Données séquentielles: concepts et principe d'anaylse

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

Plan

Plan

Structure de la section

Objectifs

- Concepts et questionnements propres à l'analyse de données séquentielles catégorielles
- Types de séquences : avec ou sans dimension temporelle, états, transitions, événements.
- Principes de l'analyse de séquences
  - approche exploratoire
  - approche "explicative", "prédictive"
- Pratique de l'analyse de séquences (TraMineR)

Objectifs

- Concepts et questionnements propres à l'analyse de données séquentielles catégorielles
- Types de séquences : avec ou sans dimension temporelle, états, transitions, événements.
- Principes de l'analyse de séquences
  - approche exploratoire
  - approche "explicative", "prédictive"
- Pratique de l'analyse de séquences (TraMineR)

Objectifs

- Concepts et questionnements propres à l'analyse de données séquentielles catégorielles
- Types de séquences : avec ou sans dimension temporelle, états, transitions, événements.
- Principes de l'analyse de séquences
  - approche exploratoire
  - approche "explicative", "prédictive"
- Pratique de l'analyse de séquences (TraMineR)

Objectifs

- Concepts et questionnements propres à l'analyse de données séquentielles catégorielles
- Types de séquences : avec ou sans dimension temporelle, états, transitions, événements.
- Principes de l'analyse de séquences
  - approche exploratoire
  - approche "explicative", "prédictive"
- Pratique de l'analyse de séquences (TraMineR)

L'équipe intervenante

Le module est donné par l'équipe du projet FN

- Mining event histories : Towards new insights on personal Swiss life courses

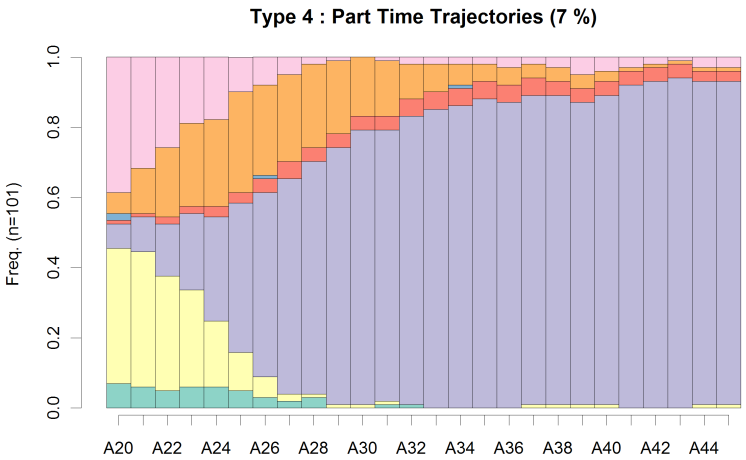
- Project FN 100012-113998 and FN-100015-122230
- Start : February 1, 2007
  End : January 31, 2011

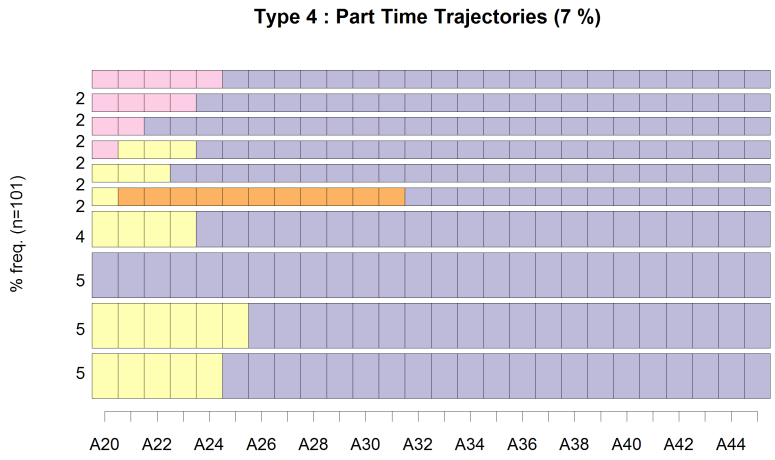
Les intervenants

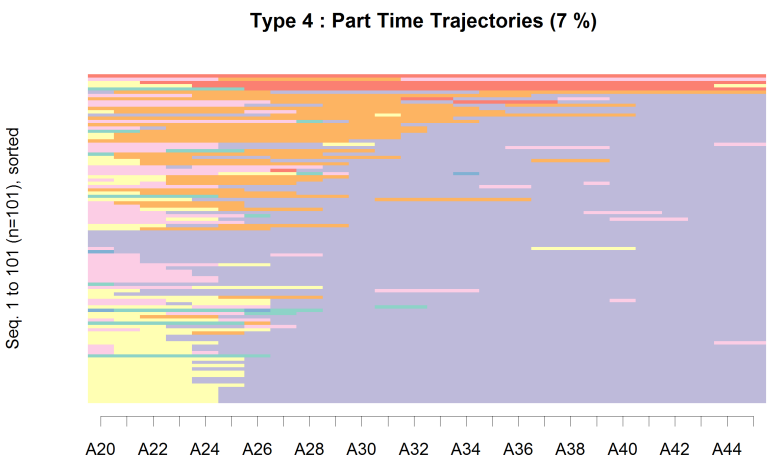
- Gilbert Ritschard, professeur de Statistique pour sc. sociales SES, U. de Genève
- Alexis Gabadinho, Démographie
- Nicolas S. Müller, Sociologie, Système d'information
- Matthias Studer, Economie, Sociologie

Structure de la section

Visualisation de séquences







Caractérisation d'un ensemble de séquences

- Séquence de mesures transversales (entropie inter, ...)

id	t_1	t_2	t_3	$\cdots$
1	B	B	D	$\cdots$
2	A	B	C	$\cdots$
3	B	B	A	$\cdots$

- Résumé de mesures longitudinales (entropie de chaque séquence, taux de transition, ...)

id	t_1	t_2	t_3	$\cdots$
1	B	B	D	$\cdots$
2	A	B	C	$\cdots$
3	B	B	A	$\cdots$

- Autres caractéristiques globales : Séquence centrotype, diversité des séquences, ...

Caractérisation d'un ensemble de séquences

- Séquence de mesures transversales (entropie inter, ...)

id	t_1	t_2	t_3	$\cdots$
1	B	B	D	$\cdots$
2	A	B	C	$\cdots$
3	B	B	A	$\cdots$

- Résumé de mesures longitudinales (entropie de chaque séquence, taux de transition, ...)

id	t_1	t_2	t_3	$\cdots$
1	B	B	D	$\cdots$
2	A	B	C	$\cdots$
3	B	B	A	$\cdots$

- Autres caractéristiques globales : Séquence centrotype, diversité des séquences, ...

Caractérisation d'un ensemble de séquences

- Séquence de mesures transversales (entropie inter, ...)

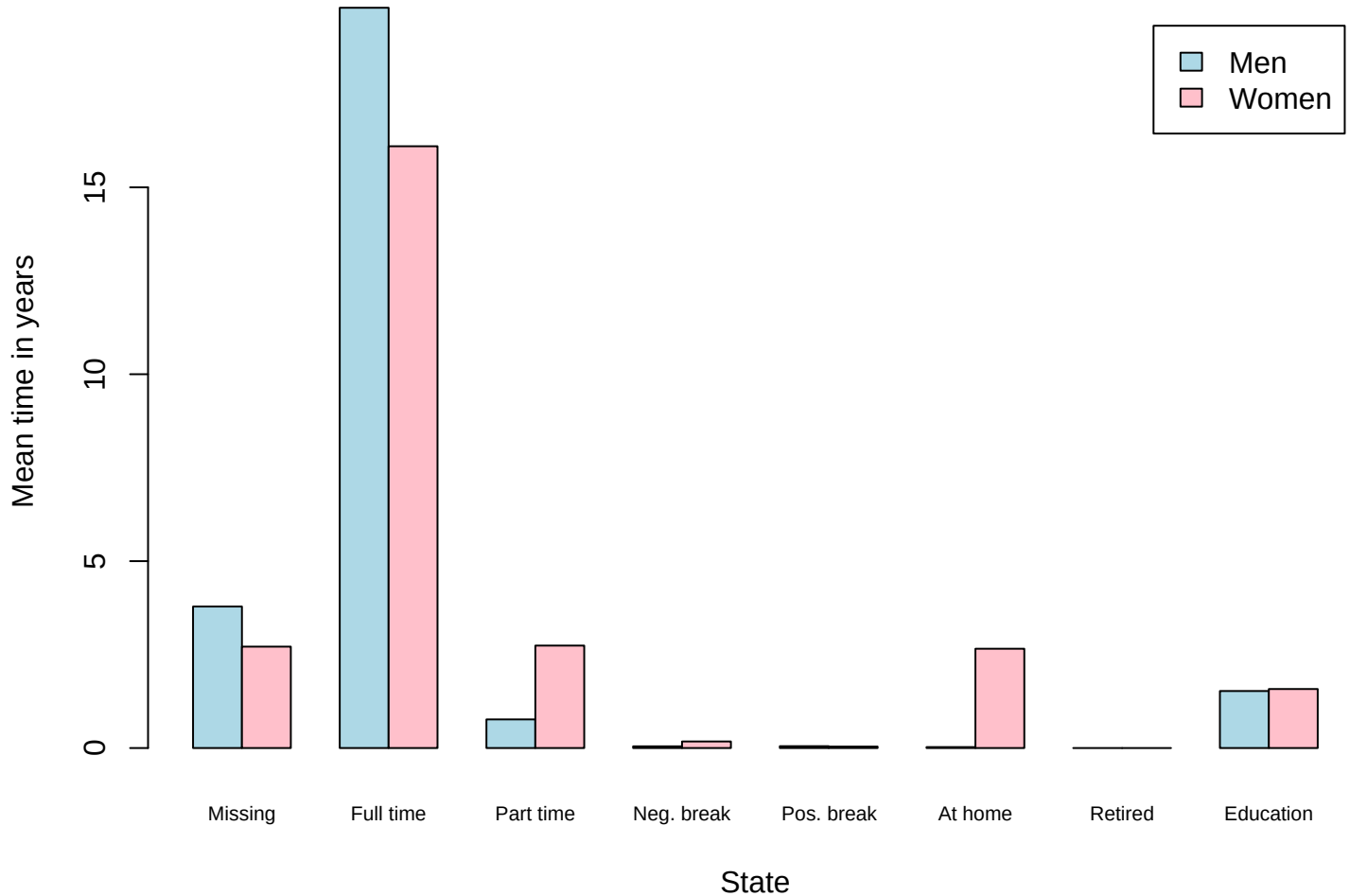
id	t_1	t_2	t_3	$\cdots$
1	B	B	D	$\cdots$
2	A	B	C	$\cdots$
3	B	B	A	$\cdots$

- Résumé de mesures longitudinales (entropie de chaque séquence, taux de transition, ...)

id	t_1	t_2	t_3	$\cdots$
1	B	B	D	$\cdots$
2	A	B	C	$\cdots$
3	B	B	A	$\cdots$

- Autres caractéristiques globales : Séquence centrotype, diversité des séquences, ...

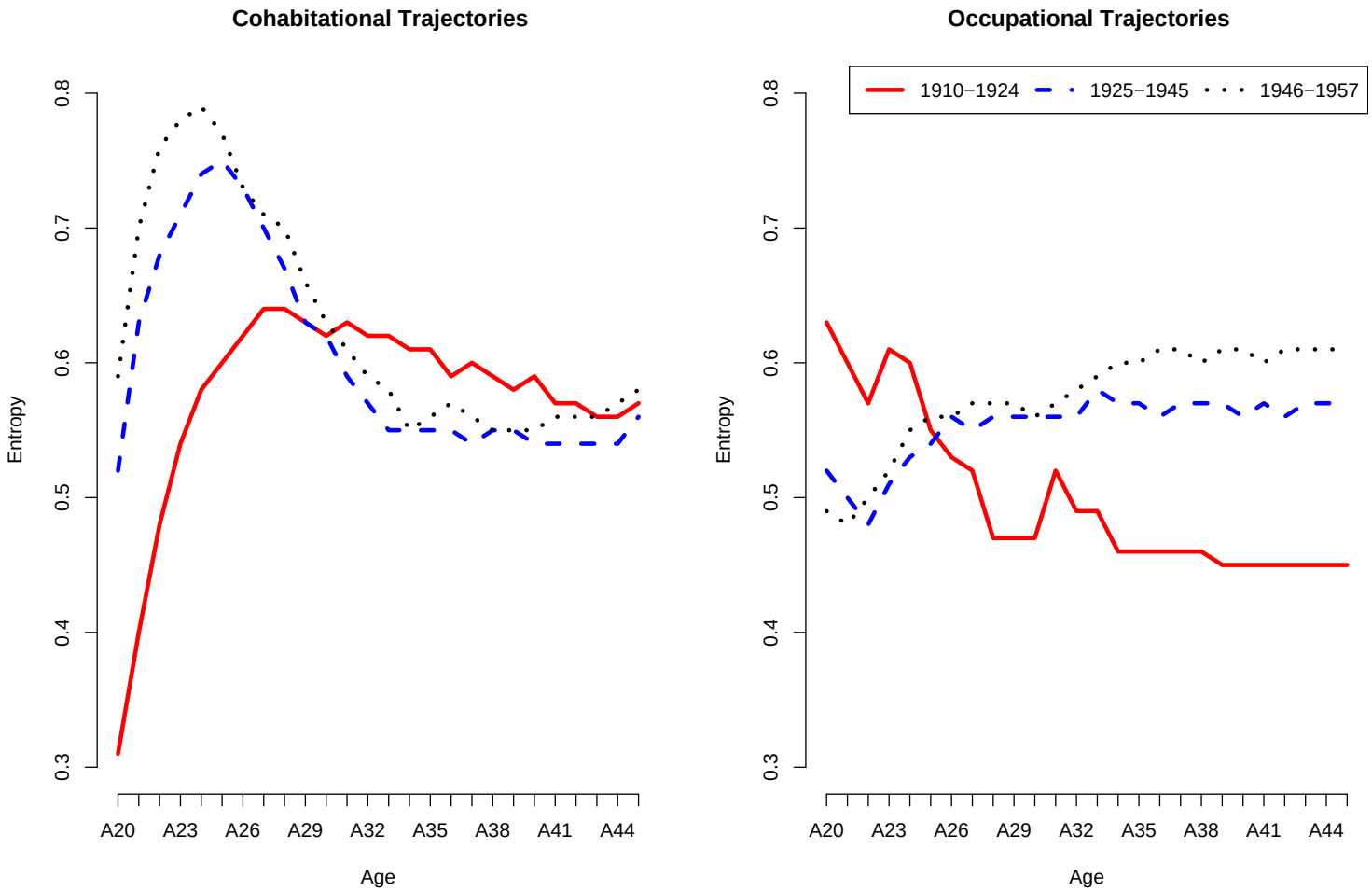
Temps moyen dans chaque état



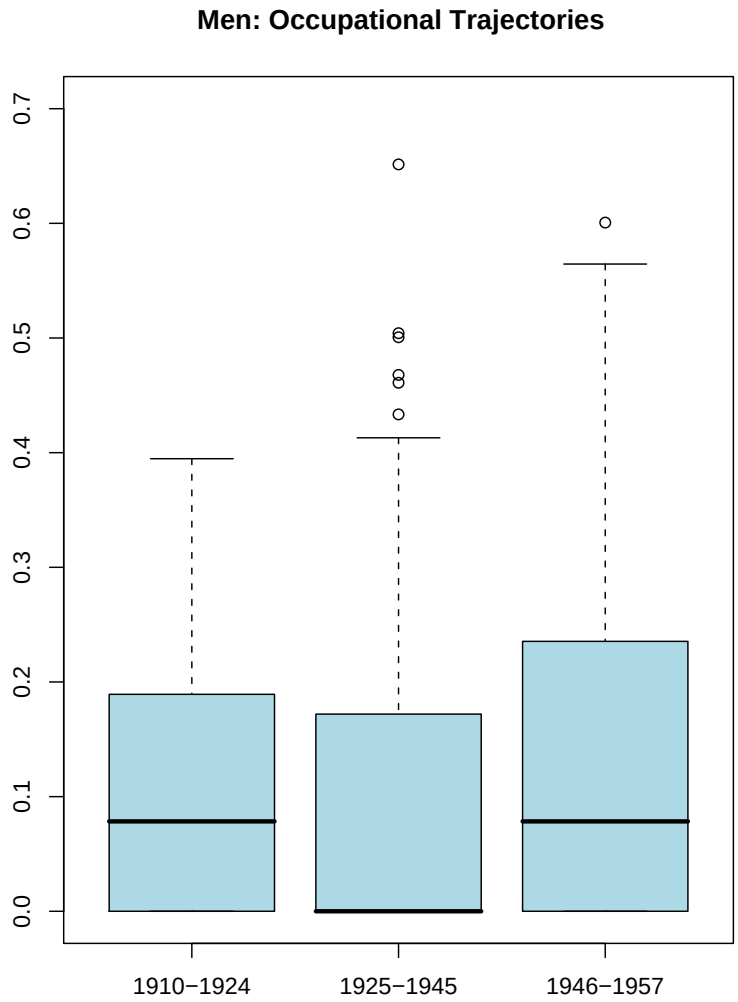
Taux de transition entre états

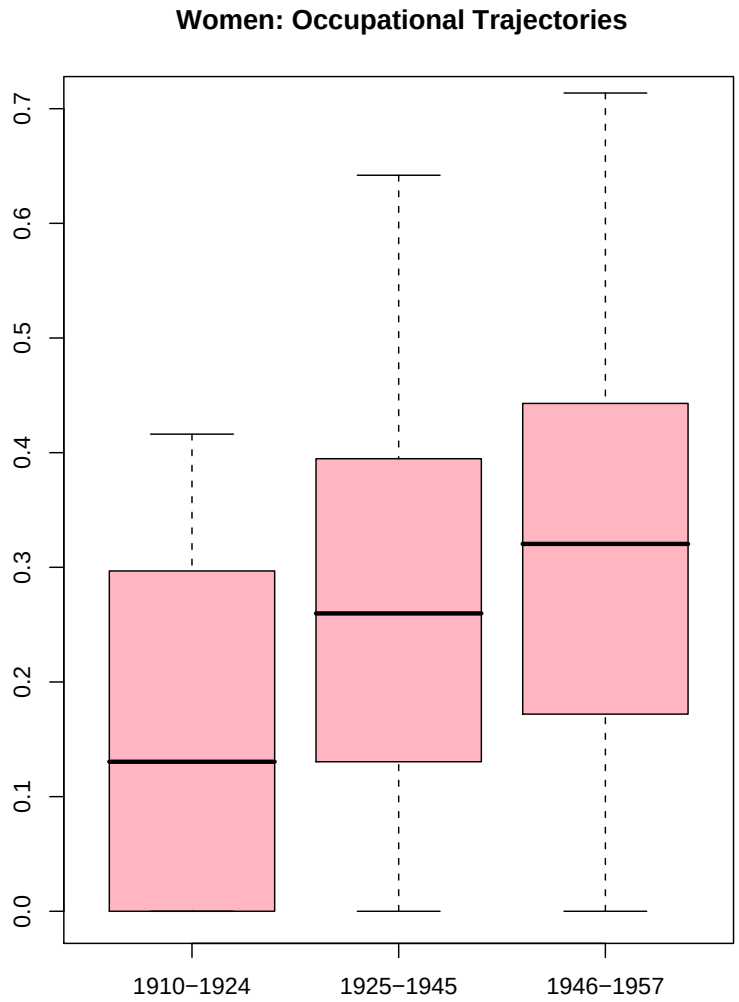
```
        [-> 0] [-> 1] [-> 2]  [-> 3]  [-> 4] [-> 5]  [-> 6] [-> 7]
[0 ->] 0.9692 0.0050 0.0041 0.00114 0.00091 0.0112 0.0e+00 0.0085
[1 ->] 0.0029 0.9706 0.0092 0.00055 0.00068 0.0128 4.6e-05 0.0032
[2 ->] 0.0047 0.0261 0.9393 0.00078 0.00130 0.0177 0.0e+00 0.0102
[3 ->] 0.0400 0.0467 0.0267 0.88000 0.00000 0.0067 0.0e+00 0.0000
[4 ->] 0.1053 0.3158 0.1053 0.00000 0.40351 0.0175 0.0e+00 0.0526
[5 ->] 0.0027 0.0066 0.0320 0.00018 0.00000 0.9564 0.0e+00 0.0020
[6 ->] 0.0000 0.0000 0.0000 0.00000 0.00000 0.0000 1.0e+00 0.0000
[7 ->] 0.0444 0.2363 0.0450 0.00117 0.00234 0.0064 0.0e+00 0.6643

[1] "Missing"    "Full time"  "Part time"  "Neg. break" "Pos. break"
[6] "At home"    "Retired"    "Education"
```

Hétérogénéité : Séquence d'entropies transversales



Entropies longitudinales





Calcul de dissimilarités

- Distances entre séquences
  - Différentes métriques (LCP, LCS, OM)
- Dès qu'on dispose des dissimilarités 2 à 2, on peut
  - Trouver la séquence centrale (centro-type)
  - Mesurer la dispersion des séquences
  - Clusteriser un ensemble de séquences
  - MDS diagramme de dispersion de séquences
  - Analyse de l'hétérogénéité d'un ensemble de séquences (ANOH)
  - Analyse des dissimilarités (par arbres d'induction) (Ne ratez pas la présentation de Matthias jeudi !)

