Notes on likelihood, p. 35–41

06 May 2021

Contents

Notes on section 2.3.1, p. 35–36

Create matrix

The transition probability matrix (**t.p.b**), $\mathbf{\Gamma}$, must have its column summing to one (sum of probability of transition). The stationnary distrib $\boldsymbol{\delta}$ is such that:

$$\begin{aligned} \boldsymbol{\delta}\mathbf{\Gamma} &= \boldsymbol{\delta} \\ \boldsymbol{\delta}\mathbb{1} &= 1 \end{aligned} \tag{1}$$

where $\mathbb{1} = (1, \ldots, 1)$. The second equation basically says $\sum_i \delta_i = 1$!

```
gammaMatrix = matrix(data = c(1/2, 1/2, 1/4, 3/4), nrow = 2, byrow = TRUE)

if (unique(rowSums(gammaMatrix)) != 1)
    warning("The sum of gamma's columns should be 1!")

## Stationnary distribution
delta = 1/3*c(1, 2)

# Check sum is one
sum(delta)
```

```
## [1] 1
```

```
t(delta) %*% gammaMatrix - t(delta)
```

```
##      [,1] [,2]
## [1,]    0    0
```

```
t(gammaMatrix) %*% delta == delta
```

```
##      [,1]
## [1,] TRUE
## [2,] TRUE
```

Get equation 2.11 and table 2.1

The law of total proba is enough to derive the equation (2.11), and then use equation (2.5) remembering that the parent of X_i is C_i, and that the parent of C_i is C_{i-1}:

$$\begin{aligned}
\Pr(X_1 = 1, X_2 = 1, X_3 = 1) &= \sum_i \sum_j \sum_k \Pr(X_1 = 1, X_2 = 1, X_3 = 1, C_1 = i, C_2 = j, C_3 = k) \\
&= \sum_i \sum_j \sum_k \Pr(C_1 = i)\Pr(X_1 = 1|C_1 = i)\times \\
&\qquad \Pr(C_2 = j|C_1 = i)\Pr(X_2 = 1|C_2 = j)\times \\
&\qquad \Pr(X_3 = 1|C_3 = k)\Pr(C_3 = k|C_2 = j) \\
&= \sum_i \sum_j \sum_k \delta_i p_i(1)\gamma_{ij} p_j(1)\gamma_{jk} p_k(1)
\end{aligned}$$

In the followinr R code, `Pr_X_C` denotes the proba $\Pr(X = x|C = c)$. In this example, C has only two possible states: 1 and 2, and X is a bernoulli variable, *i.e.*, 0 or 1.

```
# Pr_X_C = Pr(X = x | C = c)
Pr_0_1 = 1/2
Pr_1_1 = 1/2

Pr_0_2 = 1
Pr_1_2 = 1

combinations = expand.grid(rep(list(1:2), 3))
setDT(combinations)
setnames(combinations, new = c("i", "j", "k"))

setorderv(combinations, cols = colnames(combinations), order=1L)

# List all the way to get X1 = 1, X2 = 1, and X3 = 1
combinations[i == 1, pi_1 := Pr_1_1]
combinations[i == 2, pi_1 := Pr_1_2]

combinations[j == 1, pj_1 := Pr_1_1]
combinations[j == 2, pj_1 := Pr_1_2]

combinations[k == 1, pk_1 := Pr_1_1]
combinations[k == 2, pk_1 := Pr_1_2]

combinations[i == 1, delta_i := 1/3]
combinations[i == 2, delta_i := 2/3]

combinations[(i == 1) & (j == 1), gamma_ij := gammaMatrix[1, 1]]
combinations[(i == 1) & (j == 2), gamma_ij := gammaMatrix[1, 2]]
combinations[(i == 2) & (j == 1), gamma_ij := gammaMatrix[2, 1]]
combinations[(i == 2) & (j == 2), gamma_ij := gammaMatrix[2, 2]]

combinations[(j == 1) & (k == 1), gamma_jk := gammaMatrix[1, 1]]
combinations[(j == 1) & (k == 2), gamma_jk := gammaMatrix[1, 2]]
combinations[(j == 2) & (k == 1), gamma_jk := gammaMatrix[2, 1]]
combinations[(j == 2) & (k == 2), gamma_jk := gammaMatrix[2, 2]]

combinations[, product := delta_i*pi_1 * gamma_ij*pj_1 * gamma_jk*pk_1]
```

Each row correspond to one combination of (i, j, k). The sum gives the probability $\Pr(X_1 = 1, X_2 = 1, X_3 = 1) = 29/48$

Likelihood in general (p. 36)

The formulae to remember is that the likelihood, L_T, of an observation sequence $c_1, x_2, \dots, x_T$ is:

$$L_T = \boldsymbol{\delta} \boldsymbol{P}(x_1) \boldsymbol{\Gamma} \boldsymbol{P}(x_2) \cdots \boldsymbol{\Gamma} \boldsymbol{P}(x_T) \mathbb{1}'$$

and in the case δ is the stationnary distribution associated to $\boldsymbol{\Gamma}$, then we get:

$$L_T = \boldsymbol{\Gamma} \boldsymbol{\delta} \boldsymbol{P}(x_1) \boldsymbol{\Gamma} \boldsymbol{P}(x_2) \cdots \boldsymbol{\Gamma} \boldsymbol{P}(x_T) \mathbb{1}'$$

which is easier to code!

Based on p. 38, the likelihood of $\Pr(X_1 = 1, X_2 = 1, X_3 = 1)$ can be rewritten as:

```
T = 3
P1 = diag(c(Pr_1_1, Pr_1_2), nrow = 2, ncol = 2)
alpha = delta # The stationnary distrib
for (i in 1:3)
    alpha = alpha %*% gammaMatrix %*% P1

L_T = MASS::fractions(sum(alpha))
print(paste0("L_T = ", L_T))
```

```
## [1] "L_T = 29/48"
```

Good to see they are the same! However, for long series, it will be better to use the log-likelihood, in order to avoid **underflow**!

Likelihood with missing data

It is still possible to compute the likelihood when data are missing! In this case, just replace $P(x_j)$ by the identity matrix, where x_j is a missing data.