Données séquentielles : Pratique de la fouille de séquences à l'aide de TraMineR

Gilbert Ritschard
Alexis Gabadinho, Nicolas S. Müller, Matthias Studer

Département d'économétrie et Laboratoire de Démographie, Université de Genève
http://mephisto.unige.ch/biomining

Tutoriel EGC, Strasbourg, 27 janvier 2009

TraMineR

TraMineR est un package pour R qui permet de :

- Manipuler et transformer différents formats et types de données longitudinales.
- Visualiser et analyser des séquences d'états.
- Fouiller et analyser des séquences d'événements.
- Analyser des dissimilarités (analyse de pseudo-variances et arbre d'induction).
- Les résultats peuvent ensuite être utilisés dans d'autres procédures de R.

Installation de TraMineR

- Pour installer TraMineR, il faut installer R au préalable.
- Pour cela :
  - Téléchargez R depuis http://stat.ethz.ch/CRAN/bin/windows/base/release.htm.
  - Installez R.
  - Lancez R.
- Ensuite, sur la ligne de commande tapez

```
R> install.packages("TraMineR")
```

- Pour utiliser TraMineR, il faut charger la librairie avec

```
R> library("TraMineR")
```


Un court exemple

- Basé sur les données de McVicar and Anyadike-Danes (2002) distribuées avec TraMineR.
- Les données concernent les séquences de transition entre école et emploi.

McVicar, D. and M. Anyadike-Danes (2002). Predicting successful and unsuccessful transitions from school to work using sequence methods. *Journal of the Royal Statistical Society A 165*(2), 317–334.

Création des séquences d'états

Pour manipuler les séquences d'états, il est nécessaire de définir un objet spécifique à l'aide de la fonction **seqdef**. Cette opération permet de :

- définir des méthodes (fonctions) spécifiques aux séquences d'états.
- stocker des informations supplémentaires, soit :
  - Les séquences individuelles sous la forme d'une matrice.
  - L'alphabet (liste des états possibles).
  - Les couleurs associées aux états.
  - Les étiquettes des états.

```
R> data(mvad)
R> mvad.lab <- seqstatl(mvad[, 17:86])
R> mvad.scode <- c("EM", "FE", "HE", "JL", "SC", "TR")
R> mvad.seq <- seqdef(mvad[, 17:86], states = mvad.scode,
+      labels = mvad.lab)
```


Analyses des séquences d'états

- Visualisation :
  - seqdplot Distribution transversale des états (d-plot)
  - seqiplot Plot de séquences individuelles (i-plot)
  - seqfplot Plot des séquences fréquentes (f-plot)
  - seqmtplot Temps moyen passé dans chaque état
- Caractéristiques des séquences individuelles :
  - seqistatd Durée des états
  - seqient Entropie des séquences
  - seqST Turbulence
- Caractéristiques d'un ensemble de séquences :
  - seqstatd Fréquences des états et entropie par âge
  - seqtab Séquences fréquentes
- Calcul de distance entre séquences :
  - seqdist Différentes métriques

Visualisation des séquences d'états

- Représentation graphique des dix premières séquences

```
R> seqiplot(mvad.seq, withlegend = FALSE,
+       title = "Dix premieres sequences")
```

- Graphique des dix séquences les plus fréquentes (avec largeur des barres correspondant à la fréquence)

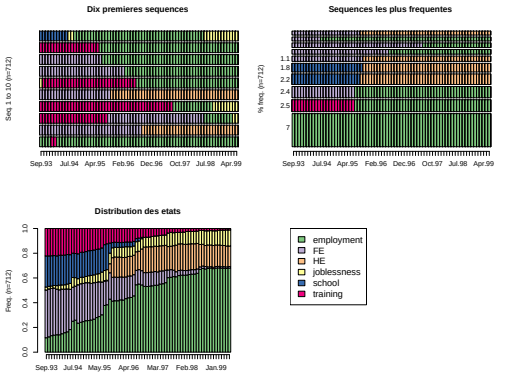
```
R> seqfplot(mvad.seq, pbarw = T, withlegend = F,
+       title = "Sequences les plus frequentes")
```

- Graphique de la distribution des états à chaque unité de temps.

```
R> seqdplot(mvad.seq, withlegend = F,
+       title = "Distribution des etats")
```

- Légendes des graphiques.

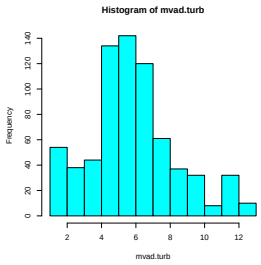
```
R> seqlegend(mvad.seq, fontsize = 1.3)
```

Turbulence des séquences

```
R> mvad.turb <- seqST(mvad.seq)
R> summary(mvad.turb)
```

```
R> hist(mvad.turb, col = "cyan")
```



Optimal matching

- Calcul des distances basé sur les taux de transition entre états.

```
R> submat <- seqsubm(mvad.seq, method = "TRATE")
R> dist.om1 <- seqdist(mvad.seq, method = "OM", indel = 1,
+       sm = submat)
```

- Construction d'une typologie des séquences sur la base des distances

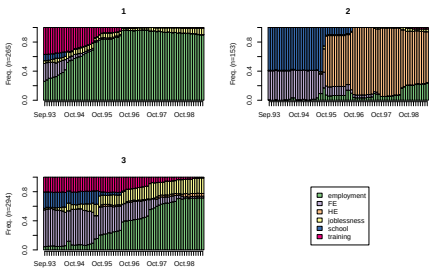
```
R> library(cluster)
```

```
R> clusterward1 <- agnes(dist.om1, diss = TRUE, method = "ward")
```

```
R> plot(clusterward1)
R> cl1.3 <- cutree(clusterward1, k = 3)
```


Graphique de la distribution des états dans chaque cluster

```
R> seqdplot(mvad.seq, group = cl1.3)
```



Séquences d'événements

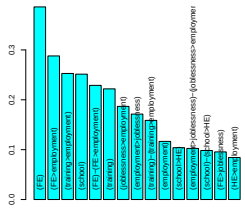
- Il faut définir un objet séquences d'événements à l'aide de la fonction **seqecreate**.

```
R> mvad.seqe <- seqecreate(mvad.seq)
```

- Cet objet nous permettra
  - seqefsub D'extraire les sous-séquences typiques d'événements
  - seqecmpgroup D'identifier les sous-séquences les plus discriminantes
  - seqeapplysub De construire une matrice des occurences des sous-séquences dans chaque séquence d'événements

Recherche de sous-séquences fréquentes

- Recherche des sous-séquences fréquentes au seuil de 0.05%.
  ```
  R> fsubseq <- seqefsub(mvad.seqe, pMinSupport = 0.05)
  ```
- Graphiques des fréquences des 15 sous-séquences les plus fréquentes.
  ```
  R> plot(fsubseq[1:15], col = "cyan")
  ```



Sous-séquences discriminantes

- Recherche des sous-séquences qui discriminent le plus l'appartenance aux clusters.
  ```
  R> discr <- seqecmpgroup(fsubseq, group = cl1.3)
  ```
- Graphique des sous-séquences discriminantes.
  ```
  R> plot(discr[1:6])
  ```