Calcul de dissimilarités

- Distances entre séquences
  - Différentes métriques (LCP, LCS, OM)
- Dès qu'on dispose des dissimilarités 2 à 2, on peut
  - Trouver la séquence centrale (centro-type)
  - Mesurer la dispersion des séquences
  - Clusteriser un ensemble de séquences
  - MDS diagramme de dispersion de séquences
  - Analyse de l'hétérogénéité d'un ensemble de séquences (ANOH)
  - Analyse des dissimilarités (par arbres d'induction) (Ne ratez pas la présentation de Matthias jeudi !)

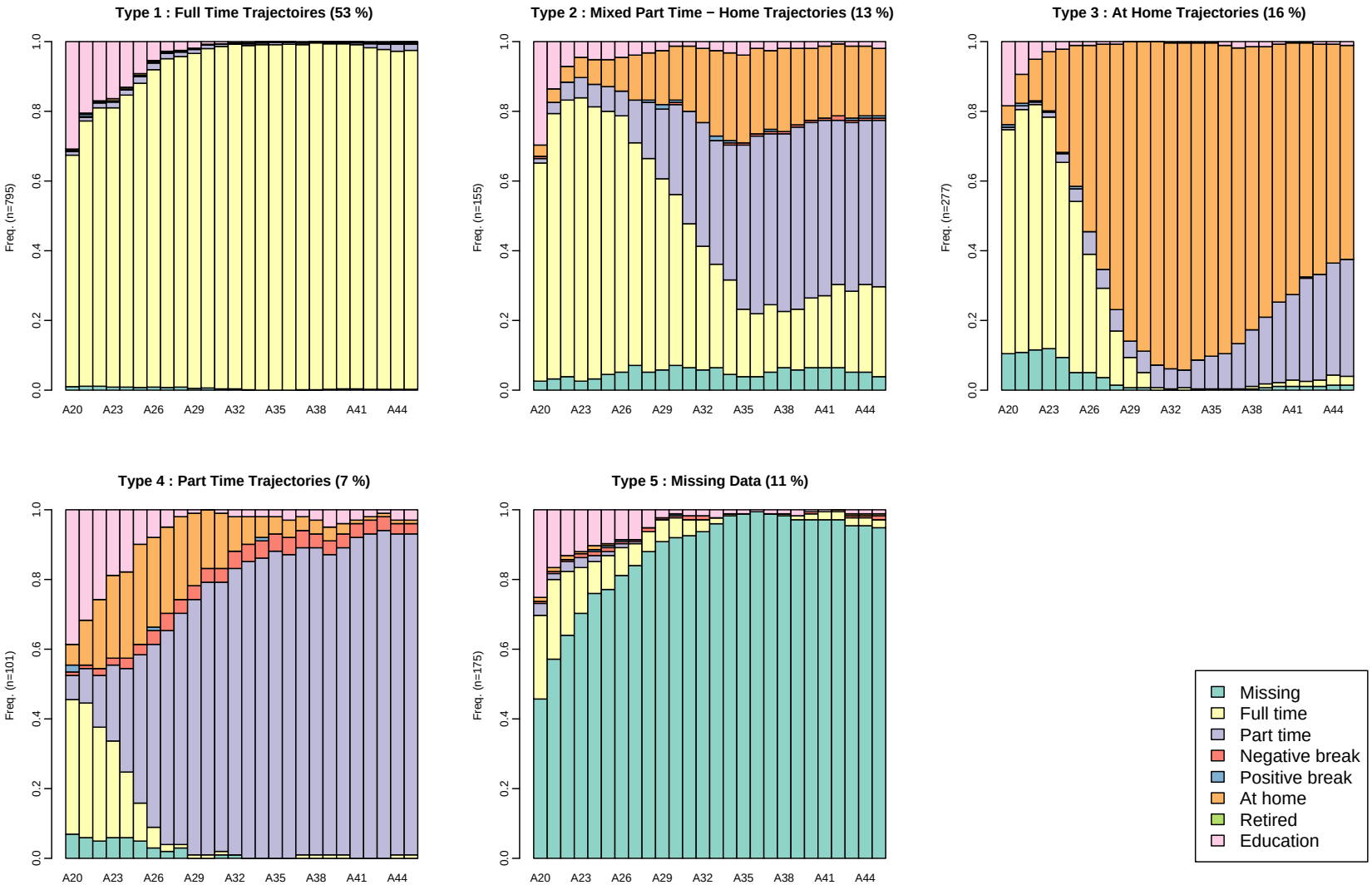
Calcul de dissimilarités

- Distances entre séquences
  - Différentes métriques (LCP, LCS, OM)
- Dès qu'on dispose des dissimilarités 2 à 2, on peut
  - Trouver la séquence centrale (centro-type)
  - Mesurer la dispersion des séquences
  - Clusteriser un ensemble de séquences
  - MDS diagramme de dispersion de séquences
  - Analyse de l'hétérogénéité d'un ensemble de séquences (ANOH)
  - Analyse des dissimilarités (par arbres d'induction) (Ne ratez pas la présentation de Matthias jeudi !)

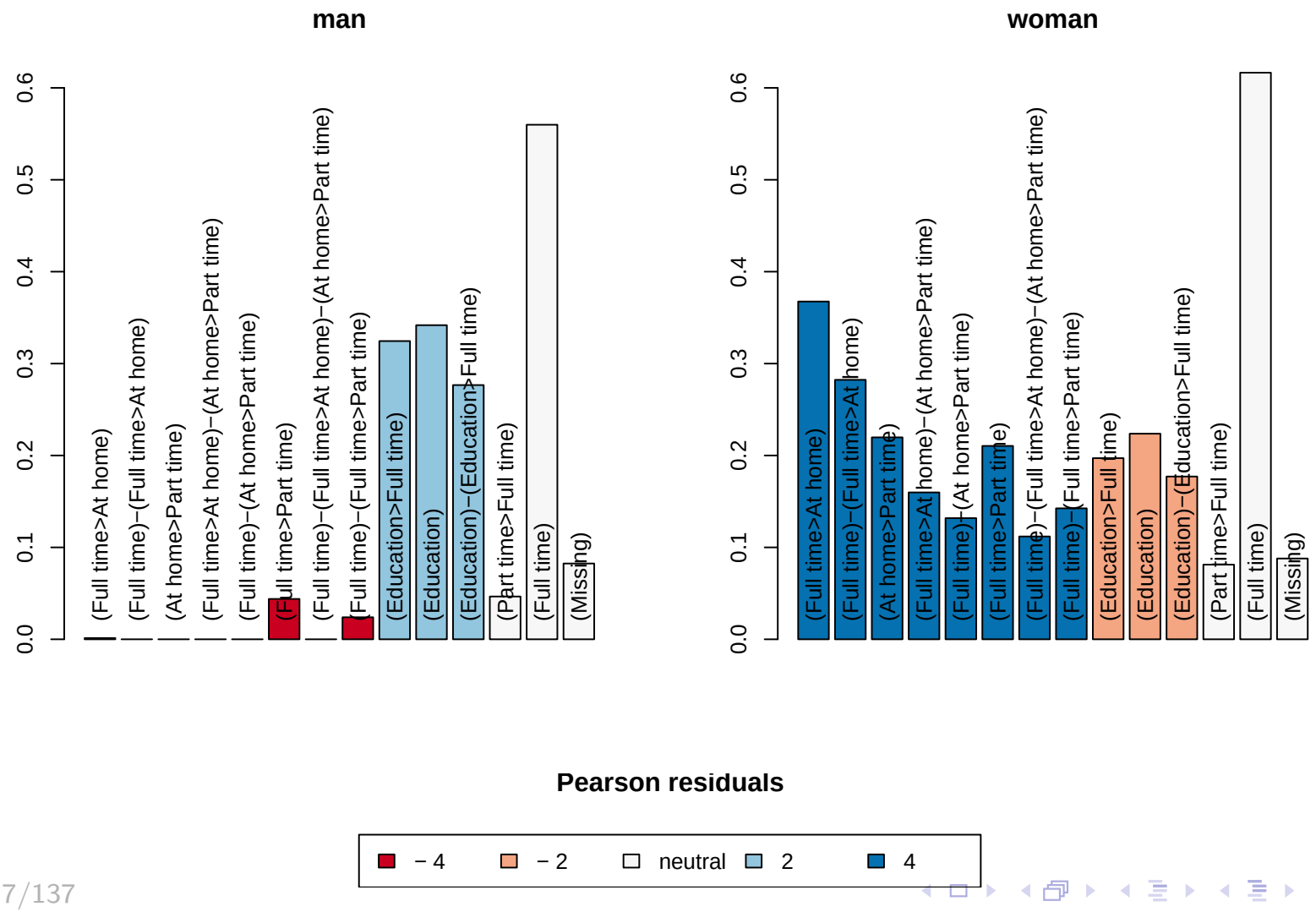
Calcul de dissimilarités

- Distances entre séquences
  - Différentes métriques (LCP, LCS, OM)
- Dès qu'on dispose des dissimilarités 2 à 2, on peut
  - Trouver la séquence centrale (centro-type)
  - Mesurer la dispersion des séquences
  - Clusteriser un ensemble de séquences
  - MDS diagramme de dispersion de séquences
  - Analyse de l'hétérogénéité d'un ensemble de séquences (ANOH)
  - Analyse des dissimilarités (par arbres d'induction)
    (Ne ratez pas la présentation de Matthias jeudi !)

Classification non supervisée, typologie



Séquences d'événements : sous-séquences discriminantes



Ce que vous ne trouverez pas ...

- Analyse des transitions par modèles markoviens ou autres modélisations statistiques
- pour modèles markoviens, voir p. ex. Berchtold and Raftery (2002)

- Analyse de survie
- voir p. ex. Hosmer and Lemeshow (1999), Hothorn et al. (2006)

- la détermination de règles d'association entre sous-séquences
- Peu (pas ?) traité dans la littérature !

Ce que vous ne trouverez pas ...

- Analyse des transitions par modèles markoviens ou autres modélisations statistiques
- pour modèles markoviens, voir p. ex. Berchtold and Raftery (2002)

- Analyse de survie
- voir p. ex. Hosmer and Lemeshow (1999), Hothorn et al. (2006)

- la détermination de règles d'association entre sous-séquences
- Peu (pas ?) traité dans la littérature !

Ce que vous ne trouverez pas ...

- Analyse des transitions par modèles markoviens ou autres modélisations statistiques
- pour modèles markoviens, voir p. ex. Berchtold and Raftery (2002)

- Analyse de survie
- voir p. ex. Hosmer and Lemeshow (1999), Hothorn et al. (2006)

- la détermination de règles d'association entre sous-séquences
- Peu (pas ?) traité dans la littérature !

Plan

Structure de la section

Séquence

Définition :

- Alphabet A : ensemble fini
- Séquence de longueur k : liste ordonnées de k éléments successivement choisis dans A

- Exemples :
  - Texte : A = ens. de lettres, mais peut aussi être constitué de mots
  - Biologie : A = ens. de nucléotides, de protéines, ...
  - Signaux on-off : $A = \{0, 1\}$
  - Comportement d'acheteurs : A = ens. de produits
  - Parcours de vie : A = ens. des états de cohabitation, de taux d'activité, ...

Séquences : notations

- Séquence x de longueur k
  - $x = (x_1, x_2, \ldots, x_k)$
  - Si pas d'ambiguïté : $x = x_1 x_2 \cdots x_k$
  - séparateur nécessaire si A inclut un symbole composé (ex : C célibataire, M marié, ME marié avec enfant C-C-M-M-ME-ME-ME)

Types de séquences

Nature des séquences

Dépend de

- information portée par la position j dans la séquence
  - dimension temporelle ?
- la nature des éléments de l'alphabet
  - objets ou changements
  - états, transitions ou événements

	Dimension temporelle	
Alphabet	Non	Oui
Objets/Etats	séquence d'objets	séquence d'états
Transitions/Evénements	(séquence de changements)	séquence d'événements

Types de séquences

Nature des séquences

Dépend de

- information portée par la position j dans la séquence
  - dimension temporelle ?
- la nature des éléments de l'alphabet
  - objets ou changements
  - états, transitions ou événements

	Dimension temporelle	
Alphabet	Non	Oui
Objets/Etats	séquence d'objets	séquence d'états
Transitions/Evénements	(séquence de changements)	séquence d'événements

Types de séquences

Nature des séquences

Dépend de

- information portée par la position j dans la séquence
  - dimension temporelle ?
- la nature des éléments de l'alphabet
  - objets ou changements
  - états, transitions ou événements

	Dimension temporelle	
Alphabet	Non	Oui
Objets/Etats	séquence d'objets	séquence d'états
Transitions/Evénements	(séquence de changements)	séquence d'événements

Types de séquences

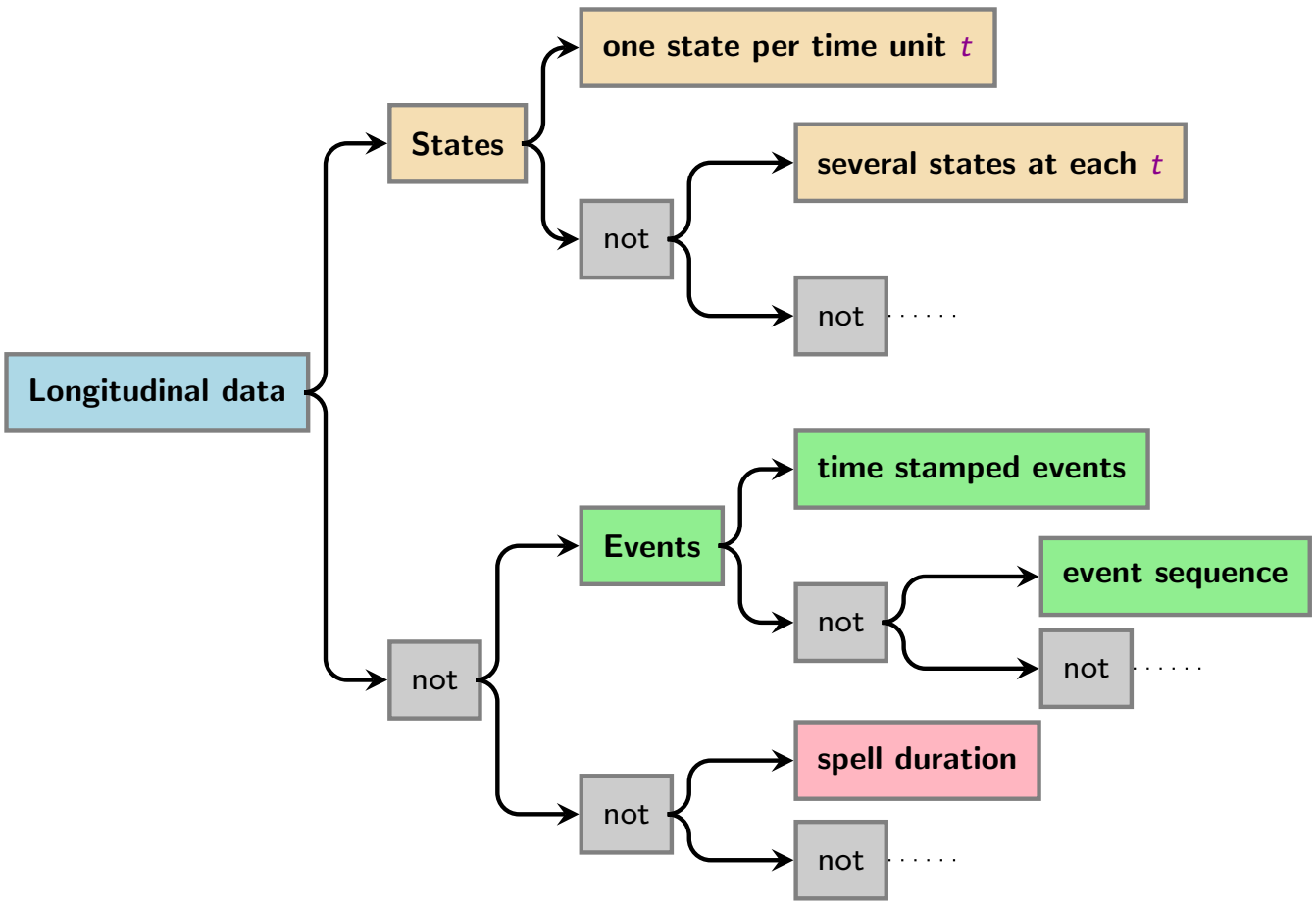
Nature des séquences

Dépend de

- information portée par la position j dans la séquence
  - dimension temporelle ?
- la nature des éléments de l'alphabet
  - objets ou changements
  - états, transitions ou événements

	Dimension temporelle	
Alphabet	Non	Oui
Objets/Etats	séquence d'objets	séquence d'états
Transitions/Evénements	(séquence de changements)	séquence d'événements

Ontologie de données chronologiques
(arbre aristotélicien)



Structure de la section

Vues alternatives de séquences chronologiques

Table: Time stamped events, record for Sandra

ending secondary school in 1970	first job in 1971	marriage in 1973

Table: State sequence view, Sandra

year	1969	1970	1971	1972	1973
marital status	single	single	single	single	married
education level	primary	secondary	secondary	secondary	secondary
job	no	no	first	first	first

Vues alternatives de séquences chronologiques

Table: Time stamped events, record for Sandra

ending secondary school in 1970	first job in 1971	marriage in 1973

Table: State sequence view, Sandra

year	1969	1970	1971	1972	1973
marital status	single	single	single	single	married
education level	primary	secondary	secondary	secondary	secondary
job	no	no	first	first	first

Transforming time stamped events into state sequences

Example : the "BioFam" data

- Data from the retrospective survey conducted in 2002 by the Swiss Household Panel (SHP)
- (with support of Federal Statistical Office, Swiss National Fund for Scientific Research, University of Neuchatel.)
- Retrospective survey : 5560 individuals
- Retained familial life events : Leaving Home, First childbirth, First marriage and First divorce.
- Age 15 to 45 $\longrightarrow$ 2601 remaining individuals, born between 1909 et 1957.

Transforming time stamped events into state sequences

Example : the "BioFam" data

- Data from the retrospective survey conducted in 2002 by the Swiss Household Panel (SHP)
- (with support of Federal Statistical Office, Swiss National Fund for Scientific Research, University of Neuchatel.)
- Retrospective survey : 5560 individuals
- Retained familial life events : Leaving Home, First childbirth, First marriage and First divorce.
- Age 15 to 45 $\longrightarrow$ 2601 remaining individuals, born between 1909 et 1957.

Transforming time stamped events into state sequences

Example : the "BioFam" data

- Data from the retrospective survey conducted in 2002 by the Swiss Household Panel (SHP)
- (with support of Federal Statistical Office, Swiss National Fund for Scientific Research, University of Neuchatel.)
- Retrospective survey : 5560 individuals
- Retained familial life events : Leaving Home, First childbirth, First marriage and First divorce.
- Age 15 to 45 $\longrightarrow$ 2601 remaining individuals, born between 1909 et 1957.

Transforming time stamped events into state sequences

Example : the "BioFam" data

- Data from the retrospective survey conducted in 2002 by the Swiss Household Panel (SHP)
- (with support of Federal Statistical Office, Swiss National Fund for Scientific Research, University of Neuchatel.)
- Retrospective survey : 5560 individuals
- Retained familial life events : Leaving Home, First childbirth, First marriage and First divorce.
- Age 15 to 45 $\longrightarrow$ 2601 remaining individuals, born between 1909 et 1957.

Deriving the states

Associate one state to each combination of events :

	LHome	marriage	childbirth	divorce
0	no	no	no	no
1	yes	no	no	no
2	no	yes	yes/no	no
3	yes	yes	no	no
4	no	no	yes	no
5	yes	no	yes	no
6	yes	yes	yes	no
7	yes/no	yes	yes/no	yes

From events to states

Example of transformation :

- events :

individual	LHome	marriage	childbirth	divorce
1	1989	1990	1992	NA

- states :

individual	...	1988	1989	1990	1991	1992	1993	...
1	...	0	0	1	3	3	6	...

- Peut-on automatiser la transformation
  - d'événements en états ?
  - d'états en événements ?

From events to states

Example of transformation :

- events :

individual	LHome	marriage	childbirth	divorce
1	1989	1990	1992	NA

- states :

individual	...	1988	1989	1990	1991	1992	1993	...
1	...	0	0	1	3	3	6	...

