Analysis of Archaeological Phases using the CRAN Package ArchaeoPhases

Anne Philippe
Université de Nantes, LMJL

Marie-Anne Vibet
Université de Nantes, LMJL

Abstract

We propose new statistical tools to analyze and to estimate archeological phases from the posterior distribution of a sequence of dates. All the functions are implemented in the R package **ArchaeoPhases**. The required inputs are simulated samples from the posterior distribution of the dates. Such Markov Chains Monte Carlo samples are provided, for instance, by ChronoModel or Oxcal, two softwares build for the chronological modeling of archeological dates. We give a practical introduction to the package **ArchaeoPhases** using published data and comment the statistical results.

Keywords: Bayesian modeling, Gap interval, Gap test, MCMC samples, Tempo plot, Time range, Transition interval.

1. Introduction

The statistical modeling within the Bayesian framework is widely used by archaeologists for constructing chronologies. This consists in the estimation of a sequence of dates where

- the observations are the measurements coming from possibly different dating methods,
- the prior information comes from geological, historical, environmental or any other considerations.

Bayesian inference is a probabilistic estimation method, in the sense that the information on the estimated parameters is given by a probability distribution called posterior probability. Usually the analytic expression of the posterior distribution is not easily computable. However, inferences may be drawn using a sample from the posterior density. More precisely, it is possible to compute a Markov chain whose target distribution is the posterior distribution.

Indeed, Markov chain Monte Carlo (MCMC) algorithms provide a way of drawing samples from the joint posterior distribution in high-dimensional Bayesian models.

Two main softwares are available to estimate chronologies: Chronomodel (see Lanos, Philippe, Lanos, and Dufresne (2016); Vibet, Philippe, Lanos, and Dufresne (2016); Lanos and Philippe (2017)) and Oxcal (see Ramsey (2016, 2009)). Both softwares return the marginal posterior distribution of all the estimated dates with elementary statistics (mean, standard deviation, HPD region,...). It is also possible to export the output from MCMC simulations (see Section 3 for the details). Thus, we can explore the joint distribution of the sequence of dates. We propose new statistical tools for analysing archaeological phases based on this joint distribution. For example we construct:

- Time intervals to characterize archaeological phases or periods, called **time range intervals**.
- Time intervals to characterize the transition between two successive phases or periods, called **transition intervals**.
- **Testing procedures to check the presence of a gap** between two successive phases or periods. A **gap interval** is estimated if we accept its existence.
- **Tempo plots** as defined by Thomas Dye (see Dye (2016)), that is "a statistical graphic designed for the archaeological study of rhythms of the long term that embodies a theory of archaeological evidence for the occurrence of events" and **their associated activity plots**.

The CRAN package **ArchaeoPhases** provides a list of functions for calculating these statistics. The inputs are samples simulated from the posterior distribution of dates (e.g. the Markov Chains simulated by Chronomodel or by Oxcal). For non-**R** users, a web application, ArchaeoPhases, has also been developed in order to take advantages of these functions without having to know R. This application is freely available on `https://archaeology-bayesian-modelling.shinyapps.io/ArchaeoPhases` and within the package ArchaeoPhases (`app_ArchaeoPhases()`).

In Section 2, we describe new statistical tools for analysing groups of estimated dates in a Bayesian framework. Section 3 provides a short description of the data published by Bosch, Mannino, Prendergast, O'Connell, Demarchi, Taylor, Niven, van der Plicht, and Hublin (2015) used to illustrate **ArchaeoPhases**. We construct a Bayesian model with Chronomodel and with Oxcal in order to estimate four Paleolithic phases. We also explain how the MCMC samples can be extracted from both softwares. In Section 4, we give a practical introduction to the first steps with **ArchaeoPhases** (version 1.0, Febuary 2017). Then in Section 5, we analyse the MCMC output of the modeling done with Chronomodel and estimate the four Paleolithic phases of Ksar Akil's site.

2. Statistical aspects

We define a chronology as a sequence of calendar dates $\tau_1, \dots, \tau_n$. The archaeological question about a chronology is not only a matter of estimation of calendar dates $\tau_1, \dots, \tau_n$ but also the characterization of historical/geological periods called archaeological phases.

