Exercise 1  A first run with TraMineR

1. Start R and load the TraMineR library.

2. Look at the help page of the biofam data provided by TraMineR which you access by typing help(biofam) or ?biofam. Find out which are the columns containing the sequence data. Specify the meaning of the numerical state codes.

3. Look at the first six rows of the data frame (head(biofam)).

4. Create a basic state sequence object and plot the sequences using, seqIplot, seqfplot and seqdplot. Comment the plots.

5. Display (print) the first 10 sequences in extended and compact form.

Exercise 2  R basics

1. Print the variable names of the biofam data frame.

2. Create an age variable by subtracting the birth year from the year of the survey and add it to the biofam data frame.

3. What is the minimum, maximum, median and mean age in the sample? (Hint: use summary())

4. What is the minimum, maximum, median and mean age of the women?

5. Add a cohort factor to the biofam data frame grouping the birth years into the following categories: 1900-1929, 1930-1939, 1940-1949, 1950-1959. (Hint: use cut())

6. Generate an histogram of the distribution of birthyear using the above birth year classes. (Look at the help of the hist function for how to do that.)

7. Produce a frequency table of the cohort factor.

8. Cross tabulate the cohort with the state at 25 years old.
Exercise 3  Rendering state sequences

1. Create a weighted state sequence object named `biofam.seq` with variables `a15` to `a30` of `biofam`, using the following state names and labels

<table>
<thead>
<tr>
<th>State</th>
<th>Name</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>P</td>
<td>Parent</td>
</tr>
<tr>
<td>1</td>
<td>L</td>
<td>Left</td>
</tr>
<tr>
<td>2</td>
<td>M</td>
<td>Married</td>
</tr>
<tr>
<td>3</td>
<td>LM</td>
<td>Left/Married</td>
</tr>
<tr>
<td>4</td>
<td>C</td>
<td>Child</td>
</tr>
<tr>
<td>5</td>
<td>LC</td>
<td>Left/Child</td>
</tr>
<tr>
<td>6</td>
<td>LMC</td>
<td>Left/Married/Child</td>
</tr>
<tr>
<td>7</td>
<td>D</td>
<td>Divorced</td>
</tr>
</tbody>
</table>

and the weights respecting the sample size.

2. Create a full sequence index plot sorted from the end for each class of the `cohort` variable created in exercise 2.

3. Print the frequencies of the first 20 sequences.

4. Create a sequence frequency plot of the 20 most frequent patterns grouped by values of the `cohort` variable and save it as a ‘jpeg’ file.

5. Compute the transition rate matrix for the `biofam.seq` sequences.

6. What is the transition rate between states ‘Left/Married’ and ‘Left/Married/Child’?

7. Display the sequence of cross-sectional state distributions by cohort.

8. Within each cohort, at what age is the cross-sectional diversity of the states at its highest?

9. Display side by side in a same plot area the mean times spent in each of the states and the sequence of modal states.

Exercise 4  Longitudinal characteristics

1. Continuing with the `biofam` data, build a table with the sequence length, the number of transitions, the number of subsequences, the longitudinal entropy, the turbulence and the complexity index.

2. Using `summary()`, look at the min, max, mean, median and quartiles of the distribution of each of the computed longitudinal characteristics.
3. Display the histogram of each longitudinal characteristic but the length in a same graphic.

4. Generate the sequences of distinct successive states (DSS) and the table with the duration in the distinct successive states. Display the last 6 of them.

5. Generate a scatterplot matrix for comparing the Entropy with the Turbulence and Complexity Index.

6. Compare the distributions of the complexity index by birth cohorts using boxplots.

7. Regress the complexity index on the birth cohort, the sex and the language of the questionnaire. Comment the results.

Exercise 5  Measuring pairwise dissimilarities

1. Compute the matrix of pairwise HAM distances between the biofam.seq sequences and display the results for the first 5 sequences.

2. Plot the first 2 sequences and check that the HAM distance is the number of non matching positions between them.

3. Check on the biofam data that the LCS distance provides the same non-normalized distances as OM with indel=1 and a constant substitution cost of 2 which we denote as OM(1,2).

4. The alphabet of biofam is based on the occurrences of four events: left home, getting married, childbirth and divorce. In order to define a substitution cost matrix reflecting the number of events which distinguish the states, build a state-properties matrix with a row per state and a column per event. Fill the matrix with ones indicating the events which should be experienced to be in the given state, and zero otherwise. When a state is independent of whether an event occurred or not indicate .5 in the corresponding cell.

5. Define the substitution cost matrix as the Euclidean distance between the rows of the previously defined state-properties matrix. (Tip: use dist to get the Euclidean distance and as.matrix to transform the result into a matrix.) Set the indel cost as half the maximum substitution cost.

6. Compute the OM dissimilarity matrix using the previously derived substitution and indel costs, and name this matrix dOM. Create a two column table with the dLCS dissimilarities as first column and the dOM dissimilarities as second column. (Hint: Use as.vector to put the elements of each dissimilarity matrix into a single column.) Make a scatterplot to compare the obtained dissimilarities for a random sample of 500 LCS distances.
Exercise 6  Clustering sequence data

1. From the previously computed dOM dissimilarity matrix, create a hierarchical cluster tree object with Ward method using sequence weights in each case (Tip: Transform the dOM matrix into a distance object with as.dist(dOM), and retrieve the weights with attr(biofam.seq, "weight").)

2. Display the hierarchical tree.

3. Select the three-cluster solution from the Ward analysis, and label the clusters by looking at the I-plots by cluster.