- Peut-on automatiser la transformation
  - d'événements en états ?
  - d'états en événements ?

Structure de la section

Séquences d'états

Formats supportés par TraMineR

Code	Data type	Several rows for same case	Usage examples
STS	State-sequence	No	Markov modeling, OM
SPS	State-permanence	No	Markov modeling, OM
SSS*	State-start	No	Markov modeling, OM
SRS	Shifted-replicated-sequence	Yes	Mobility tree
DSS	Distinct-state-sequence	No	OM without time reference
SPELL	Spell	Yes	Survival analysis
PPER*	Person-period	Yes	Discrete survival analysis

Formats de séquences d'états : exemples - I

Code | **Example**

STS

Id	18	19	20	21	22	23	24	25	26	27
101	S	S	S	M	M	MC	MC	MC	MC	D
102	S	S	S	MC	MC	MC	MC	MC	MC	MC

SPS

Id	1	2	3	4
101	(S,3)	(M,2)	(MC,4)	(D,1)
102	(S,3)	(MC,7)		

SSS*

Id	1	2	3	4
101	(S,18)	(M,21)	(MC,23)	(D,27)
102	(S,18)	(MC,21)		

SRS

Id	$t-9$	$t-8$	$t-7$	$t-6$	$t-5$	$t-4$	$t-3$	$t-2$	$t-1$	t
101	S	S	S	M	M	MC	MC	MC	MC	D
101	.	S	S	S	M	M	MC	MC	MC	MC
101	.	.	S	S	S	M	M	MC	MC	MC
⋮										
101	.	.	.	.	.	.	.	.	S	S
102	S	S	S	MC	MC	MC	MC	MC	MC	MC
102	.	S	S	S	MC	MC	MC	MC	MC	MC
⋮										

DSS

Id	1	2	3	4
101	S	M	MC	D
102	S	MC		

Formats de séquences d'états : exemples - II

Code	Example				
	Id	*Index*	*From*	*To*	*State*
	101	1	18	20	Single (S)
	101	2	21	22	Married (M)
SPELL	101	3	23	26	Married w Children (MC)
	101	4	27	27	Divorced (D)
	102	1	18	20	Single (S)
	102	2	21	27	Married w Children (MC)
	Id	*Index*	*Age*	*State*	
	101	1	18	Single (S)	
	101	2	19	Single (S)	
	101	3	20	Single (S)	
	101	4	21	Married (M)	
PPER*	⋮	⋮	⋮		
	101	10	27	Divorced (D)	
	102	1	18	Single (S)	
	⋮	⋮	⋮		

Séquences d'événements

Formats supportés par TraMineR

Code	Data type	Several rows for same case	Usage examples
FCE*	Fixed-column-event	No	Survival analysis
HTSE*	Horizontal time-stamped-event	No	Event sequence mining
TSE	Vertical time-stamped-event	Yes	Event sequence mining

Formats de séquences d'événements : exemples

Code: FCE*

Id	*#marr.*	*1st marr.*	*2nd marr.*	...	*#child.*	*1st child*	*2nd child*	...
101	1	21	.	.	2	23	26	.
102	1	21	.	.	1	21	.	.

Code: HTSE*

Id	1	2	3	...
101	(marriage, 21)	(childbirth, 23)	(childbirth, 26)	(divorce, 27)
102	(marriage, 21)	(childbirth, 21)		

Code: TSE

Id	*Time*	*Event*
101	21	Marriage
101	23	Childbirth
101	26	Childbirth
101	27	Divorce
102	21	Marriage
102	21	Childbirth

Plan

Données mvad

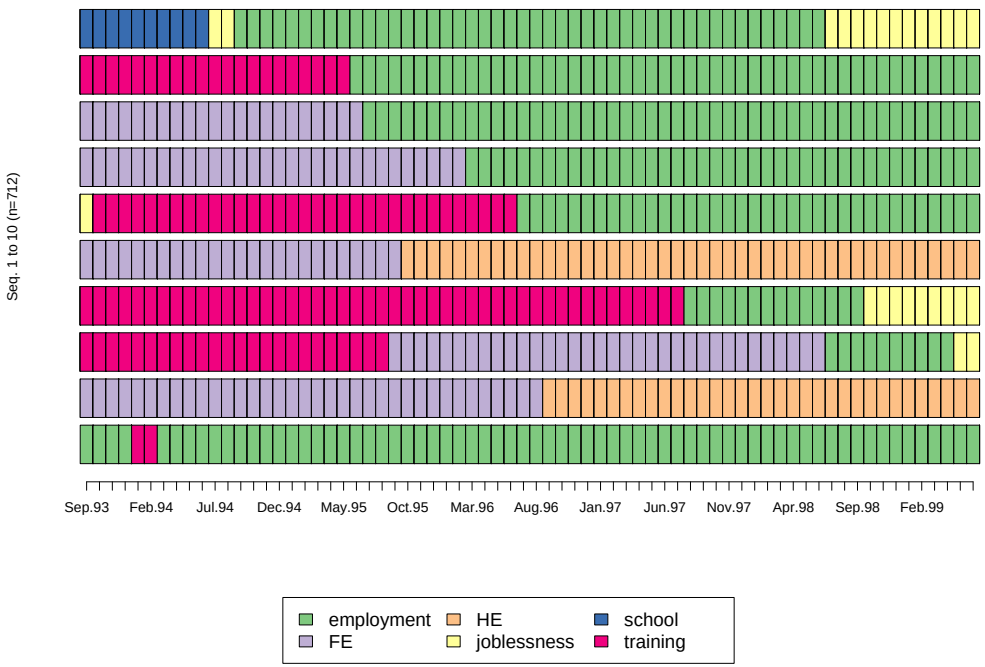
- Nous utilisons dans la suite à titre illustratif les données mvad (McVicar and Anyadike-Danes, 2002)
- Données mensuelles (70 mois) concernant la transition de l'école à l'emploi de jeunes Irlandais.
- Création de l'objet 'séquences d'états"

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

Structure de la section

i-plot : Plot de séquences individuelles (A)

- Le *plot de séquences individuelles* (i-plot) visualise chaque séquence par une barre horizontale. (Scherer, 2001; Brzinsky-Fay et al., 2006)
- i-plot des 10 premières séquences



i-plot : Plot de séquences individuelles (B)

- Le i-plot d'un ensemble de séquences présente plus d'intérêt si elles sont triées et/ou groupées en fonction des valeurs d'une variable.
- Voici comment représenter les séquences en les groupant selon les diplômes obtenus à la fin de l'école obligatoire (gcse5eq) et en les triant par confession

```
R> seqiplot(mvad.seq, tlim = 0, space = 0, group = mvad$gcse5eq,
+         sortv = mvad$catholic, border = NA)
```

i-plot : Plot de séquences individuelles (B)

- Le i-plot d'un ensemble de séquences présente plus d'intérêt si elles sont triées et/ou groupées en fonction des valeurs d'une variable.
- Voici comment représenter les séquences en les groupant selon les diplômes obtenus à la fin de l'école obligatoire (gcse5eq) et en les triant par confession

```
R> seqiplot(mvad.seq, tlim = 0, space = 0, group = mvad$gcse5eq,
+         sortv = mvad$catholic, border = NA)
```

i-plots par groupe

Fréquences des séquences

- Chercher les séquences les plus fréquentes
- `seqtab()` donne les séquences par ordre décroissant (ici les 10 plus fréquentes)

```
R> seqtab(mvad.seq, tlim = 10)
```

	Freq	Percent
(EM,70)	50	7.02
(TR,22)-(EM,48)	18	2.53
(FE,22)-(EM,48)	17	2.39
(SC,24)-(HE,46)	16	2.25
(SC,25)-(HE,45)	13	1.83
(FE,25)-(HE,45)	8	1.12
(FE,34)-(EM,36)	7	0.98
(FE,46)-(EM,24)	7	0.98
(FE,10)-(EM,60)	6	0.84
(FE,24)-(HE,46)	6	0.84

Fréquences des séquences

- Chercher les séquences les plus fréquentes
- `seqtab()` donne les séquences par ordre décroissant (ici les 10 plus fréquentes)

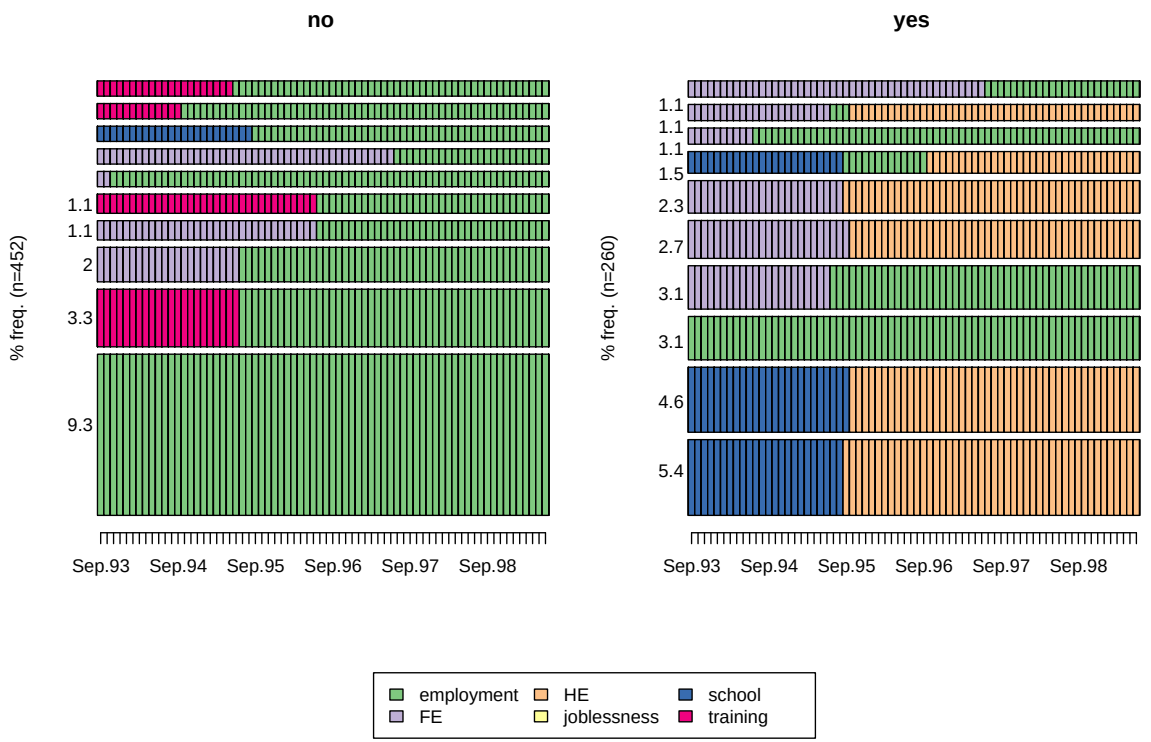
```
R> seqtab(mvad.seq, tlim = 10)
```

	Freq	Percent
(EM,70)	50	7.02
(TR,22)-(EM,48)	18	2.53
(FE,22)-(EM,48)	17	2.39
(SC,24)-(HE,46)	16	2.25
(SC,25)-(HE,45)	13	1.83
(FE,25)-(HE,45)	8	1.12
(FE,34)-(EM,36)	7	0.98
(FE,46)-(EM,24)	7	0.98
(FE,10)-(EM,60)	6	0.84
(FE,24)-(HE,46)	6	0.84

f-plot : séquences selon fréquences

- La fonction `seqfplot()` visualise les séquences les plus fréquentes (ici selon `gcse5eq`).

```
R> seqfplot(mvad.seq, group = mvad$gcse5eq, pbarw = TRUE)
```



Distributions transversales des états

- Distributions des états selon la position.
- La fonction `seqstatd()` donne ces distributions pour chaque position (ici pour 8 premières positions).

```
R> seqstatd(mvad.seq[, 1:8])

$Frequencies
   Sep.93 Oct.93 Nov.93 Dec.93 Jan.94 Feb.94 Mar.94 Apr.94
EM   0.12   0.12   0.13   0.14   0.14   0.14   0.15   0.16
FE   0.39   0.39   0.38   0.38   0.37   0.36   0.36   0.35
HE   0.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00
JL   0.02   0.02   0.02   0.02   0.03   0.04   0.03   0.04
SC   0.25   0.25   0.24   0.24   0.24   0.24   0.24   0.24
TR   0.22   0.22   0.22   0.22   0.22   0.22   0.22   0.21

$ValidStates
Sep.93 Oct.93 Nov.93 Dec.93 Jan.94 Feb.94 Mar.94 Apr.94
   712    712    712    712    712    712    712    712

$Entropy
Sep.93 Oct.93 Nov.93 Dec.93 Jan.94 Feb.94 Mar.94 Apr.94
  0.77   0.77   0.78   0.78   0.79   0.80   0.80   0.81
```

Distributions transversales des états

- Distributions des états selon la position.
- La fonction `seqstatd()` donne ces distributions pour chaque position (ici pour 8 premières positions).

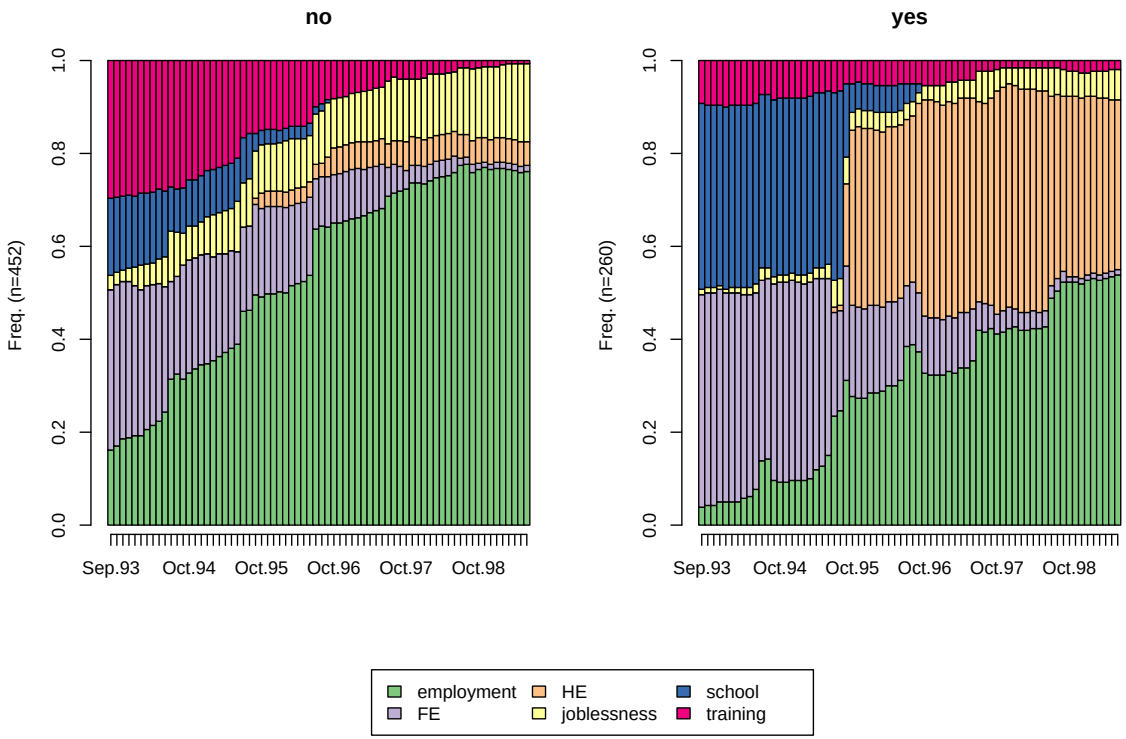
```
R> seqstatd(mvad.seq[, 1:8])
```

```
$Frequencies
   Sep.93 Oct.93 Nov.93 Dec.93 Jan.94 Feb.94 Mar.94 Apr.94
EM   0.12   0.12   0.13   0.14   0.14   0.14   0.15   0.16
FE   0.39   0.39   0.38   0.38   0.37   0.36   0.36   0.35
HE   0.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00
JL   0.02   0.02   0.02   0.02   0.03   0.04   0.03   0.04
SC   0.25   0.25   0.24   0.24   0.24   0.24   0.24   0.24
TR   0.22   0.22   0.22   0.22   0.22   0.22   0.22   0.21

$ValidStates
Sep.93 Oct.93 Nov.93 Dec.93 Jan.94 Feb.94 Mar.94 Apr.94
   712    712    712    712    712    712    712    712

$Entropy
Sep.93 Oct.93 Nov.93 Dec.93 Jan.94 Feb.94 Mar.94 Apr.94
  0.77   0.77   0.78   0.78   0.79   0.80   0.80   0.81
```

d-plot : Séquences des distributions transversales

- La fonction `seqdplot()` visualise les distributions transversales (ici selon `gcse5eq`).

```
R> seqdplot(mvad.seq, group = mvad$gcse5eq)
```

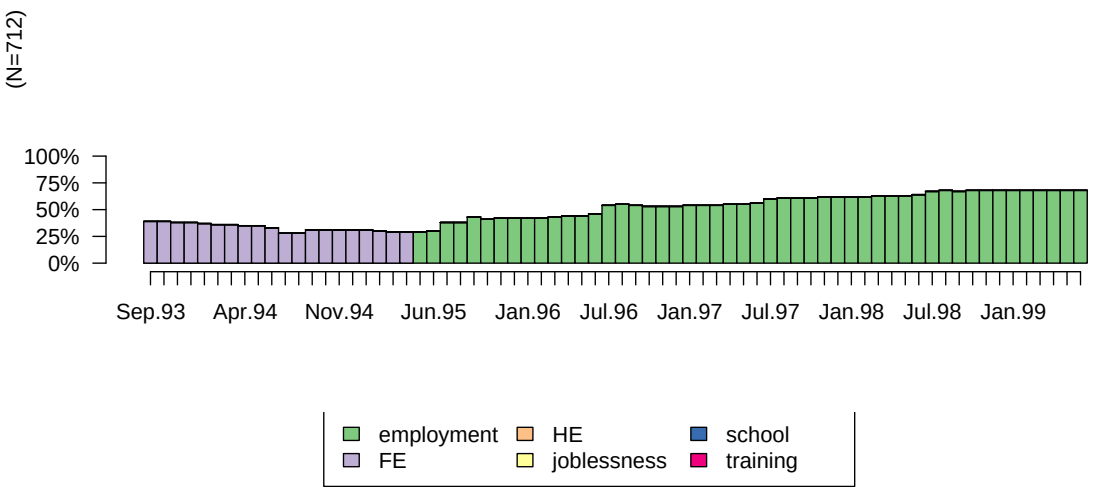


Structure de la section

Séquence des états les plus fréquents

- Séquence des états modaux(avec leurs fréquences)

```
 [1] "FE" "FE" "FE" "FE" "FE" "FE" "FE" "FE" "FE" "FE" "FE" "FE" "FE" "FE"
[15] "FE" "FE" "FE" "FE" "FE" "FE" "EM" "EM"
```



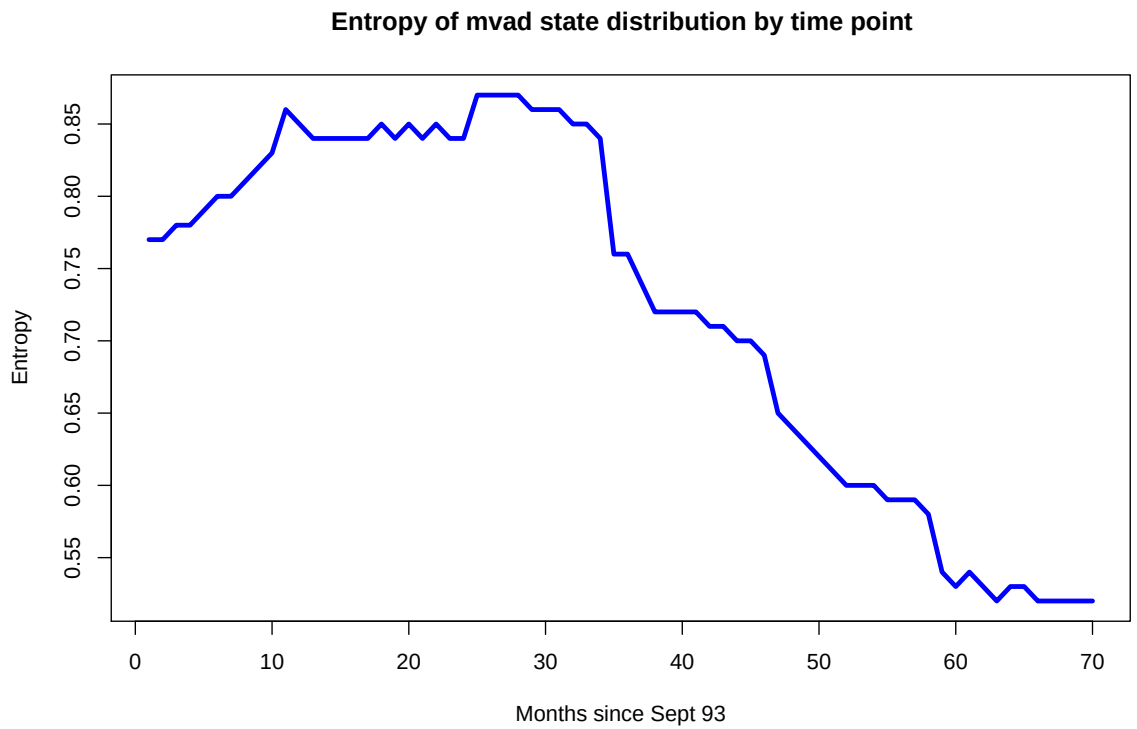
Entropie transversale

- Entropie de chaque distribution transversale $(p_1, \ldots, p_a)$, avec $a = |A|$ taille de l'alphabet
- Entropie de Shannon

$$h(p_1, \ldots, p_s) = -\sum_{i=1}^{s} p_i \log_2(p_i)$$

Graphique de la série des entropies

```
R> sd <- seqstatd(mvad.seq)
R> plot(sd$Entropy, main = "Entropy of mvad state distribution by time point",
+       xlab = "Months since Sept 93", ylab = "Entropy", type = "l",
+       lwd = 3.5, col = "blue")
```