The estimation of the phase can only be done from the estimated dates associated to this phase.
We define a phase (denoted P) by a collection of dates, τ_i for $i \in I$ where I is a subset of $\{1, \dots, n\}$ and n the total number of calendar dates.

A phase is commonly summarized by two parameters: its start and its end. Two different approaches are considered in order to estimate these two dates.

1. The start and the end are modeled, and so additional parameters are incorporated in the model. This is the approach implemented in Oxcal where the parameters are denoted t_a and t_b. This requires to do prior assumption on the distribution of the dates on the estimated period $[t_a, t_b]$.
2. The start (resp. the end) is estimated by the minimum (resp. the maximum) of the dates included in the phase. Thus the approach does not integrate a model of the phase, but it is only a post-processing step based on the estimation of a sequence of dates. These statistics are implemented in ChronoModel and Oxcal.

Our approach is in the same spirit of the second one in the sense that we propose statistical tools of post-processing without additional modeling.

Notation : we denote by $\mathcal{M}$ the set of measurements coming from dating methods. We assume that MCMC samples from the joint posterior distribution $p(\tau_1, \dots, \tau_n \,|\, \mathcal{M})$ of all the dates $\tau_1, \dots, \tau_n$ are available (using for instance ChronoModel or Oxcal).

2.1. Time Range interval of a phase

We propose to characterize a phase using a time interval called a **time range interval**. A time range interval summarizes the dates defining the phase by a time period having fixed posterior probability. It gives also an idea of the start, the end and the duration of a phase.

Definition 1. *Let $P = \{\tau_i \; i \in I \subset \{1, \dots, n\}\}$ be a phase. The $100(1-\gamma)\%$ phase time range is the shortest interval $[a, b]$ such that*

$$P(\tau_i \in]a, b[, \; \forall i \in I) | \mathcal{M}) = 1 - \gamma \qquad (1)$$

This means that the time interval $[a, b]$ contains all the dates $\{\tau_i \; i \in I\}$ with a fixed posterior probability $1 - \gamma$ (e.g., 95 %, 68 % ...).

We denote $\alpha = \min(\tau_i, \forall i \in I)$ and $\beta = \max(\tau_i, \forall i \in I)$. Equation 1 can be also rewritten of the form

$$P(a \leq \alpha \leq \beta \leq b \,|\, \mathcal{M}) = 1 - \gamma \qquad (2)$$

Therefore the construction of the time range interval depends only on the joint posterior distribution of (α, β).

The collection of intervals $[a, b]$ satisfying Equation 2 is described by this set of intervals:

$$\{[a(\epsilon) \, ; \, b(\epsilon)], \quad \epsilon \in [0, \gamma]\}$$

where

- $a(\epsilon) = F_{\alpha}^{-}(\epsilon \,|\, \mathcal{M})$ with $F_{\alpha}^{-}(\cdot \,|\, \mathcal{M})$ is the quantile function of the posterior distribution of α,
- $b(\epsilon) = F_{\beta}^{-}(\frac{1-\gamma}{1-\epsilon} \,|\, \alpha \geq a(\epsilon), \mathcal{M})$ with $F_{\beta}^{-}(\cdot \,|\, \alpha \geq a(\epsilon), \mathcal{M})$ is the quantile function of the conditional distribution of β given $\alpha \geq a(\epsilon)$ and $\mathcal{M}$.

Indeed, $\alpha \leq \beta$ happens almost surely, so we have

$$\begin{aligned} P(a(\epsilon) \leq \alpha \leq \beta \leq b(\epsilon) \,|\, \mathcal{M}) &= P(a(\epsilon) \leq \alpha, \beta \leq b(\epsilon) \,|\, \mathcal{M}) \\ &= P(\beta \leq b(\epsilon) \,|\, a(\epsilon) \leq \alpha, \mathcal{M}) P(a(\epsilon) \leq \alpha \,|\, \mathcal{M}) \\ &= \frac{1-\gamma}{1-\epsilon}(1-\epsilon) \end{aligned}$$

The last equality comes from the definition of $a(\epsilon)$ and $b(\epsilon)$. Therefore the shortest interval is obtained by taking $[a(\epsilon^*), b(\epsilon^*)]$ where

$$\epsilon^* = \mathrm{argmin}_{\epsilon \in]0,\gamma[} \, (b(\epsilon) - a(\epsilon))$$

