* axis; Shannon's unit of non-randomness is the bit (short for 'binary digit') (Shannon, 1948).