Structure de la section

Durée moyenne passée dans chaque état (A)

- Durée passée dans chacun des états pour chaque séquence individuelle

```
R> mvad.statd <- seqistatd(mvad.seq)
R> mvad.statd[1:5, ]
```

```
     EM FE HE JL SC TR
[1,] 68  0  0  0  0  2
[2,]  0 36 34  0  0  0
[3,] 10 34  0  2  0 24
[4,] 14  0  0  9  0 47
[5,]  0 25 45  0  0  0
```

- Moyenne par colonne

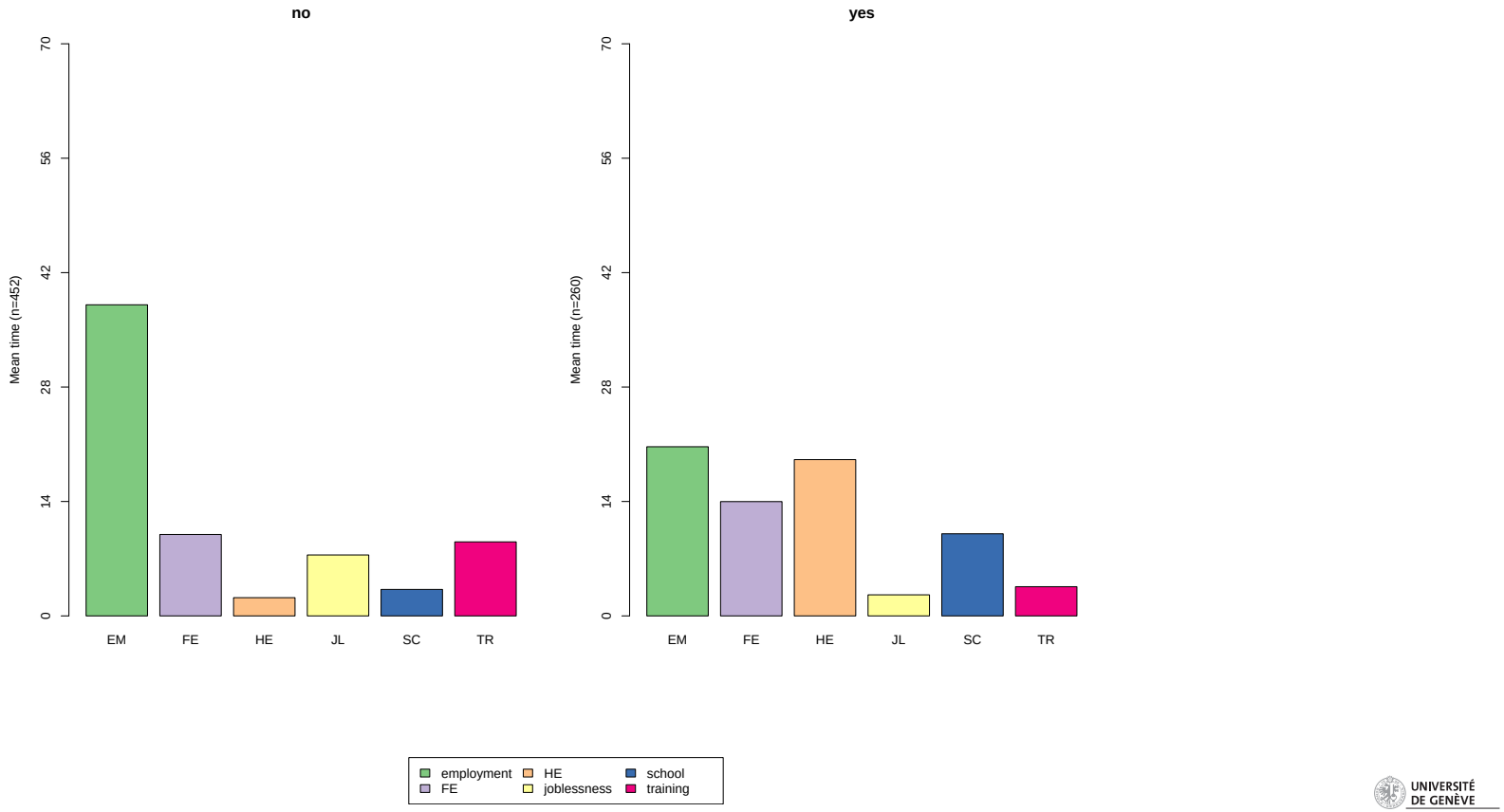
```
R> mt <- apply(mvad.statd, 2, mean)
R> mt
```

```
       EM        FE        HE        JL        SC        TR
31.721910 11.426966  8.398876  5.674157  5.723315  7.054775
```

Plot des durées moyennes

- Plot des durées moyennes selon gcse5eq

```
R> seqmtplot(mvad.seq, group = mvad$gcse5eq)
```



Structure de la section

Taux de transition

- Taux de transition : estimation des prob. $p(x_{it} \mid x_{j(t-1)})$

```
R> round(seqtrate(mvad.seq), digits = 4)
```

	[-> EM]	[-> FE]	[-> HE]	[-> JL]	[-> SC]	[-> TR]
[EM ->]	0.9864	0.0020	0.0025	0.0065	0.0004	0.0022
[FE ->]	0.0279	0.9514	0.0066	0.0090	0.0010	0.0041
[HE ->]	0.0102	0.0002	0.9872	0.0019	0.0000	0.0005
[JL ->]	0.0418	0.0084	0.0023	0.9387	0.0005	0.0084
[SC ->]	0.0142	0.0081	0.0182	0.0056	0.9509	0.0029
[TR ->]	0.0383	0.0036	0.0000	0.0136	0.0004	0.9442

Entropie longitudinale

- égale à 0 si une séquence ne comporte qu'un seul état (si la personne reste dans le même état durant toute la durée d'observation, par exemple `A-A-A-A-A-A-A-A`)
- maximum si la séquence comprend un même nombre de chacun des états de l'alphabet (la personne a passé une durée équivalente dans tous les états possibles, par exemple `A-A-B-B-C-C-D-D`)
- Par défaut, TraMineR normalise l'entropie par l'entropie de l'alphabet

$$h_{std}(p_1, \ldots, p_a) = \frac{-\sum_{i=1}^{a} p_i \log_2(p_i)}{h(A)}$$

avec p_i proportion de positions dans l'état i.

Entropie longitudinale

- égale à 0 si une séquence ne comporte qu'un seul état (si la personne reste dans le même état durant toute la durée d'observation, par exemple `A-A-A-A-A-A-A-A`)
- maximum si la séquence comprend un même nombre de chacun des états de l'alphabet (la personne a passé une durée équivalente dans tous les états possibles, par exemple `A-A-B-B-C-C-D-D`)
- Par défaut, TraMineR normalise l'entropie par l'entropie de l'alphabet

$$h_{std}(p_1, \ldots, p_a) = \frac{-\sum_{i=1}^{a} p_i \log_2(p_i)}{h(A)}$$

avec p_i proportion de positions dans l'état i.

Entropie d'une séquence (B)

- On calcule l'entropie pour les séquences du fichier *mvad*

```
R> mvad.ient <- seqient(mvad.seq)

R> mvad.ient[1:6, ]

      [1]       [2]       [3]       [4]       [5]       [6]
0.07240966 0.38662498 0.61243051 0.47611545 0.36375226 0.42259527
```

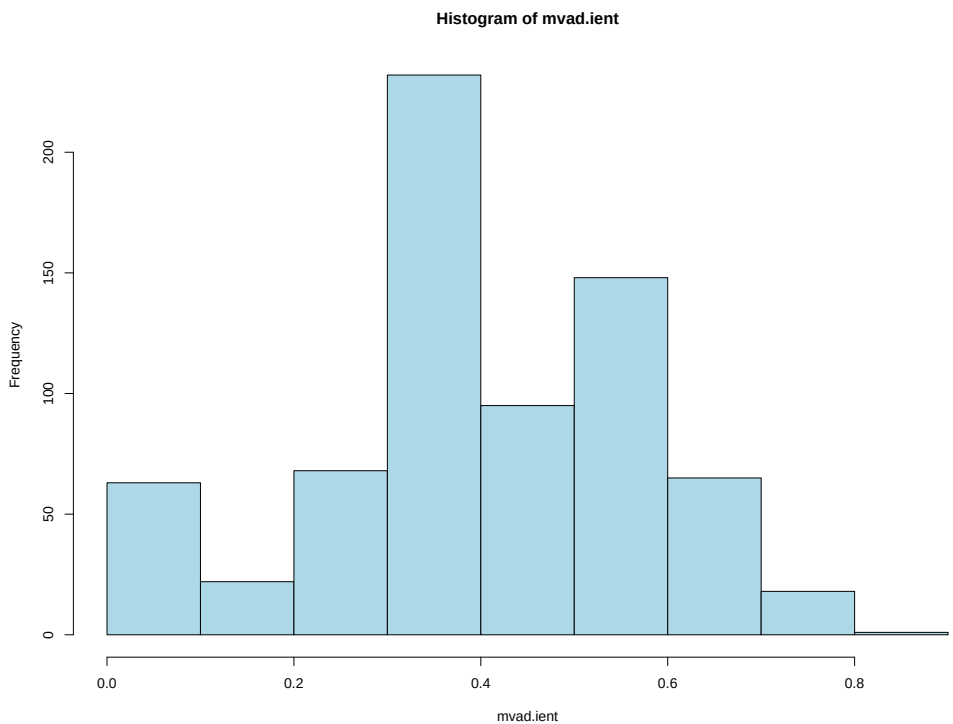
- Les valeurs sont comprises entre 0 et 1 (par défaut l'entropie est normalisée)

```
R> min(mvad.ient)

[1] 0

R> max(mvad.ient)

[1] 0.854786
```

Entropie d'une séquence - Histogramme

- Distribution des entropies pour les séquences de *mvad*

```
R> hist(mvad.ient, col = "LightBlue")
```



Turbulence

- Entropie ne tient pas compte de l'ordre
- Turbulence : mesure alternative proposée par Elzinga and Liefbroer (2007) tenant compte indirectement de l'ordre.
- Elle est basée sur
  - le nombre $\phi(x)$ de sous-séquences distinctes contenues dans la suite des états distincts composant la séquence
    `x=S-U-M-C` (16 sous-séquences) plus turbulente que
    `y=S-U-S-C` (15 sous-séquences)
  - la variance des durées passées dans chacun des états distincts t_i
    S/10-U/2-M/132 trajectoire moins turbulente que
    S/48-U/48-M/48

Turbulence

- Entropie ne tient pas compte de l'ordre
- Turbulence : mesure alternative proposée par Elzinga and Liefbroer (2007) tenant compte indirectement de l'ordre.
- Elle est basée sur
  - le nombre $\phi(x)$ de sous-séquences distinctes contenues dans la suite des états distincts composant la séquence `x=S-U-M-C` (16 sous-séquences) plus turbulente que `y=S-U-S-C` (15 sous-séquences)
  - la variance des durées passées dans chacun des états distincts t_i S/10-U/2-M/132 trajectoire moins turbulente que S/48-U/48-M/48

Turbulence

- Entropie ne tient pas compte de l'ordre
- Turbulence : mesure alternative proposée par Elzinga and Liefbroer (2007) tenant compte indirectement de l'ordre.
- Elle est basée sur
  - le nombre $\phi(x)$ de sous-séquences distinctes contenues dans la suite des états distincts composant la séquence `x=S-U-M-C` (16 sous-séquences) plus turbulente que `y=S-U-S-C` (15 sous-séquences)
  - la variance des durées passées dans chacun des états distincts t_i S/10-U/2-M/132 trajectoire moins turbulente que S/48-U/48-M/48

Turbulence

- Entropie ne tient pas compte de l'ordre
- Turbulence : mesure alternative proposée par Elzinga and Liefbroer (2007) tenant compte indirectement de l'ordre.
- Elle est basée sur
  - le nombre $\phi(x)$ de sous-séquences distinctes contenues dans la suite des états distincts composant la séquence `x=S-U-M-C` (16 sous-séquences) plus turbulente que `y=S-U-S-C` (15 sous-séquences)
  - la variance des durées passées dans chacun des états distincts t_i S/10-U/2-M/132 trajectoire moins turbulente que S/48-U/48-M/48

Turbulence (suite)

- Il nous faut la séquence des états distincts (DSS)
- Dans le format SPS, une séquence est représentée sous la forme d'une suite d'états distincts et des durées qui y sont associées

```
R> print(mvad.seq[1, ], format = "SPS")

    Sequence
[1] (EM,4)-(TR,2)-(EM,64)
```

- La DSS pour la séquence précédente est

```
R> seqdss(mvad.seq[1, ])

    Sequence
[1] EM-TR-EM
```

- Le nombre de sous-séquences contenues dans la DSS précédente est

```
R> seqsubsn(mvad.seq[1, ], DSS = TRUE)

[1] 7
```

Turbulence (suite)

- Il nous faut la séquence des états distincts (DSS)
- Dans le format SPS, une séquence est représentée sous la forme d'une suite d'états distincts et des durées qui y sont associées

```
R> print(mvad.seq[1, ], format = "SPS")
```

```
     Sequence
[1] (EM,4)-(TR,2)-(EM,64)
```

- La DSS pour la séquence précédente est

```
R> seqdss(mvad.seq[1, ])
```

```
     Sequence
[1] EM-TR-EM
```

- Le nombre de sous-séquences contenues dans la DSS précédente est

```
R> seqsubsn(mvad.seq[1, ], DSS = TRUE)
```

```
[1] 7
```

Turbulence (suite)

- Il nous faut la séquence des états distincts (DSS)
- Dans le format SPS, une séquence est représentée sous la forme d'une suite d'états distincts et des durées qui y sont associées

```
R> print(mvad.seq[1, ], format = "SPS")
```

```
     Sequence
[1] (EM,4)-(TR,2)-(EM,64)
```

- La DSS pour la séquence précédente est

```
R> seqdss(mvad.seq[1, ])
```

```
     Sequence
[1] EM-TR-EM
```

- Le nombre de sous-séquences contenues dans la DSS précédente est

```
R> seqsubsn(mvad.seq[1, ], DSS = TRUE)
```

```
[1] 7
```

Turbulence : formule

- Formule pour une séquence x

$$T(x) = \log_2 \left(\phi(x) \frac{s^2_{t,max}(x) + 1}{s^2_t(x) + 1} \right)$$

- où s^2_t est la variance des durées passées dans chacun des états distincts et $s^2_{t,max}$ est la valeur maximum que peut atteindre cette variance compte tenu de la durée totale de la séquence.
- Ce maximum est

$$s^2_{t,max} = (n-1)(1-\bar{t})$$

- où $\bar{t}$ est la moyenne des durées consécutives passées dans les états distincts.
- $\bar{t}$ = durée de la séquence/nombre de ses états distincts.

Turbulence : formule

- Formule pour une séquence x

$$T(x) = \log_2 \left(\phi(x) \frac{s^2_{t,max}(x) + 1}{s^2_t(x) + 1} \right)$$

- où s^2_t est la variance des durées passées dans chacun des états distincts et $s^2_{t,max}$ est la valeur maximum que peut atteindre cette variance compte tenu de la durée totale de la séquence.
- Ce maximum est

$$s^2_{t,max} = (n-1)(1-\bar{t})$$

- où $\bar{t}$ est la moyenne des durées consécutives passées dans les états distincts.
- $\bar{t}$ = durée de la séquence/nombre de ses états distincts.

Calcul de la Turbulence

- La turbulence est calculée avec la fonction `seqST()`

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

- Turbulence des 6 premières séquences

```
R> mvad.turb[1:6]
```

```
[1]  3.076599 11.176173  6.411073  4.807756  5.517962  4.987055
```

- La mesure n'est pas normalisée

```
R> min(mvad.turb)
```

```
[1] 1
```

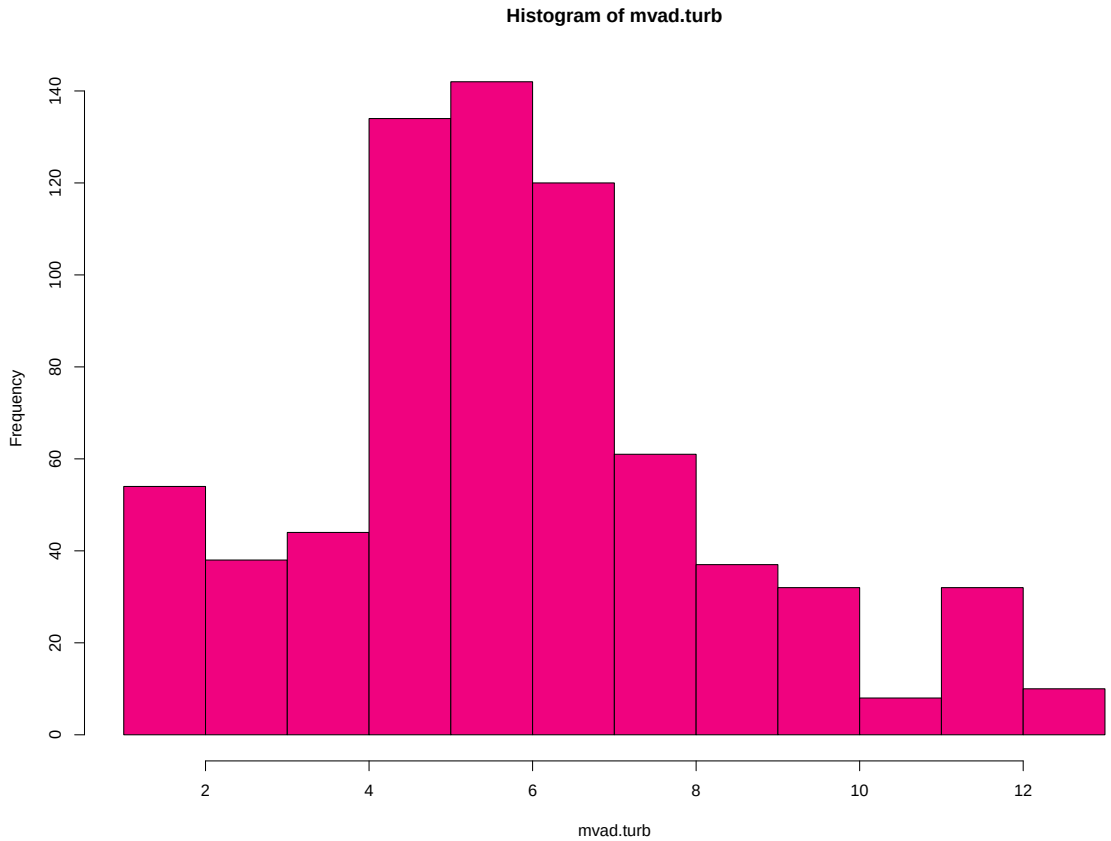
```
R> max(mvad.turb)
```

```
[1] 12.95858
```