In practice, the values of $a(\epsilon)$ and $b(\epsilon)$ are estimated by the empirical quantiles calculated on the MCMC outputs. Using the output of the MCMC algorithm $(\tau_1^{(k)}, \ldots, \tau_n^{(k)})_{k=1,\ldots,N_{MCMC}}$, we produce $(\alpha^{(k)}, \beta^{(k)})_{t=1,\ldots,N_{MCMC}}$ the MCMC sample corresponding to the posterior distribution of (α, β),

$$\alpha^{(k)} = \min(\tau_i^{(k)},\ i \in I) \text{ and } \beta^{(k)} = \max(\tau_i^{(k)},\ i \in I).$$

$a(\epsilon)$ is estimated by the empirical quantile of the sample $\{\alpha(1), \ldots, \alpha(N_{MCMC})\}$ (sample corresponding to the marginal posterior distribution of α). $b(\epsilon)$ is estimated by the empirical quantile of $\{\beta(t), t \in T\}$ where $T = \{j \,|\, \alpha(j) \geq \epsilon\}$

2.2. Transition interval between two successive phases

The **transition interval** between two successive phases is the shortest interval that covers the end of the oldest phase and the start of the youngest phase. The start and the end are estimated by the minimum and the maximum of the dates included in the phase. It gives an idea of the transition period between two successive phases. From a computational point of view this is equivalent to the phase time range calculated between the end of the oldest phase and the start of the youngest phase.

Definition 2. *Consider a succession of two phases, P_1 and P_2. Assume P_1 is older than P_2. We denote by α_i (respectively β_i) the minimum (respectively the maximum) of the group of dates P_i ($i = 1, 2$). The $100(1-\gamma)\%$ transition interval is the shortest interval $[a, b]$ such that*

$$P(a \leq \beta_1 \leq \alpha_2 \leq b \,|\, \mathcal{M}) = 1 - \gamma \tag{3}$$

The construction of this interval is obtained by the same step as the range time interval by replacing (α, β) by (β_1, α_2).

2.3. Testing procedure for gap between two successive phases

Successive phases may also be separated in time. Indeed there may exist **a gap** between them. This testing procedure check whether a gap exists between two successive phases with fixed probability. If a gap exists, it is an interval that covers the end of one phase and the start of the successive one with fixed posterior probability.

Definition 3. *Consider a succession of two phases, P_1 and P_2. Assume P_1 is older than P_2. We denote by α_i (respectively β_i) the minimum (respectively the maximum) of the group of dates P_i ($i = 1, 2$).*

The $100(1-\gamma)\%$ gap between these successive phases (if it exists) is the longest interval $[a, b]$ such that

$$P(\beta_1 \le a \le b \le \alpha_2 \,|\, \mathcal{M}) = 1 - \gamma \tag{4}$$

The first step consists in the construction of all the couples $a(\epsilon), b(\epsilon)$ such that

$$P(\beta_1 \le a(\epsilon)\,,\, b(\epsilon) \le \alpha_2 \,|\, \mathcal{M}) = 1 - \gamma$$

where

- $a(\epsilon) = F^{-}_{\beta_1}(1-\epsilon \,|\, \mathcal{M})$ and $F^{-}_{\beta_1}(\cdot \,|\, \mathcal{M})$ is the quantile function of the posterior distribution of β_1,
- $b(\epsilon) = F^{-}_{\alpha_2}(\frac{\gamma-\epsilon}{1-\epsilon} \,|\, \beta_1 < a(\epsilon), \mathcal{M})$ and $F^{-}_{\alpha_2}(\cdot \,|\, \beta_1 < a(\epsilon), \mathcal{M})$ is the quantile function of the conditional distribution of α_2 given $\beta_1 < a(\epsilon)$ and $\mathcal{M}$.

Indeed

$$\begin{aligned} P(\beta_1 \le a(\epsilon)\,,\, b(\epsilon) \le \alpha_2 \,|\, \mathcal{M}) &= P(\beta_1 \le a(\epsilon) \,|\, \mathcal{M}) P(b(\epsilon) \le \alpha_2 \,|\, \beta_1 \le a(\epsilon)\,,\, \mathcal{M}) \\ &= (1-\epsilon)(1 - \frac{\gamma-\epsilon}{1-\epsilon}) = 1 - \gamma \end{aligned}$$

Testing procedure : If the set $E = \{\epsilon \,|\, a(\epsilon) < b(\epsilon)\}$ is empty, then we conclude that, with probability $1-\gamma$, no gap exists between these successive phases. Otherwise, the gap interval is the longest interval $[a(\epsilon^*), b(\epsilon^*)]$ where

$$\epsilon^* = \text{argmax}_{\epsilon \in E}\,(b(\epsilon) - a(\epsilon))$$

2.4. Tempo Plot: Rhythms of occurrence of events

Dye (2016) proposes a graphical tool, called **tempo plot**, to evaluate the rhythms of occurrence of events. We propose a Bayesian interpretation of the estimate. The quantity of interest cannot be viewed as a counting process because the date of the events $(\tau_1, \ldots, \tau_n)$ are not observed. They are estimated from a Bayesian chronological model (e.g Oxcal, ChronoModel). For each date t, the aim is to estimate the number of events $N(t)$ which occurs before the date t, we have

$$N(t) = \sum_{i=1}^{n} \mathbb{I}_{]-\infty,t]}(\tau_i)$$

As $N(t)$ is a function of the parameters $(\tau_1, \ldots, \tau_n)$, we can easily estimate $N(t)$ from the joint posterior distribution of $(\tau_1, \ldots, \tau_n)$. The Bayes estimate of $N(t)$ (under quadratic loss) is the posterior mean of $N(t)$ i.e.,

$$\begin{aligned} \hat{N}(t) &= \mathbb{E}(N(t) \,|\, \mathcal{M}) \\ &= \mathbb{E}(\sum_{i=1}^{n} \mathbb{I}_{]-\infty,t]}(\tau_i) \,|\, \mathcal{M}) \\ &= \sum_{i=1}^{n} P(\tau_i < t \,|\, \mathcal{M}). \end{aligned} \tag{5}$$

Using the output of the MCMC algorithm $(\tau_1^{(k)}, \ldots, \tau_n^{(k)})_{k=1,\ldots,N_{MCMC}}$, we can approximate the Bayes estimate $\hat{N}(t)$ by taking

$$\begin{aligned} &= \sum_{i=1}^{n} \frac{1}{N_{MCMC}} \sum_{k=1}^{N_{MCMC}} \mathbb{I}_{]-\infty,t]}(\tau_i^{(k)}) \\ &= \frac{n}{N_{MCMC}} \sum_{k=1}^{N_{MCMC}} F^{(k)}(t) \end{aligned}$$

where $F^{(k)}$ is the empirical cumulative distribution of the sample $(\tau_1^{(k)}, \ldots, \tau_n^{(k)})$ (output of the iteration number of k of MCMC algorithms)

$$F^{(k)}(t) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}_{]-\infty,t]}(\tau_i^{(k)})$$

The N_{MCMC} functions $nF^{(k)}(.)$ y provide a sample from the posterior distribution of N. Therefore we can easily build a credible confidence region for the function N. Indeed for each t we take the smaller posterior interval approximated from the sample $nF^{(k)}(t), k = 1, \ldots, N_{MCMC}$. An alternative is to use Gaussian approximation to get the confidence intervals.

Alternative to Tempo plot consist in representing the activity, that is the first derivative of $\hat{N}$ defined in (5).

3. Bayesian modeling of the Ksar Akil's stratigraphy

We present an illustrative example of the use of the **ArchaeoPhases** package. The aim is to estimate phases or groups of dates using the MCMC output of the Bayesian modeling.

We estimate the archaeological chronology by two Bayesian models implemented in Oxcal and ChronoModel. Note that these softwares do not allow to choose the same Bayesian model. Indeed, the concept of "Event model", that allows to combine dates, is only implemented in ChronoModel (see Lanos and Philippe (2017)). We do not intend to discuss the choice of modeling nor to compare the results obtained by the modeling done with Oxcal and the one with ChronoModel.