Turbulence - Histogramme

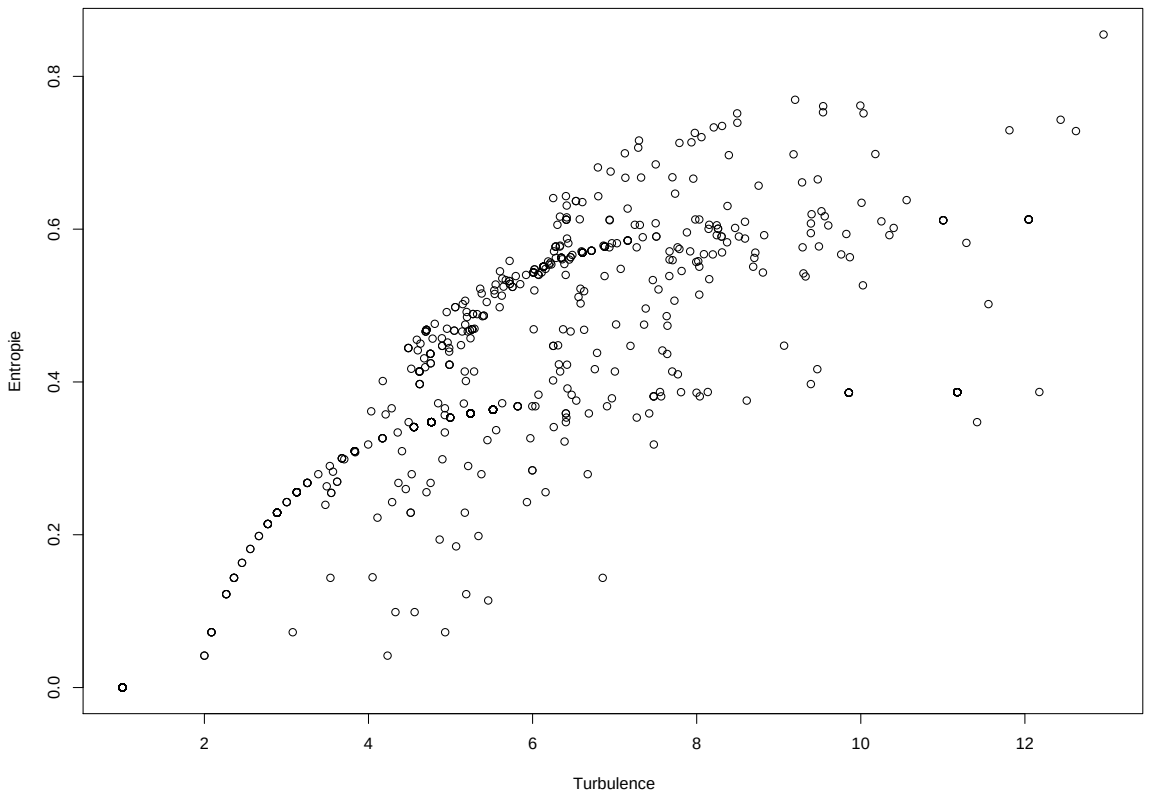
- Distribution des turbulences pour les séquences de *mvad*

```
R> hist(mvad.turb, col = attr(mvad.seq, "cpal")[6])
```



Comparaison Turbulence-Entropie

```
R> plot(mvad.turb, mvad.ient, xlab = "Turbulence", ylab = "Entropie")
```



Structure de la section

Mesures disponibles dans TraMineR

- Trois types de mesures de proximité sont disponibles :
  1. Longest Common Prefix (LCP)
  2. Longest Common Subsequence (LCS)
  3. Optimal Matching (OM)

LCP : Longest Common Prefix

- Le LCP donne le plus long préfixe commun entre deux séquences.
- La mesure LLCP donne la longueur de ce préfixe :

```
R> mvad.seq[2, ]

    Sequence
[1] FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-

R> mvad.seq[5, ]

    Sequence
[1] FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-FE-

R> seqLLCP(mvad.seq[2, ], mvad.seq[5, ])

[1] 25
```

- La taille du plus long préfixe commun entre ces deux séquences est de **25**.

Distance LCP

- Pour l'instant nous avons une mesure de la proximité.
- Pour obtenir une mesure de la distance, Elzinga (2008) propose
  - une distance : $d_P(x,y) = |x| + |y| - 2\mathcal{A}_p(x,y)$
  - une distance normalisée : $D_P(x,y) = 1 - \frac{\mathcal{A}_p(x,y)}{\sqrt{|x| \cdot |y|}}$
- Où $\mathcal{A}_p(x,y)$ est la taille du LCP entre x et y, et $|\ x\ |$ et $|\ y\ |$ sont les longueurs de x et y.

Distance LCP

- Pour l'instant nous avons une mesure de la proximité.
- Pour obtenir une mesure de la distance, Elzinga (2008) propose
  - une distance : $d_P(x,y) = |x| + |y| - 2\mathcal{A}_p(x,y)$
  - une distance normalisée : $D_P(x,y) = 1 - \frac{\mathcal{A}_p(x,y)}{\sqrt{|x| \cdot |y|}}$
- Où $\mathcal{A}_p(x,y)$ est la taille du LCP entre x et y, et $\mid x \mid$ et $\mid y \mid$ sont les longueurs de x et y.

Distance LCP

- Pour l'instant nous avons une mesure de la proximité.
- Pour obtenir une mesure de la distance, Elzinga (2008) propose
  - une distance : $d_P(x, y) = |x| + |y| - 2\mathcal{A}_p(x, y)$
  - une distance normalisée : $D_P(x, y) = 1 - \frac{\mathcal{A}_p(x,y)}{\sqrt{|x| \cdot |y|}}$
- Où $\mathcal{A}_p(x, y)$ est la taille du LCP entre x et y, et $| x |$ et $| y |$ sont les longueurs de x et y.

Distance LCP

- Pour l'instant nous avons une mesure de la proximité.
- Pour obtenir une mesure de la distance, Elzinga (2008) propose
  - une distance : $d_P(x, y) = |x| + |y| - 2\mathcal{A}_p(x, y)$
  - une distance normalisée : $D_P(x, y) = 1 - \frac{\mathcal{A}_p(x,y)}{\sqrt{|x| \cdot |y|}}$
- Où $\mathcal{A}_p(x, y)$ est la taille du LCP entre x et y, et $\mid x \mid$ et $\mid y \mid$ sont les longueurs de x et y.

LCP dans TraMineR

- Dans TraMineR, les matrices de distances s'obtiennent avec la fonction

  `seqdist()`

- On choisit le type de distance avec l'option `method=...`
- Pour obtenir la version normalisée d'une distance, on ajoute l'option `norm=TRUE`

LCP dans TraMineR

- Dans TraMineR, les matrices de distances s'obtiennent avec la fonction

  `seqdist()`
- On choisit le type de distance avec l'option `method=...`
- Pour obtenir la version normalisée d'une distance, on ajoute l'option `norm=TRUE`

LCP dans TraMineR

- Dans TraMineR, les matrices de distances s'obtiennent avec la fonction

  `seqdist()`
- On choisit le type de distance avec l'option `method=...`
- Pour obtenir la version normalisée d'une distance, on ajoute l'option `norm=TRUE`

Exemple sur les 6 premières séquences de McVicar

Distance LCP non-normalisée

```
R> seqdist(mvad.seq[1:6, ], method = "LCP", norm = FALSE)
```

```
     [1] [2] [3] [4] [5] [6]
[1]    0 140 140 140 140 140
[2] 140   0 140 140  90 140
[3] 140 140   0  92 140 140
[4] 140 140  92   0 140 140
[5] 140  90 140 140   0 140
[6] 140 140 140 140 140   0
```

Exemple sur les 6 premières séquences de McVicar

Distance LCP normalisée

```
R> seqdist(mvad.seq[1:6, ], method = "LCP", norm = TRUE)
```

	[1]	[2]	[3]	[4]	[5]	[6]
[1]	0	1.0000000	1.0000000	1.0000000	1.0000000	1
[2]	1	0.0000000	1.0000000	1.0000000	0.6428571	1
[3]	1	1.0000000	0.0000000	0.6571429	1.0000000	1
[4]	1	1.0000000	0.6571429	0.0000000	1.0000000	1
[5]	1	0.6428571	1.0000000	1.0000000	0.0000000	1
[6]	1	1.0000000	1.0000000	1.0000000	1.0000000	0

LCS : Longest Common Subsequences

- Cette mesure calcule la taille des sous-séquences partagées par 2 séquences
- Exemple :
  - x : 1-1-1-2-2-3-3
  - y : 1-1-1-4-4-3-3
- LCS = **5**
- La mesure de dissimilarité associée est :
  $d_{LCS}(x,y) = \mathcal{A}_\ell(x,x) + \mathcal{A}_\ell(y,y) - 2\mathcal{A}_\ell(x,y)$
- ou dans sa forme normalisée : $D_{LCS}(x,y) = \frac{\mathcal{A}_\ell(x,y)}{\sqrt{|x| \cdot |y|}}$

LCS : Longest Common Subsequences

- Cette mesure calcule la taille des sous-séquences partagées par 2 séquences
- Exemple :
  - x : 1-1-1-2-2-3-3
  - y : 1-1-1-4-4-3-3
- LCS = **5**
- La mesure de dissimilarité associée est :
  $d_{LCS}(x,y) = \mathcal{A}_\ell(x,x) + \mathcal{A}_\ell(y,y) - 2\mathcal{A}_\ell(x,y)$
- ou dans sa forme normalisée : $D_{LCS}(x,y) = \frac{\mathcal{A}_\ell(x,y)}{\sqrt{|x| \cdot |y|}}$

LCS : Longest Common Subsequences

- Cette mesure calcule la taille des sous-séquences partagées par 2 séquences
- Exemple :
  - x : 1-1-1-2-2-3-3
  - y : 1-1-1-4-4-3-3
- LCS = **5**
- La mesure de dissimilarité associée est :
  $d_{LCS}(x,y) = \mathcal{A}_\ell(x,x) + \mathcal{A}_\ell(y,y) - 2\mathcal{A}_\ell(x,y)$
- ou dans sa forme normalisée : $D_{LCS}(x,y) = \frac{\mathcal{A}_\ell(x,y)}{\sqrt{|x| \cdot |y|}}$

LCS : Longest Common Subsequences

- Cette mesure calcule la taille des sous-séquences partagées par 2 séquences
- Exemple :
  - x : 1-1-1-2-2-3-3
  - y : 1-1-1-4-4-3-3
- LCS = **5**
- La mesure de dissimilarité associée est :
  $d_{LCS}(x,y) = \mathcal{A}_\ell(x,x) + \mathcal{A}_\ell(y,y) - 2\mathcal{A}_\ell(x,y)$
- ou dans sa forme normalisée : $D_{LCS}(x,y) = \frac{\mathcal{A}_\ell(x,y)}{\sqrt{|x| \cdot |y|}}$

LCS : exemple

```
R> x <- c(1, 1, 1, 2, 2, 3, 3)
R> y <- c(1, 1, 1, 4, 4, 3, 3)
R> seqdist(seqdef(rbind(x, y)), method = "LCS")
```

```
    [1] [2]
[1]   0   4
[2]   4   0
```

```
R> seqdist(seqdef(rbind(x, y)), method = "LCS", norm = TRUE)
```

```
          [1]       [2]
[1] 0.0000000 0.2857143
[2] 0.2857143 0.0000000
```

L'optimal matching (alignement optimal)

- On cherche à calculer la distance entre deux séquences
- Principe de la distance de Levenshtein
- S'inspire de l'alignement pratiqué en biologie (séquences ADN ou de protéines)
- Introduite par Abbott and Forrest (1986) dans les sciences sociales

L'optimal matching (alignement optimal)

- On cherche à calculer la distance entre deux séquences
- Principe de la distance de Levenshtein
- S'inspire de l'alignement pratiqué en biologie (séquences ADN ou de protéines)
- Introduite par Abbott and Forrest (1986) dans les sciences sociales

L'optimal matching (alignement optimal)

- On cherche à calculer la distance entre deux séquences
- Principe de la distance de Levenshtein
- S'inspire de l'alignement pratiqué en biologie (séquences ADN ou de protéines)
- Introduite par Abbott and Forrest (1986) dans les sciences sociales

L'optimal matching (alignement optimal)

- On cherche à calculer la distance entre deux séquences
- Principe de la distance de Levenshtein
- S'inspire de l'alignement pratiqué en biologie (séquences ADN ou de protéines)
- Introduite par Abbott and Forrest (1986) dans les sciences sociales

Optimal matching : principes de base

- On dispose de deux types d'opérations sur les séquences :
  - Insertion ou suppression d'un élément
  - Substitution d'un élément
- Chaque opération a un coût.
- La distance est le coût minimum nécessaire pour transformer une séquence en une autre.

Optimal matching : principes de base

- On dispose de deux types d'opérations sur les séquences :
  - Insertion ou suppression d'un élément
  - Substitution d'un élément
- Chaque opération a un coût.
- La distance est le coût minimum nécessaire pour transformer une séquence en une autre.

Optimal matching : principes de base

- On dispose de deux types d'opérations sur les séquences :
  - Insertion ou suppression d'un élément
  - Substitution d'un élément
- Chaque opération a un coût.
- La distance est le coût minimum nécessaire pour transformer une séquence en une autre.

OM : exemple

Soient deux séquences :

1	0	1	1	2	2	2	3
2	0	1	1	2	2	3	3

Insertion de l'élément « 2 » :

1	0	1	1	2	2	2	3	
2	0	1	1	2	2	2	3	3

Suppression de l'élément « 3 » :

1	0	1	1	2	2	2	3	
2	0	1	1	2	2	2	3	

Les deux séquences sont maintenant identiques.

OM : exemple

Soient deux séquences :

1	0	1	1	2	2	2	3
2	0	1	1	2	2	3	3

Insertion de l'élément « 2 » :

1	0	1	1	2	2	2	3	
2	0	1	1	2	2	2	3	3

Suppression de l'élément « 3 » :

1	0	1	1	2	2	2	3
2	0	1	1	2	2	2	3

Les deux séquences sont maintenant identiques.

OM : exemple

Soient deux séquences :

1	0	1	1	2	2	2	3
2	0	1	1	2	2	3	3

Insertion de l'élément « 2 » :

1	0	1	1	2	2	2	3	
2	0	1	1	2	2	2	3	3

Suppression de l'élément « 3 » :

1	0	1	1	2	2	2	3	
2	0	1	1	2	2	2	3	

Les deux séquences sont maintenant identiques.

OM : exemple

Soient deux séquences :

1	0	1	1	2	2	2	3
2	0	1	1	2	2	3	3

Insertion de l'élément « 2 » :

1	0	1	1	2	2	2	3	
2	0	1	1	2	2	2	3	3

Suppression de l'élément « 3 » :

1	0	1	1	2	2	2	3
2	0	1	1	2	2	2	3

Les deux séquences sont maintenant identiques.

OM : exemple substitution

Soient deux séquences :

1	0	1	1	2	2	2	3
2	0	1	1	2	2	3	3

Substitution de l'élément « 3 »par l'élément « 2 » :

1	0	1	1	2	2	2	3
2	0	1	1	2	2	2	3

OM : exemple substitution

Soient deux séquences :

1	0	1	1	2	2	2	3
2	0	1	1	2	2	3	3

Substitution de l'élément « 3 »par l'élément « 2 » :

1	0	1	1	2	2	2	3
2	0	1	1	2	2	2	3

Définition des coûts

- On attribue aux deux types d'opération (insertion/suppression et substitution) des coûts.
- Les coûts de substitution peuvent être définis sous la forme d'une matrice.

Définition des coûts

- On attribue aux deux types d'opération (insertion/suppression et substitution) des coûts.
- Les coûts de substitution peuvent être définis sous la forme d'une matrice.

Stratégie de définition des coûts

- Coûts uniques
- Taux de transition $c_{i,j} = c_{j,i} = 2 - p(i_t \mid j_{t-1}) - p(j_t \mid i_{t-1})$
- Coûts optimaux (Gauthier et al., 2008)
- Coûts définis « à la main »

Stratégie de définition des coûts

- Coûts uniques
- Taux de transition $c_{i,j} = c_{j,i} = 2 - p(i_t \mid j_{t-1}) - p(j_t \mid i_{t-1})$
- Coûts optimaux (Gauthier et al., 2008)
- Coûts définis « à la main »

Stratégie de définition des coûts

- Coûts uniques
- Taux de transition $c_{i,j} = c_{j,i} = 2 - p(i_t \mid j_{t-1}) - p(j_t \mid i_{t-1})$
- Coûts optimaux (Gauthier et al., 2008)
- Coûts définis « à la main »

Stratégie de définition des coûts

- Coûts uniques
- Taux de transition $c_{i,j} = c_{j,i} = 2 - p(i_t \mid j_{t-1}) - p(j_t \mid i_{t-1})$
- Coûts optimaux (Gauthier et al., 2008)
- Coûts définis « à la main »

Mise en œuvre de l'optimal matching dans TraMineR

- Création de l'objet séquence avec `seqdef()`
- Création de la matrice des coûts de substitution avec `seqsubm()`
- Calcul de la matrice des distances avec la fonction `seqdist(..., method="OM", indel=..., sm=...)`

Mise en œuvre de l'optimal matching dans TraMineR

- Création de l'objet séquence avec `seqdef()`
- Création de la matrice des coûts de substitution avec `seqsubm()`
- Calcul de la matrice des distances avec la fonction `seqdist(..., method="OM", indel=..., sm=...)`

Mise en œuvre de l'optimal matching dans TraMineR

- Création de l'objet séquence avec `seqdef()`
- Création de la matrice des coûts de substitution avec `seqsubm()`
- Calcul de la matrice des distances avec la fonction `seqdist(..., method="OM", indel=..., sm=...)`

Matrice des coûts : coûts uniques

```
R> subm.unique <- seqsubm(mvad.seq, method = "CONSTANT", cval = 2)
R> subm.unique
```

	EM->	FE->	HE->	JL->	SC->	TR->
EM->	0	2	2	2	2	2
FE->	2	0	2	2	2	2
HE->	2	2	0	2	2	2
JL->	2	2	2	0	2	2
SC->	2	2	2	2	0	2
TR->	2	2	2	2	2	0

Matrice des coûts : coûts à la main

```
R> subm.custom <- matrix(c(0, 1, 1, 2, 1, 1, 1, 0, 1, 2,
+       1, 2, 1, 1, 0, 3, 1, 2, 2, 2, 3, 0, 3, 1, 1, 1, 1,
+       3, 0, 2, 1, 2, 2, 1, 2, 0), nrow = 6, ncol = 6, byrow = TRUE,
+       dimnames = list(mvad.shortlab, mvad.shortlab))
R> subm.custom
```

	EM	FE	HE	JL	SC	TR
EM	0	1	1	2	1	1
FE	1	0	1	2	1	2
HE	1	1	0	3	1	2
JL	2	2	3	0	3	1
SC	1	1	1	3	0	2
TR	1	2	2	1	2	0

Matrice des coûts : taux de transition

```
R> subm.txrate <- seqsubm(mvad.seq, method = "TRATE")
R> subm.txrate
```

	EM->	FE->	HE->	JL->	SC->	TR->
EM->	0.00000	1.97008	1.98723	1.95173	1.98536	1.95950
FE->	1.97008	0.00000	1.99318	1.98266	1.99092	1.99235
HE->	1.98723	1.99318	0.00000	1.99584	1.98184	1.99949
JL->	1.95173	1.98266	1.99584	0.00000	1.99385	1.97808
SC->	1.98536	1.99092	1.98184	1.99385	0.00000	1.99666
TR->	1.95950	1.99235	1.99949	1.97808	1.99666	0.00000

Le calcul des distances

- Une fois que les coûts de substitution ont été définis, on peut calculer les distances :

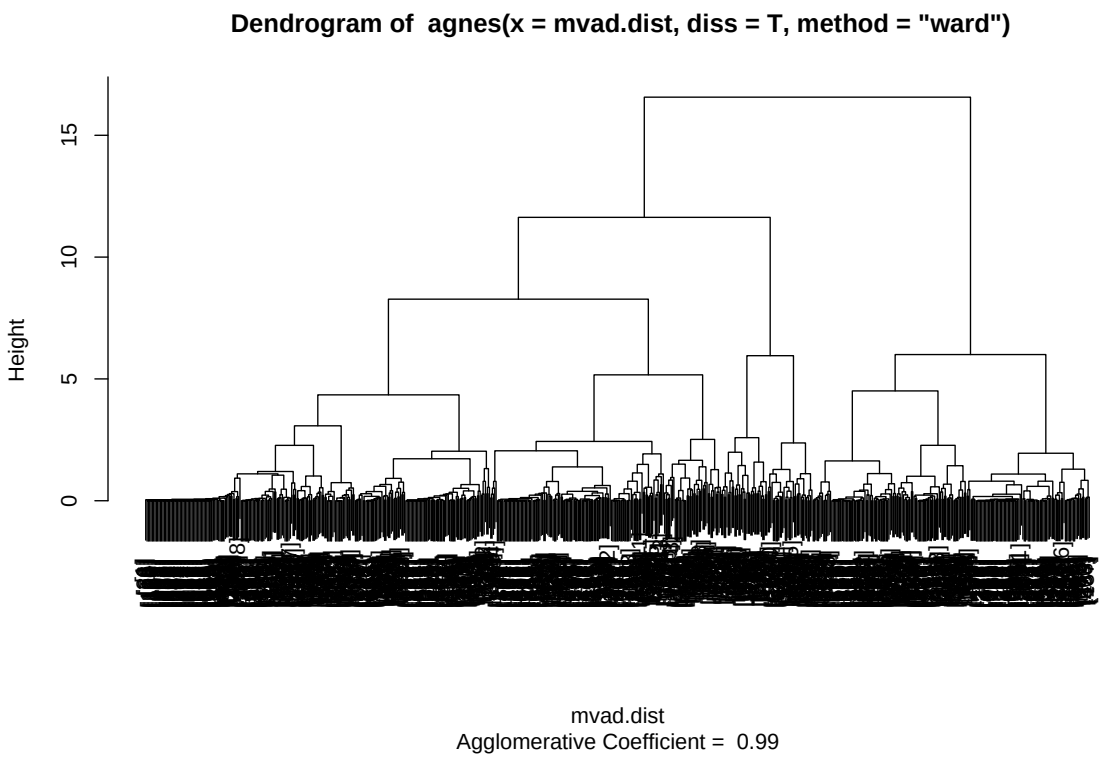
```
R> mvad.dist <- seqdist(mvad.seq, method = "OM", indel = 4,
+        sm = subm.custom, norm = TRUE)

R> round(mvad.dist[1:10, 1:10], digits = 2)
```

```
      [1]  [2]  [3]  [4]  [5]  [6]  [7]  [8]  [9] [10]
[1]  0.00 1.03 0.86 0.90 1.03 0.47 0.46 0.34 0.27 0.57
[2]  1.03 0.00 1.23 1.93 0.16 1.49 0.57 0.69 1.30 1.37
[3]  0.86 1.23 0.00 1.01 1.39 0.70 1.14 1.20 0.59 1.26
[4]  0.90 1.93 1.01 0.00 1.93 0.46 1.36 1.24 0.63 0.90
[5]  1.03 0.16 1.39 1.93 0.00 1.49 0.64 0.69 1.30 1.37
[6]  0.47 1.49 0.70 0.46 1.49 0.00 0.91 0.80 0.20 0.99
[7]  0.46 0.57 1.14 1.36 0.64 0.91 0.00 0.11 0.73 0.80
[8]  0.34 0.69 1.20 1.24 0.69 0.80 0.11 0.00 0.61 0.69
[9]  0.27 1.30 0.59 0.63 1.30 0.20 0.73 0.61 0.00 0.79
[10] 0.57 1.37 1.26 0.90 1.37 0.99 0.80 0.69 0.79 0.00
```

Structure de la section

Classification hiérarchique (Ward)

```
R> library(cluster)
R> mvad.clusterward <- agnes(mvad.dist, diss = T, method = "ward")
R> plot(mvad.clusterward, ask = F, which.plots = 2)
```



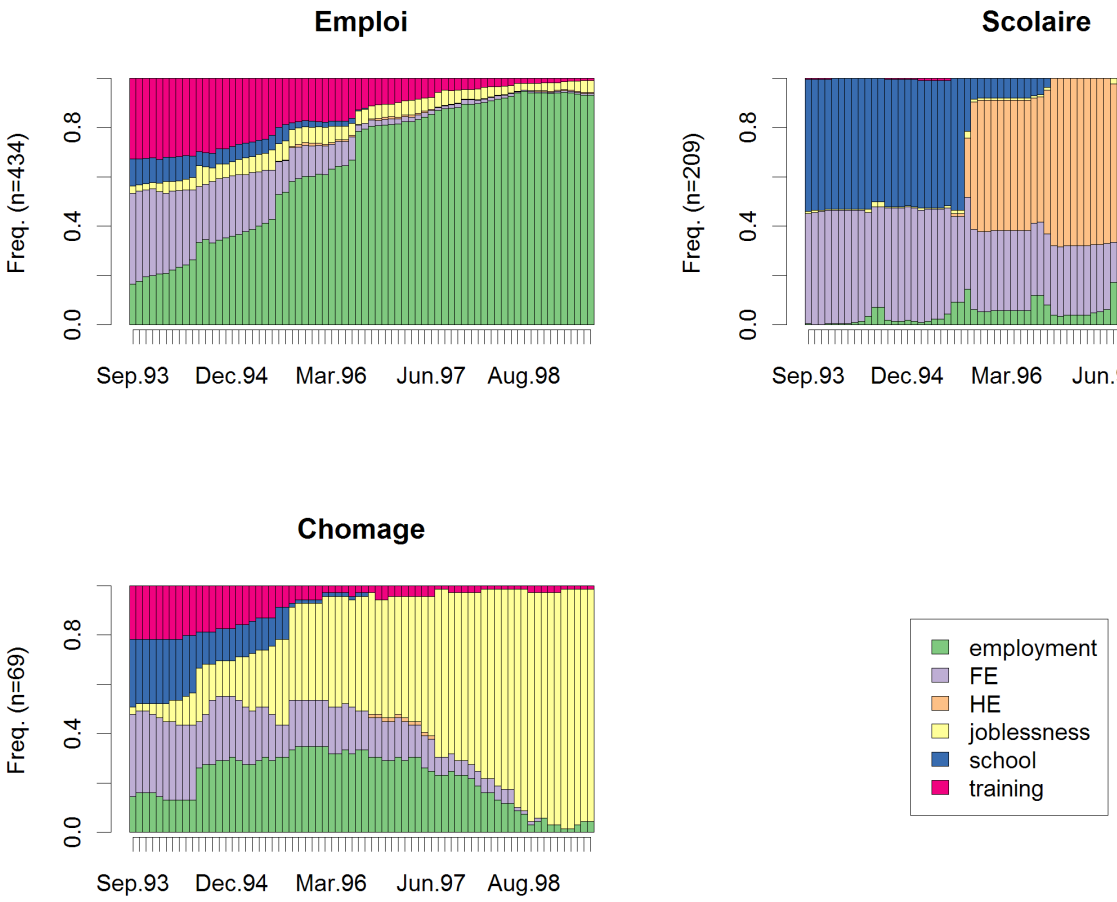
Récupération des groupes d'appartenance

- Une fois l'analyse en cluster effectuée, on choisit le nombre de groupes.
- On coupe l'arbre à la hauteur désirée, et on place l'appartenance au groupe dans un vecteur :

```
R> mvad.cl3 <- cutree(mvad.clusterward, k = 3)
R> mvad.cl3[1:10]

 [1] 1 2 1 1 2 1 1 1 1 3

R> mvad.cl3.factor <- factor(mvad.cl3, levels = c(1, 2,
+       3), labels = c("Emploi", "Scolaire", "Chomage"))
```

Distributions transversales

```
R> seqdplot(mvad.seq, group = mvad.cl3.factor)
```

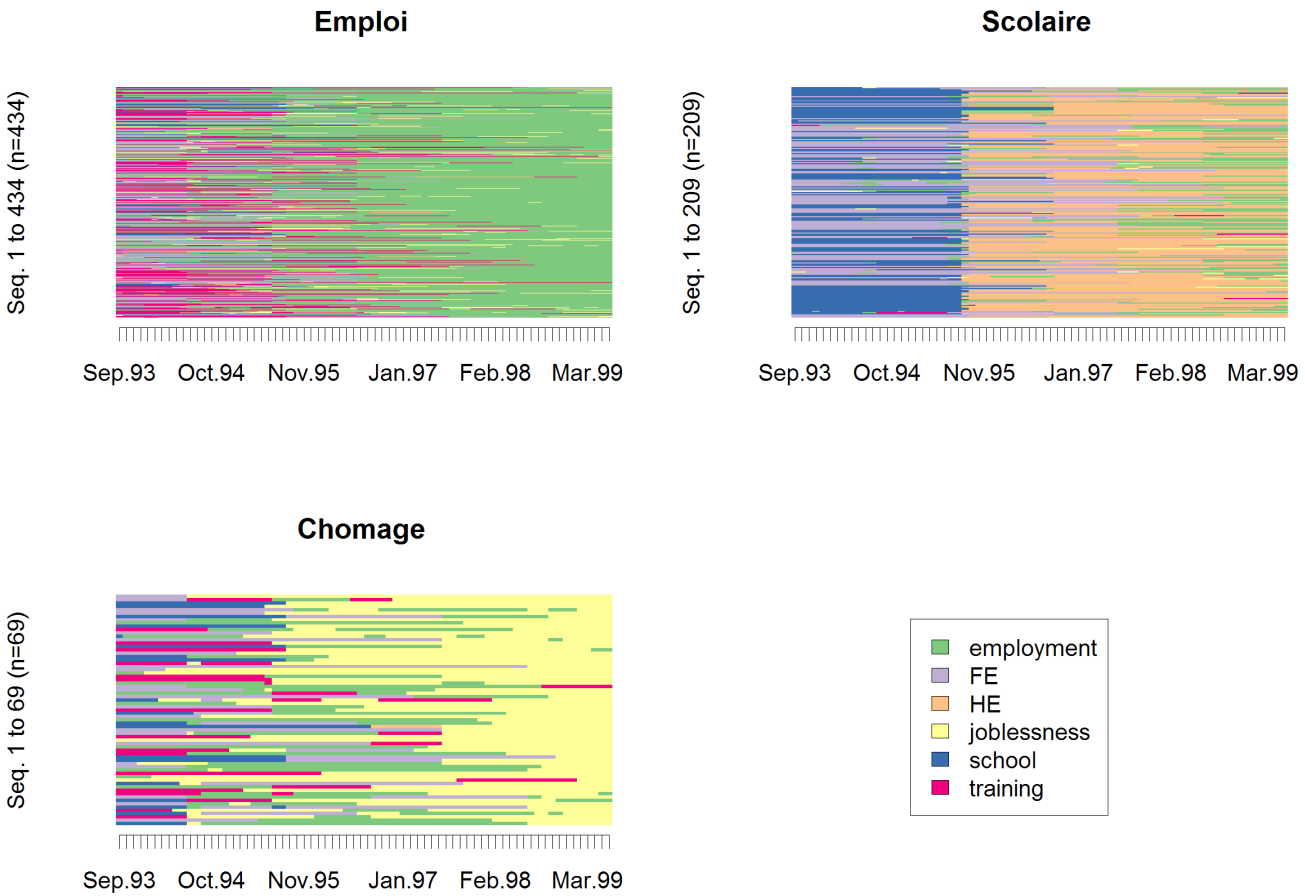


Séquences les plus fréquentes

```
R> seqfplot(mvad.seq, group = mvad.cl3.factor)
```

Distributions individuelles

```
R> seqiplot(mvad.seq, group = mvad.cl3.factor, tlim = 0, border = NA,
+         space = 0)
```

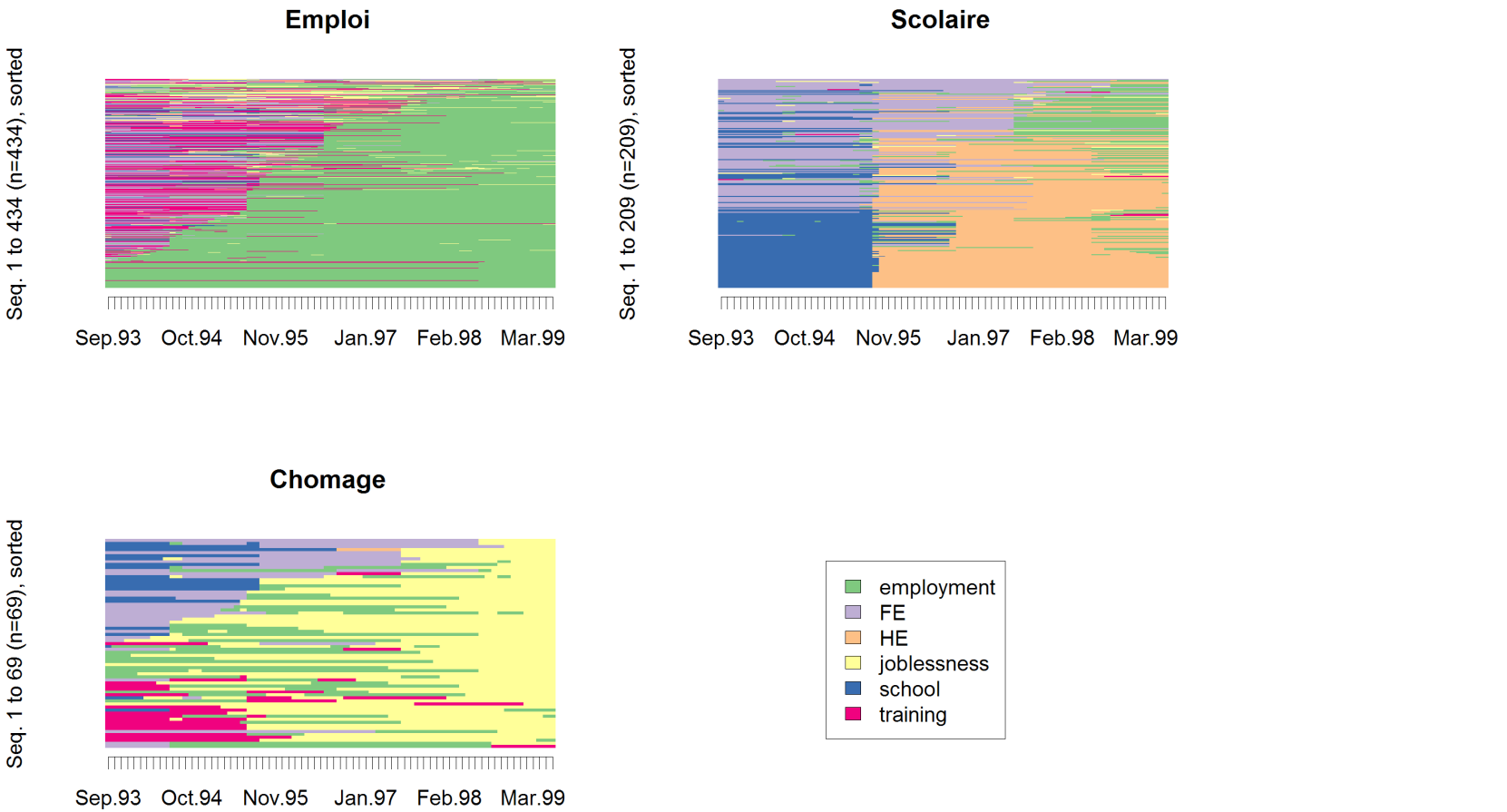


Distance à la séquence la plus fréquente

- Le graphique précédent est plus lisible si on ordonne les séquences
- Selon distance à la séquence la plus fréquente

```
R> mvad.distom <- numeric(nrow(mvad))
R> mvad.distom[mvad.cl3 == 1] <- seqdist(mvad.seq[mvad.cl3 ==
+       1, ], refseq = 0, method = "OM", indel = 4, sm = subm.custom)
R> mvad.distom[mvad.cl3 == 2] <- seqdist(mvad.seq[mvad.cl3 ==
+       2, ], refseq = 0, method = "OM", indel = 4, sm = subm.custom)
R> mvad.distom[mvad.cl3 == 3] <- seqdist(mvad.seq[mvad.cl3 ==
+       3, ], refseq = 0, method = "OM", indel = 4, sm = subm.custom)
```

```
R> seqiplot(mvad.seq, group = mvad.cl3.factor, tlim = 0, border = NA,
+         space = 0, sortv = mvad.distom)
```

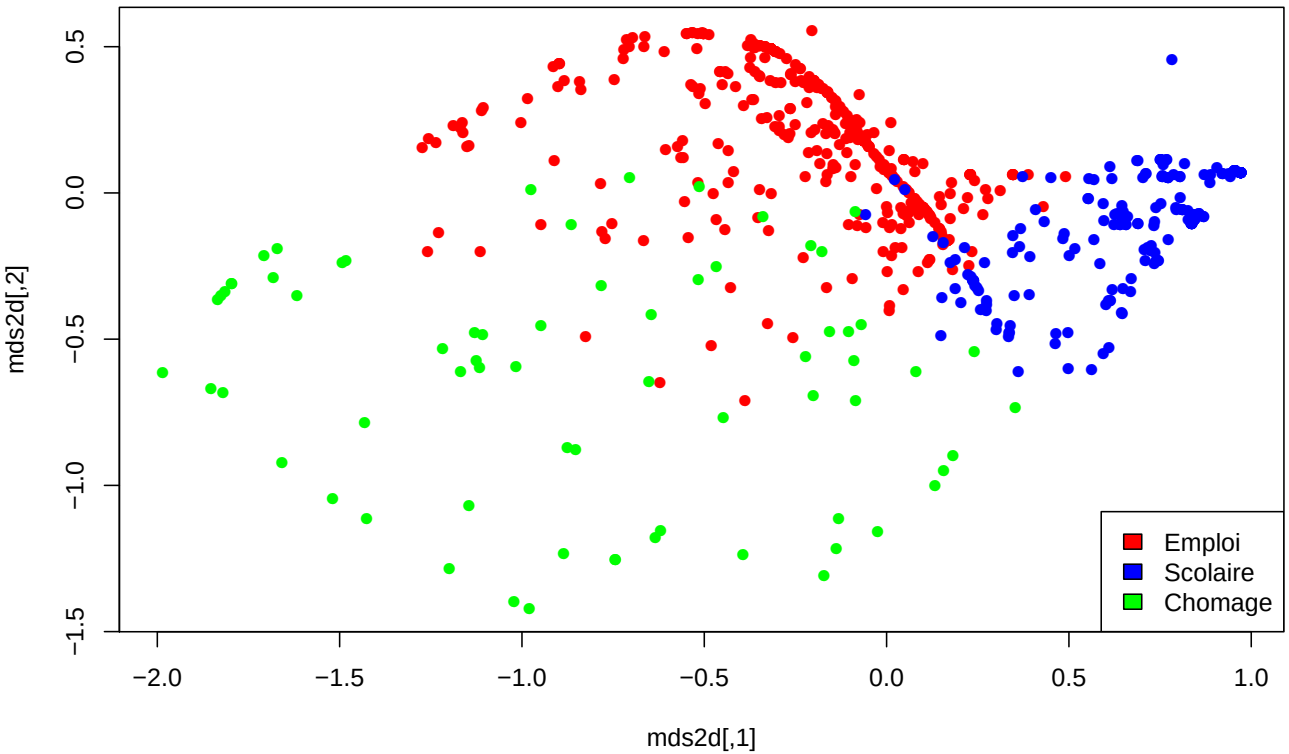


Nuage de points (MDS)

- En utilisant le multidimensional scaling (MDS), on peut visualiser les séquences sous forme de nuage de points

```
R> mds2d <- cmdscale(mvad.dist, k = 2)

R> plot(mds2d, type = "n")
R> points(mds2d[mvad.cl3 == 1, ], pch = 16, col = "red")
R> points(mds2d[mvad.cl3 == 2, ], pch = 16, col = "blue")
R> points(mds2d[mvad.cl3 == 3, ], pch = 16, col = "green")
R> legend("bottomright", fill = c("red", "blue", "green"),
+       legend = c("Emploi", "Scolaire", "Chomage"))
```

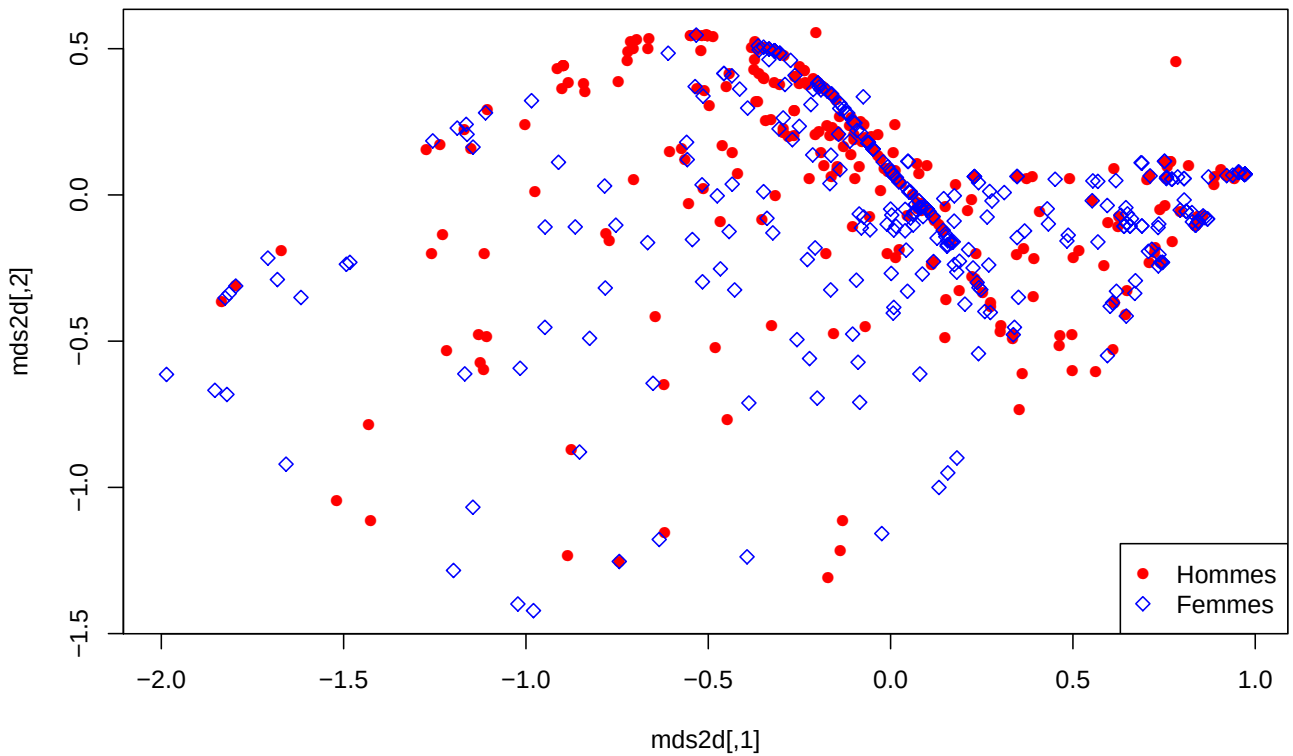
Plot des nuages par cluster



Plot du nuage de points, selon sexe

```
R> plot(mds2d, type = "n")
R> points(mds2d[mvad$male == "yes", ], pch = 16, col = "red")
R> points(mds2d[mvad$male == "no", ], pch = 23, col = "blue")
R> legend("bottomright", col = c("red", "blue"), pch = c(16,
+       23), legend = c("Hommes", "Femmes"))
```

Plot des nuages par sexe

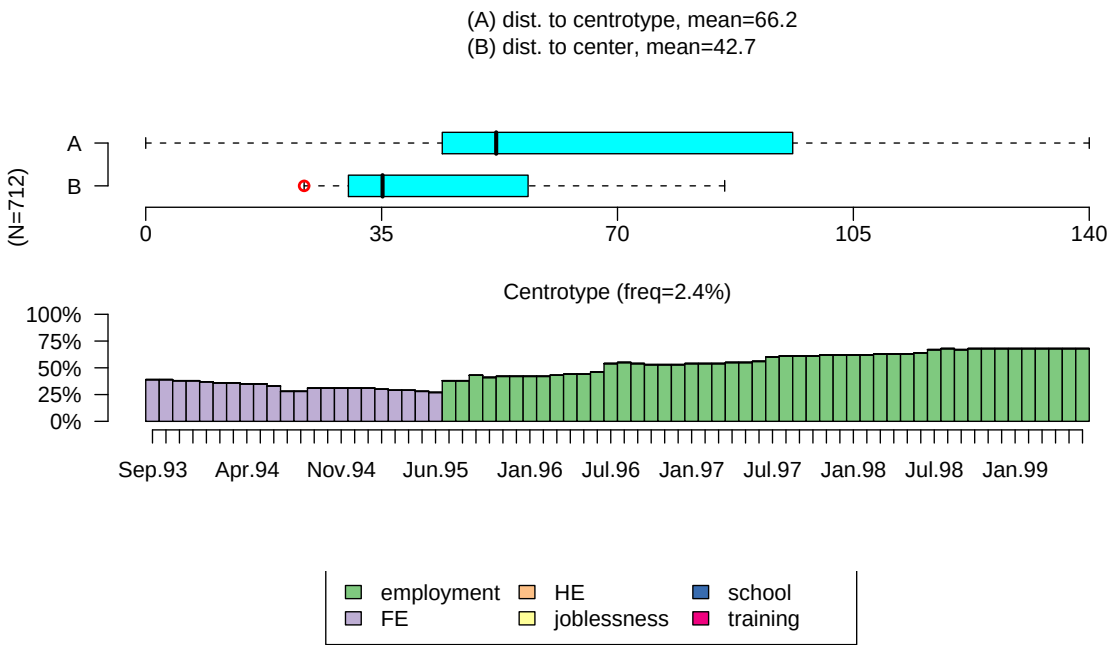


Structure de la section

Centrotype

- Séquence dont la somme des distances aux autres est minimale.

```
R> TraMineR:::seqcplot(mvad.seq, dist.matrix = distMatLCS, method = "dist")
```



Dispersion de l'ensemble de séquences

- A partir d'une matrice de distances, on peut calculer la pseudo-inertie de l'ensemble des séquences
- Somme des carrés SS s'exprime en termes des distances 2 à 2

$$SS = \sum_{i=1}^{n}(y_i - \bar{y})^2 = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=i+1}^{n}(y_i - y_j)^2$$

$$= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=i+1}^{n} d_{ij}$$

- En posant d_{ij} égal à la dist OM, LCP, LCS, ... , on obtient notre SS.
- Peut appliquer principe de l'ANOVA (voir présentation Matthias jeudi).

Dispersion de l'ensemble de séquences

- A partir d'une matrice de distances, on peut calculer la pseudo-inertie de l'ensemble des séquences
- Somme des carrés SS s'exprime en termes des distances 2 à 2

$$SS = \sum_{i=1}^{n}(y_i - \bar{y})^2 = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=i+1}^{n}(y_i - y_j)^2$$

$$= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=i+1}^{n} d_{ij}$$

- En posant d_{ij} égal à la dist OM, LCP, LCS, ... , on obtient notre SS.
- Peut appliquer principe de l'ANOVA (voir présentation Matthias jeudi).

Dispersion de l'ensemble de séquences

- A partir d'une matrice de distances, on peut calculer la pseudo-inertie de l'ensemble des séquences
- Somme des carrés SS s'exprime en termes des distances 2 à 2

$$SS = \sum_{i=1}^{n}(y_i - \bar{y})^2 = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=i+1}^{n}(y_i - y_j)^2$$

$$= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=i+1}^{n} d_{ij}$$

- En posant d_{ij} égal à la dist OM, LCP, LCS, ... , on obtient notre SS.
- Peut appliquer principe de l'ANOVA (voir présentation Matthias jeudi).

Calcul de la dispersion des séquences

```
R> distMatLCS[1:6, 1:7]
```

```
     [1] [2] [3] [4] [5] [6] [7]
[1]   0 140 116 108 140  64  60
[2] 140   0  72 140  22 140  80
[3] 116  72   0  68  90  72  60
[4] 108 140  68   0 140  46 112
[5] 140  22  90 140   0 140  90
[6]  64 140  72  46 140   0  68
```

```
R> dissvar(distMatLCS)
```

```
[1] 42.74502
```

Plan

Structure de la section

Objectifs

- Analyse de séquences d'événements
  - On se centre sur les événements.
  - On cherche à comprendre la succession des événements.
  - Existe-t-il des "patterns" typiques d'événements ?
  - Quels patterns dépendent d'un facteur donné ?
- Recherche de patterns d'états.

Objectifs

- Analyse de séquences d'événements
  - On se centre sur les événements.
  - On cherche à comprendre la succession des événements.
  - Existe-t-il des "patterns" typiques d'événements ?
  - Quels patterns dépendent d'un facteur donné ?
- Recherche de patterns d'états.

Quelques définitions

- Séquence d'événements : suite ordonnée de transitions.
- Transition : ensemble non-ordonné d'événements.

Exemple

(Départ, Couple) $\rightarrow$ (Mariage) $\rightarrow$ (Enfant)

- (Départ, Couple) et (Mariage) sont des transitions.
- "Départ", "Couple" et "Mariage" sont des événements.

Quelques définitions

- Séquence d'événements : suite ordonnée de transitions.
- Transition : ensemble non-ordonné d'événements.

Exemple

(Départ, Couple) → (Mariage) → (Enfant)

- (Départ, Couple) et (Mariage) sont des transitions.
- "Départ", "Couple" et "Mariage" sont des événements.

Sous-séquence

- Une sous-séquence est une séquence d'événements où l'ordre d'apparition des événements et des transitions est respecté.

Exemple

A (Départ, Couple) $\rightarrow$ (Mariage) $\rightarrow$ (Enfant).

B (Départ, Mariage) $\rightarrow$ (Enfant).

C (Départ) $\rightarrow$ (Enfant).

- C est une sous-séquence de A et B, car l'ordre des événements est conservé.
- B n'est pas une sous-séquence de A, car l'ordre des événements est différent à cause de "Mariage".

Sous-séquence

Une sous-séquence est dite :

- fréquente si on l'observe un nombre minimum de fois, appelé le support minimum.
- discriminante si sa fréquence d'apparition dépend significativement d'un facteur.

Sous-séquence

Une sous-séquence est dite :

- fréquente si on l'observe un nombre minimum de fois, appelé le support minimum.
- discriminante si sa fréquence d'apparition dépend significativement d'un facteur.

Sous-séquence

Une sous-séquence est dite :

- fréquente si on l'observe un nombre minimum de fois, appelé le support minimum.
- discriminante si sa fréquence d'apparition dépend significativement d'un facteur.

Structure de la section

Format de donnée

- Dans TraMineR, on doit créer un objet séquences d'événements avec `seqecreate()`
- Peut se faire à partir de plusieurs formats
  - "Time Stamped Event" (TSE) qui permet de spécifier ses propres événements.
  - Séquence d'états, avec plusieurs options pour la conversion automatique :
    - **transition** Un événement distinct pour chaque transition entre deux états.
    - **state** Un pour chaque début de période dans un nouvel état.
    - **period** Un événement pour chaque début et chaque fin de période dans un état.

Format de donnée

- Dans TraMineR, on doit créer un objet séquences d'événements avec `seqecreate()`
- Peut se faire à partir de plusieurs formats
  - "Time Stamped Event" (TSE) qui permet de spécifier ses propres événements.
  - Séquence d'états, avec plusieurs options pour la conversion automatique :
    - **transition** Un événement distinct pour chaque transition entre deux états.
    - **state** Un pour chaque début de période dans un nouvel état.
    - **period** Un événement pour chaque début et chaque fin de période dans un état.

Format de donnée TSE

Le format "Time Stamped Event" (TSE) :

- **id** Identifiant individuel.
- **timestamp** Date de l'événement au format numérique (réel).
- **event** Type d'événement.
- Une ligne par événement.

```
R> data(actcal.tse)
R> head(actcal.tse)
```

```
  id time      event
1  1    0   PartTime
2  2    0 NoActivity
3  2    4      Start
4  2    4   FullTime
5  2   11       Stop
6  3    0   PartTime
```

Création d'un objet séquences d'événements

Depuis le format TSE

- Fonction `seqecreate()`.
- Les paramètres `id`, `timestamp` et `event` permettent de spécifier les colonnes du format TSE.

```
R> actcal.seqe <- seqecreate(id = actcal.tse$id, timestamp = actcal.tse$time,
+        event = actcal.tse$event)
```

Création d'un objet séquences d'événements

Depuis un objet séquences d'états

- Fonction `seqecreate()`.
- Le paramètre `tevent` permet de contrôler la conversion automatique.
- Ici, un événement par transition.

```
R> data(mvad)
R> mvad.shortlab <- c("EM", "FE", "HE", "JL", "SC", "TR")
R> mvad.seq <- seqdef(mvad[, 17:86], labels = mvad.shortlab)
R> mvad.seqe <- seqecreate(mvad.seq, tevent = "transition")
```

Matrices des transitions

La conversion se fait à l'aide d'une matrice de transitions

```
R> seqetm(mvad.seq, method = "transition")
```

	employment	FE	HE	joblessness	school	training
employment	"EM"	"EM>FE"	"EM>HE"	"EM>JL"	"EM>SC"	"EM>TR"
FE	"FE>EM"	"FE"	"FE>HE"	"FE>JL"	"FE>SC"	"FE>TR"
HE	"HE>EM"	"HE>FE"	"HE"	"HE>JL"	"HE>SC"	"HE>TR"
joblessness	"JL>EM"	"JL>FE"	"JL>HE"	"JL"	"JL>SC"	"JL>TR"
school	"SC>EM"	"SC>FE"	"SC>HE"	"SC>JL"	"SC"	"SC>TR"
training	"TR>EM"	"TR>FE"	"TR>HE"	"TR>JL"	"TR>SC"	"TR"

Représentation textuelle

- Chaque séquence est présentée sous la forme

  `(e1,e2,...)-temps-(e2,...)-temps`
- où `(e1,e2,...)` désigne une transition comportant un ou plusieurs événements simultanés.
- "`temps`" désigne le temps écoulé entre deux événements ou jusqu'à la fin de la séquence (temps d'observation)

```
R> print(mvad.seqe[2])
[1] (FE)-36.00-(FE>HE)-34.00
```

Représentation textuelle

- Chaque séquence est présentée sous la forme

  `(e1,e2,...)-temps-(e2,...)-temps`
- où `(e1,e2,...)` désigne une transition comportant un ou plusieurs événements simultanés.
- "`temps`" désigne le temps écoulé entre deux événements ou jusqu'à la fin de la séquence (temps d'observation)

```
R> print(mvad.seqe[2])
[1] (FE)-36.00-(FE>HE)-34.00
```

Structure de la section

Recherche de sous-séquences fréquentes

On doit spécifier

- Les séquences d'événements.
- Le support minimum (`pMinSupport`).

```
R> mvad.fsubseq <- seqefsub(mvad.seqe, pMinSupport = 0.01)
R> mvad.fsubseq[1:5]

   Subsequence      data
1         (FE) 0.3862360
2      (FE>EM) 0.2879213
3      (TR>EM) 0.2528090
4         (SC) 0.2514045
5 (FE)-(FE>EM) 0.2289326

Computed on 712 event sequences with the following contraints
 Constraint Value
     maxGap    NA
 windowSize    NA
     ageMin    NA
     ageMax    NA
  ageMaxEnd    NA
```

Recherche de sous-séquences fréquentes

On doit spécifier

- Les séquences d'événements.
- Le support minimum (`pMinSupport`).

```
R> mvad.fsubseq <- seqefsub(mvad.seqe, pMinSupport = 0.01)
R> mvad.fsubseq[1:5]

   Subsequence      data
1         (FE) 0.3862360
2      (FE>EM) 0.2879213
3      (TR>EM) 0.2528090
4         (SC) 0.2514045
5 (FE)-(FE>EM) 0.2289326

Computed on 712 event sequences with the following contraints
 Constraint Value
     maxGap    NA
 windowSize    NA
     ageMin    NA
     ageMax    NA
  ageMaxEnd    NA
```

Recherche de sous-séquences fréquentes

On doit spécifier

- Les séquences d'événements.
- Le support minimum (`pMinSupport`).

```
R> mvad.fsubseq <- seqefsub(mvad.seqe, pMinSupport = 0.01)
R> mvad.fsubseq[1:5]
```

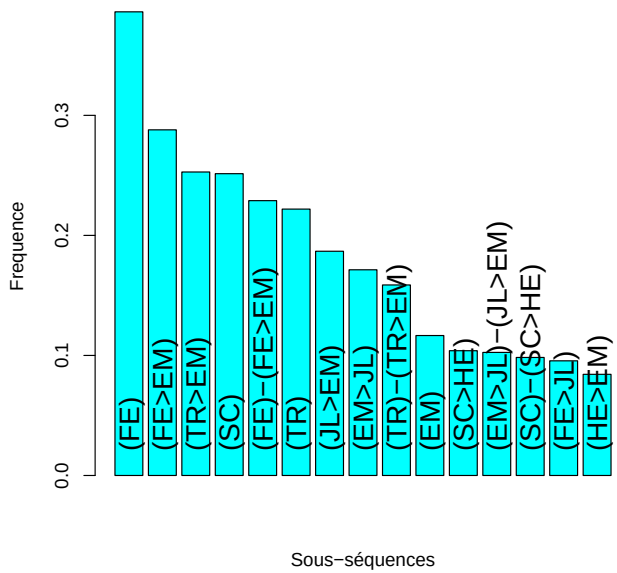
```
   Subsequence      data
1         (FE) 0.3862360
2      (FE>EM) 0.2879213
3      (TR>EM) 0.2528090
4         (SC) 0.2514045
5 (FE)-(FE>EM) 0.2289326

Computed on 712 event sequences with the following contraints
 Constraint Value
     maxGap    NA
 windowSize    NA
     ageMin    NA
     ageMax    NA
  ageMaxEnd    NA
```

Graphique de fréquences

La commande `plot` peut être utilisée directement sur les résultats

- On sélectionne les sous-séquences à inclure à l'aide des indices.
- Les autres paramètres sont ceux de la fonction `barplot()`

```
R> plot(mvad.fsubseq[1:15], col = "cyan", ylab = "Frequence",
+       xlab = "Sous-séquences", cex = 1.5)
```



Recherche de sous-séquences discriminantes

- On cherche à identifier les sous-séquences fréquentes les plus fortement associées à un facteur.
- Le pouvoir discriminant est déterminé à l'aide d'un test du χ^2.
- On peut effectuer une correction de Bonferroni à l'aide du paramètre `method="bonferroni"`.

Recherche de sous-séquences discriminantes

- On cherche à identifier les sous-séquences fréquentes les plus fortement associées à un facteur.
- Le pouvoir discriminant est déterminé à l'aide d'un test du χ^2.
- On peut effectuer une correction de Bonferroni à l'aide du paramètre `method="bonferroni"`.

Recherche de sous-séquences discriminantes

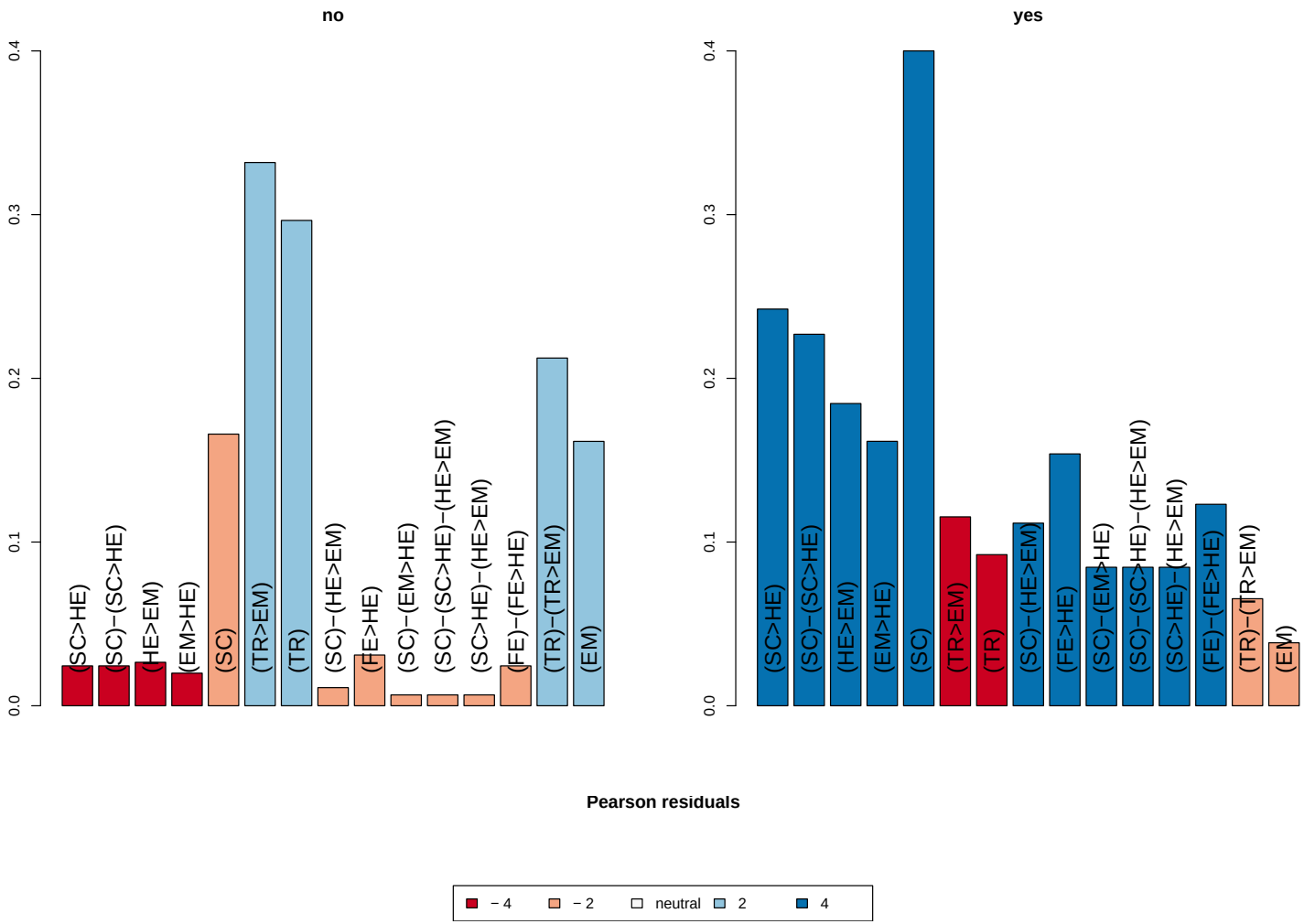
- On cherche à identifier les sous-séquences fréquentes les plus fortement associées à un facteur.
- Le pouvoir discriminant est déterminé à l'aide d'un test du χ^2.
- On peut effectuer une correction de Bonferroni à l'aide du paramètre `method="bonferroni"`.

Recherche de sous-séquences discriminantes

```
R> mvad.discr <- seqecmpgroup(mvad.fsubseq, group = mvad$gcse5eq)
R> mvad.discr[1:5]
   Subsequence subseq.data.cres...      p.value statistic index    Freq.no
1      (SC>HE)          0.10393258 1.445408e-19  81.88088    11 0.02433628
2 (SC)-(SC>HE)          0.09831461 7.250286e-18  74.14723    13 0.02433628
3      (HE>EM)          0.08426966 7.487216e-13  51.41219    15 0.02654867
4      (EM>HE)          0.07162921 5.019013e-12  47.67954    21 0.01991150
5         (SC)          0.25140449 7.798571e-12  46.81571     4 0.16592920
   Freq.yes  Resid.no Resid.yes
1 0.2423077 -5.249117  6.920999
2 0.2269231 -5.016083  6.613742
3 0.1846154 -4.227342  5.573781
4 0.1615385 -4.108312  5.416839
5 0.4000000 -3.624293  4.778657

Computed on 712 event sequences with the following contraints
 Constraint Value
     maxGap    NA
 windowSize    NA
     ageMin    NA
     ageMax    NA
  ageMaxEnd    NA
```

Représentation graphique

```
R> plot(mvad.discr[1:15], cex = 1.5)
```



Structure de la section

Recherche de patterns d'états

- En associant un événement au début de chaque épisode passé dans un état
- les sous-séquences fréquentes correspondent à des patterns d'états
- On peut ainsi par exemple chercher les patterns les plus discriminants entre clusters.

```
R> mvad.pat <- seqecreate(mvad.seq, tevent = "state")
R> mvad.pat.fsubseq <- seqefsub(mvad.pat, pMinSupport = 0.01)
R> discr.pat.cluster <- seqecmpgroup(mvad.pat.fsubseq, group = mvad.cl3.factor)
R> plot(discr.pat.cluster[1:10])
```

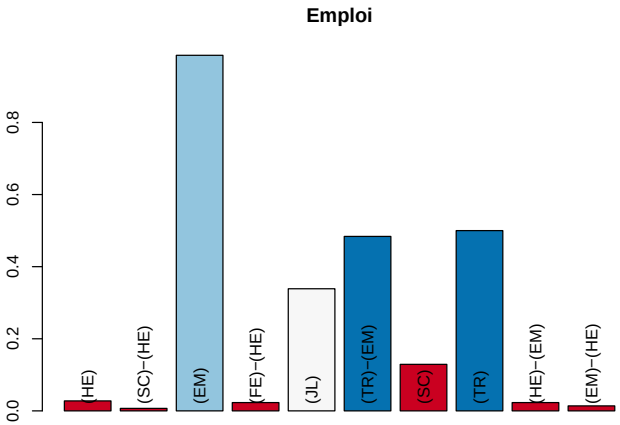
Recherche de patterns d'états

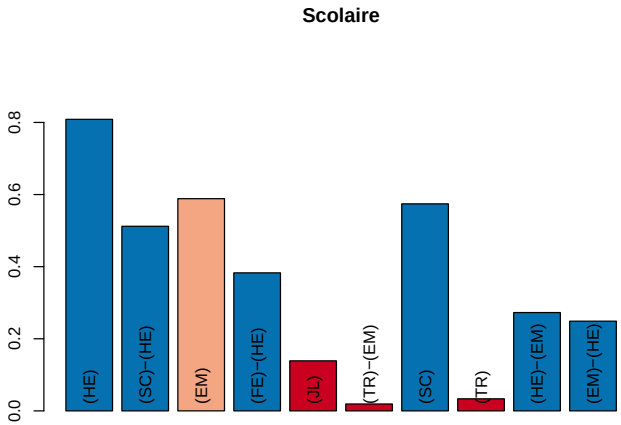
- En associant un événement au début de chaque épisode passé dans un état
- les sous-séquences fréquentes correspondent à des patterns d'états
- On peut ainsi par exemple chercher les patterns les plus discriminants entre clusters.

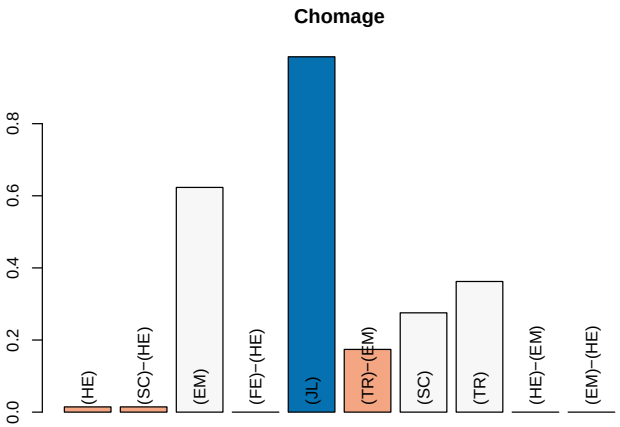
```
R> mvad.pat <- seqecreate(mvad.seq, tevent = "state")
R> mvad.pat.fsubseq <- seqefsub(mvad.pat, pMinSupport = 0.01)
R> discr.pat.cluster <- seqecmpgroup(mvad.pat.fsubseq, group = mvad.cl3.factor)
R> plot(discr.pat.cluster[1:10])
```

Patterns d'états discriminants

Les 10 plus discriminants



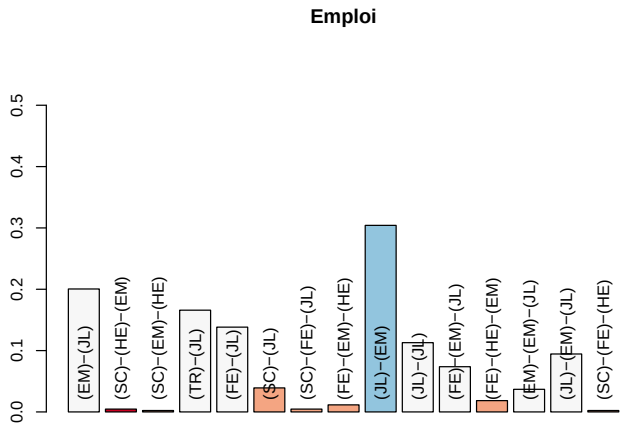


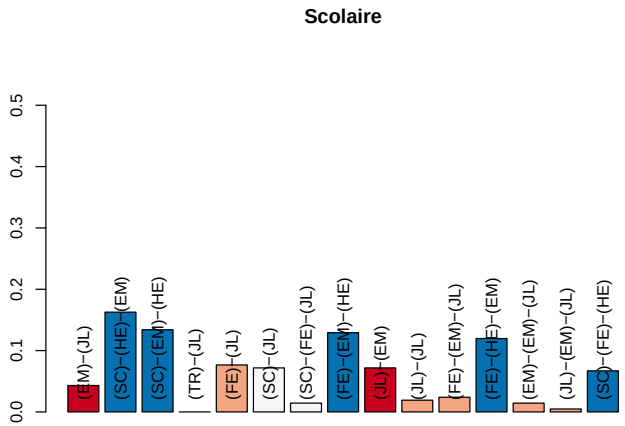


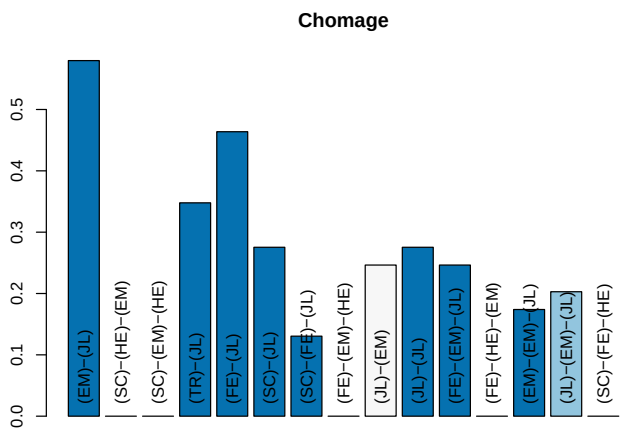
Patterns d'états discriminants

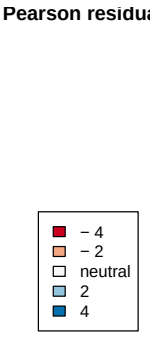
Les 15 suivants

```
R> plot(discr.pat.cluster[11:25])
```









Structure de la section

Recherche de sous-séquences spécifiques

- On recherche des sous-séquences prédéfinies.
- Par exemple, (JL) → (EM) et (EM) → (JL).

```
R> subseq <- c("(JL)-(EM)", "(EM)-(JL)")
R> mysubseq <- seqefsub(mvad.pat, strsubseq = subseq)
R> mysubseq
```

```
          Subsequence   Support
(JL)-(EM)   (JL)-(EM) 0.2303371
(EM)-(JL)   (EM)-(JL) 0.1910112

Computed on 712 event sequences with the following contraints
 Constraint Value
     maxGap    NA
 windowSize    NA
     ageMin    NA
     ageMax    NA
  ageMaxEnd    NA
```

Matrices des occurrences

La matrices des occurrences permet de :

- Compter le nombre d'occurrences.
- Calculer l'âge à la première occurrence.

```
R> mysubseq.occ <- seqeapplysub(mysubseq, method = "count")
R> mysubseq.occ[c(655, 701), ]
```

	(JL)-(EM)	(EM)-(JL)
(SC)-24.00-(JL)-3.00-(EM)-43.00	1	0
(FE)-4.00-(EM)-39.00-(JL)-13.00-(EM)-14.00	1	1

```
R> mysubseq.age <- seqeapplysub(mysubseq, method = "age")
R> mysubseq.age[c(655, 701), ]
```

	(JL)-(EM)	(EM)-(JL)
(SC)-24.00-(JL)-3.00-(EM)-43.00	24	-1
(FE)-4.00-(EM)-39.00-(JL)-13.00-(EM)-14.00	43	4

Structure de la section

Contraintes temporelles

On peut contraindre la recherche à l'aide de la fonction `seqeconstraint()` qui prend les paramètres suivants :

- `maxGap` Temps maximum entre deux événements.
- `windowSize` Durée maximale de la sous-séquence.
- `ageMin` Âge minimum au début de la sous-séquence
- `ageMax` Âge maximum au début de la sous-séquence
- `ageMaxEnd` Âge maximum à la fin de la sous-séquence

On peut n'en spécifier que quelques-un.

Contraintes temporelles

On peut contraindre la recherche à l'aide de la fonction `seqeconstraint()` qui prend les paramètres suivants :

- `maxGap` Temps maximum entre deux événements.
- `windowSize` Durée maximale de la sous-séquence.
- `ageMin` Âge minimum au début de la sous-séquence
- `ageMax` Âge maximum au début de la sous-séquence
- `ageMaxEnd` Âge maximum à la fin de la sous-séquence

On peut n'en spécifier que quelques-un.

Contraintes temporelles

On peut contraindre la recherche à l'aide de la fonction `seqeconstraint()` qui prend les paramètres suivants :

- `maxGap` Temps maximum entre deux événements.
- `windowSize` Durée maximale de la sous-séquence.
- `ageMin` Âge minimum au début de la sous-séquence
- `ageMax` Âge maximum au début de la sous-séquence
- `ageMaxEnd` Âge maximum à la fin de la sous-séquence

On peut n'en spécifier que quelques-un.

Contraintes temporelles

On peut contraindre la recherche à l'aide de la fonction `seqeconstraint()` qui prend les paramètres suivants :

- `maxGap` Temps maximum entre deux événements.
- `windowSize` Durée maximale de la sous-séquence.
- `ageMin` Âge minimum au début de la sous-séquence
- `ageMax` Âge maximum au début de la sous-séquence
- `ageMaxEnd` Âge maximum à la fin de la sous-séquence

On peut n'en spécifier que quelques-un.

Contraintes temporelles

```
R> myconstraint <- seqeconstraint(windowSize = 6)
R> mysubseq <- seqefsub(mvad.pat, constraint = myconstraint,
+       pMinSupport = 0.01)
R> mysubseq[5:10]
```

```
  Subsequence       data
1        (SC) 0.27387640
2        (HE) 0.25561798
3   (JL)-(EM) 0.11938202
4   (EM)-(JL) 0.05477528
5   (EM)-(HE) 0.04915730
6   (TR)-(EM) 0.04073034

Computed on 712 event sequences with the following contraints
 Constraint Value
     maxGap    NA
 windowSize     6
     ageMin    NA
     ageMax    NA
  ageMaxEnd    NA
```

Contraintes temporelles

```
R> myconstraint <- seqeconstraint(maxGap = 2, ageMin = 12)
R> mysubseq <- seqefsub(mvad.pat, constraint = myconstraint,
+       pMinSupport = 0.01)
R> mysubseq[1:5]
```

```
  Subsequence      data
1        (EM) 0.6783708
2        (JL) 0.2921348
3        (HE) 0.2556180
4        (FE) 0.1460674
5        (TR) 0.1207865

Computed on 712 event sequences with the following contraints
 Constraint Value
     maxGap     2
 windowSize    NA
     ageMin    12
     ageMax    NA
  ageMaxEnd    NA
```

Plan

Séquence d'analyse (séquences d'états) - I

- Examiner distribution des séquences (seqdplot, seqfplot, seqiplot)
- Caractéristiques de l'ensemble des séquences
  - Séquence représentative (la plus fréquente, la plus centrale, ...)
  - Dispersion des séquences (à partir des mesures de dissimilarités)
  - Séquence de caractéristiques transversales (entropies, état modal, ...)
  - Distribution de caractéristiques longitudinales (entropie, turbulence, durée dans chaque état, ...)
    - Liens entre caractéristiques longitudinales de séquences parallèles (famille-profession, ego-partenaire, ...)
- Analyses précédentes par groupes (sexe, cohorte de naissance, ...), comparaisons

Séquence d'analyse (séquences d'états) - I

- Examiner distribution des séquences (seqdplot, seqfplot, seqiplot)
- Caractéristiques de l'ensemble des séquences
  - Séquence représentative (la plus fréquente, la plus centrale, ...)
  - Dispersion des séquences (à partir des mesures de dissimilarités)
  - Séquence de caractéristiques transversales (entropies, état modal, ...)
  - Distribution de caractéristiques longitudinales (entropie, turbulence, durée dans chaque état, ...)
    - Liens entre caractéristiques longitudinales de séquences parallèles (famille-profession, ego-partenaire, ...)
- Analyses précédentes par groupes (sexe, cohorte de naissance, ...), comparaisons

Séquence d'analyse (séquences d'états) - I

- Examiner distribution des séquences (seqdplot, seqfplot, seqiplot)
- Caractéristiques de l'ensemble des séquences
  - Séquence représentative (la plus fréquente, la plus centrale, ...)
  - Dispersion des séquences (à partir des mesures de dissimilarités)
  - Séquence de caractéristiques transversales (entropies, état modal, ...)
  - Distribution de caractéristiques longitudinales (entropie, turbulence, durée dans chaque état, ...)
    - Liens entre caractéristiques longitudinales de séquences parallèles (famille-profession, ego-partenaire, ...)
- Analyses précédentes par groupes (sexe, cohorte de naissance, ...), comparaisons

Séquence d'analyse (séquences d'états) - I

- Examiner distribution des séquences (seqdplot, seqfplot, seqiplot)
- Caractéristiques de l'ensemble des séquences
  - Séquence représentative (la plus fréquente, la plus centrale, ...)
  - Dispersion des séquences (à partir des mesures de dissimilarités)
  - Séquence de caractéristiques transversales (entropies, état modal, ...)
  - Distribution de caractéristiques longitudinales (entropie, turbulence, durée dans chaque état, ...)
    - Liens entre caractéristiques longitudinales de séquences parallèles (famille-profession, ego-partenaire, ...)
- Analyses précédentes par groupes (sexe, cohorte de naissance, ...), comparaisons

Séquence d'analyse (séquences d'états) - II

- Etudes des proximités entre séquences individuelles
  - Recherche de typologies (analyse en clusters)
    - Liens entre clusters et facteurs explicatifs (sexe, cohorte, ...), modèles logit ...
  - Diagramme de dispersion (Multi-dimensional scaling)
  - ANOH (Analyse de l'hétérogénéité) : Part de l'hétérogénéité expliquée par un ou des facteurs.
  - Segmentation en groupes homogènes par arbre d'induction

Séquences d'analyse (séquences d'événements)

- Recherche des sous-séquences fréquentes
- Dans le cadre de contraintes sur
  - Evénements initiaux et finaux (leaving home et mariage par exemple)
  - Age minimal et maximal (entre 20 et 50 ans)
  - Durée maximale (10 ans)
- Liens entre événement fréquent et facteurs explicatifs (sexe, cohorte, ...), modèles logistiques.
- Effet de connaître l'événement sur entropie, ou autre variable dépendante ...
- Recherche des événements les plus discriminants pour une variable catégorielle donnée (sexe, cluster, ...)

Séquences d'analyse (séquences d'événements)

- Recherche des sous-séquences fréquentes
- Dans le cadre de contraintes sur
  - Evénements initiaux et finaux (leaving home et mariage par exemple)
  - Age minimal et maximal (entre 20 et 50 ans)
  - Durée maximale (10 ans)
- Liens entre événement fréquent et facteurs explicatifs (sexe, cohorte, ...), modèles logistiques.
- Effet de connaître l'événement sur entropie, ou autre variable dépendante ...
- Recherche des événements les plus discriminants pour une variable catégorielle donnée (sexe, cluster, ...)

Séquences d'analyse (séquences d'événements)

- Recherche des sous-séquences fréquentes
- Dans le cadre de contraintes sur
  - Evénements initiaux et finaux (leaving home et mariage par exemple)
  - Age minimal et maximal (entre 20 et 50 ans)
  - Durée maximale (10 ans)
- Liens entre événement fréquent et facteurs explicatifs (sexe, cohorte, ...), modèles logistiques.
- Effet de connaître l'événement sur entropie, ou autre variable dépendante ...
- Recherche des événements les plus discriminants pour une variable catégorielle donnée (sexe, cluster, ...)

Séquences d'analyse (séquences d'événements)

- Recherche des sous-séquences fréquentes
- Dans le cadre de contraintes sur
  - Evénements initiaux et finaux (leaving home et mariage par exemple)
  - Age minimal et maximal (entre 20 et 50 ans)
  - Durée maximale (10 ans)
- Liens entre événement fréquent et facteurs explicatifs (sexe, cohorte, ...), modèles logistiques.
- Effet de connaître l'événement sur entropie, ou autre variable dépendante ...
- Recherche des événements les plus discriminants pour une variable catégorielle donnée (sexe, cluster, ...)

Plan

Références I

Abbott, A. and J. Forrest (1986). Optimal matching methods for historical sequences. *Journal of Interdisciplinary History 16*, 471–494.

Abbott, A. and A. Tsay (2000). Sequence analysis and optimal matching methods in sociology, Review and prospect. *Sociological Methods and Research 29*(1), 3–33. (With discussion, pp 34-76).

Berchtold, A. and A. E. Raftery (2002). The mixture transition distribution model for high-order Markov chains and non-gaussian time series. *Statistical Science 17*(3), 328–356.

Billari, F. C. (2001). The analysis of early life courses: Complex description of the transition to adulthood. *Journal of Population Research 18*(2), 119–142.

Brzinsky-Fay, C., U. Kohler, and M. Luniak (2006). Sequence analysis with Stata. *The Stata Journal 6*(4), 435–460.

Elzinga, C. H. (2008). Sequence analysis: Metric representations of categorical time series. *Sociological Methods and Research*. forthcoming.

Références II

Elzinga, C. H. and A. C. Liefbroer (2007). De-standardization of family-life trajectories of young adults: A cross-national comparison using sequence analysis. *European Journal of Population 23*, 225–250.

Gabadinho, A., G. Ritschard, M. Studer, and N. S. Müller (2008). Mining sequence data in R with TraMineR: A user's guide. Technical report, Department of Econometrics and Laboratory of Demography, University of Geneva, Geneva. (TraMineR is on CRAN the Comprehensive R Archive Network).

Gauthier, J.-A., E. D. Widmer, P. Bucher, and C. Notredame (2008). How much does it cost? Optimization of costs in sequence analysis of social science data. *Sociological Methods and Research*. (forthcoming).

Hosmer, D. W. and S. Lemeshow (1999). *Applied Survival Analysis, Regression Modeling of Time to Event Data*. New York: Wiley.

Hothorn, T., K. Hornik, and A. Zeileis (2006). party: A laboratory for recursive part(y)itioning. User's manual.

Références III

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.

Ritschard, G., A. Gabadinho, N. S. Müller, and M. Studer (2009). Mining event histories: A social science perspective. *International Journal of Data Mining, Modelling and Management 1*(1), 68–90.

Scherer, S. (2001). Early career patterns: A comparison of Great Britain and West Germany. *European Sociological Review 17*(2), 119–144.