As both softwares are not connected with R, The modeling must first be carried out using one of both softwares, in order to extract the MCMC samples of the joint posterior distribution. Hereafter, we explain how one should do to export the MCMC output required for the use of the **ArchaeoPhases** package.

3.1. The Ksar Akil's Paleolithic stratigraphy

At Ksar Akil (Lebanon), a deep Paleolithic stratigraphic sequence was investigated in order to established the chronology of the site (see Bosch *et al.* (2015)). This stratigraphic sequence included Initial Upper Paleolithic (IUP), from layer XXV (the bottom of the sequence) to layer XXI, Ahmarian, from layer XX to layer XIV, Upper Paleolithic (UP), from layer XV to layer VI, and Epi-Paleolithic (EPI), from layer V to layer I. In addition, 16 shell ornaments were found throughout the stratigraphy and dated by AMS radiocarbon technique.

The aim of this modeling was to establish the chronology of the succession of these Paleolithic phases: IUP, Ahmarian, UP and EPI using these 16 radiocarbon dates and the prior information coming from the stratigraphy.

3.2. Modeling with Oxcal

The authors used an original modeling with Oxcal. The description of the model is given in Bosch *et al.* (2015) and the script is available in the Supporting Information material. See Bosch *et al.* (2015) for the modeling details.

Extracting the MCMC sample of the joint distribution

Once the modeling is done with Oxcal, the MCMC_Sample command should be added to the script.

```
MCMC_Sample([Name], [Interval], [Max]) {...} ) ;
```

This command allows all of the MCMC samples to be saved to a file at defined intervals and with a maximum number of samples taken (default 1000); the file can be found as a document. This file only includes those samples generated within the run analysis.

3.3. Modeling with Chronomodel

Chronomodel has a user-friendly graphical interface easily manipulated in order to define the hierarchical model developed in Lanos and Philippe (2017).

From the information given by Bosch *et al.* (2015), we propose a Bayesian model for the Ksar Akil's stratigraphy implemented in ChronoModel (version 1.5). In this model, the study period is set from -50 000 to -25 000. A target event is the death of a shell. Each event is associated with only one radiocarbon date. Events are named according to the layer in which the shells were found (for instance, as they were four shells found in layer XVI, the four associated events were named "Layer XVI.1", ... "Layer XVI.4"). Stratigraphic constraints are also included in the model, they are symbolised by the black arrows in Figure 1a.
These 16 events are gathered in Paleolithic phases according to the layer in which the shells

were found (see Figure 1b.).

For this modeling, 3 chains are run in parallel. For each chain, 1 000 iterations are used during the Burn-in period, 20 batches of 500 iterations are used in the Adapt period, 100 000 iterations are drawn in the Acquire period by only 1 out of 10 are kept in order to break the correlation structure.

Extracting the MCMC sample of the joint distribution

Now, the Markov Chains simulated by ChronoModel have to be extracted and then imported into R in order to use the **ArchaeoPhases** package.
Once your modeling is done with ChronoModel and once the RUN command is launched, you may save all MCMC samples in format BC/AD.

To do that, use CSV Results on the right hand side of the window in the Results tab (see Figure 2). Several CSV files are created: a file called ”events.csv” containing the MCMC samples of all events, a file called ”phases.csv” containing the date of the minimum (alpha) and the date of the maximum (beta) of all phases if at least one phase is modeled, and a file per phase (if any) containing the MCMC samples of the parameters of the phase (the minimum and the maximum date) and all events included in it.
All thoses files may be imported into R in order to use the **ArchaeoPhases** package.

3.4. Needs for ArchaeoPhases

To understand chronologies, one can gather dates with common criteria into groups or phases. These groups are characterised by the date of the minimum of the group and the date of the maximum of the group (see Figure 1b. for our example).
For chronologies estimated with ChronoModel, the minimum (called alpha) and the maximum (called beta) of each group of dates or each phase are computed and saved in the file called ”phases.csv”.
For chronologies constructed with Oxcal, as the function MCMC_Sample() does not allow to save the MCMC samples of the minimum and the maximum (functions First() and Last()), we need to create a file containing these values. To do that, we can use the function `CreateMinMaxGroup()` of the **ArchaeoPhases** package.

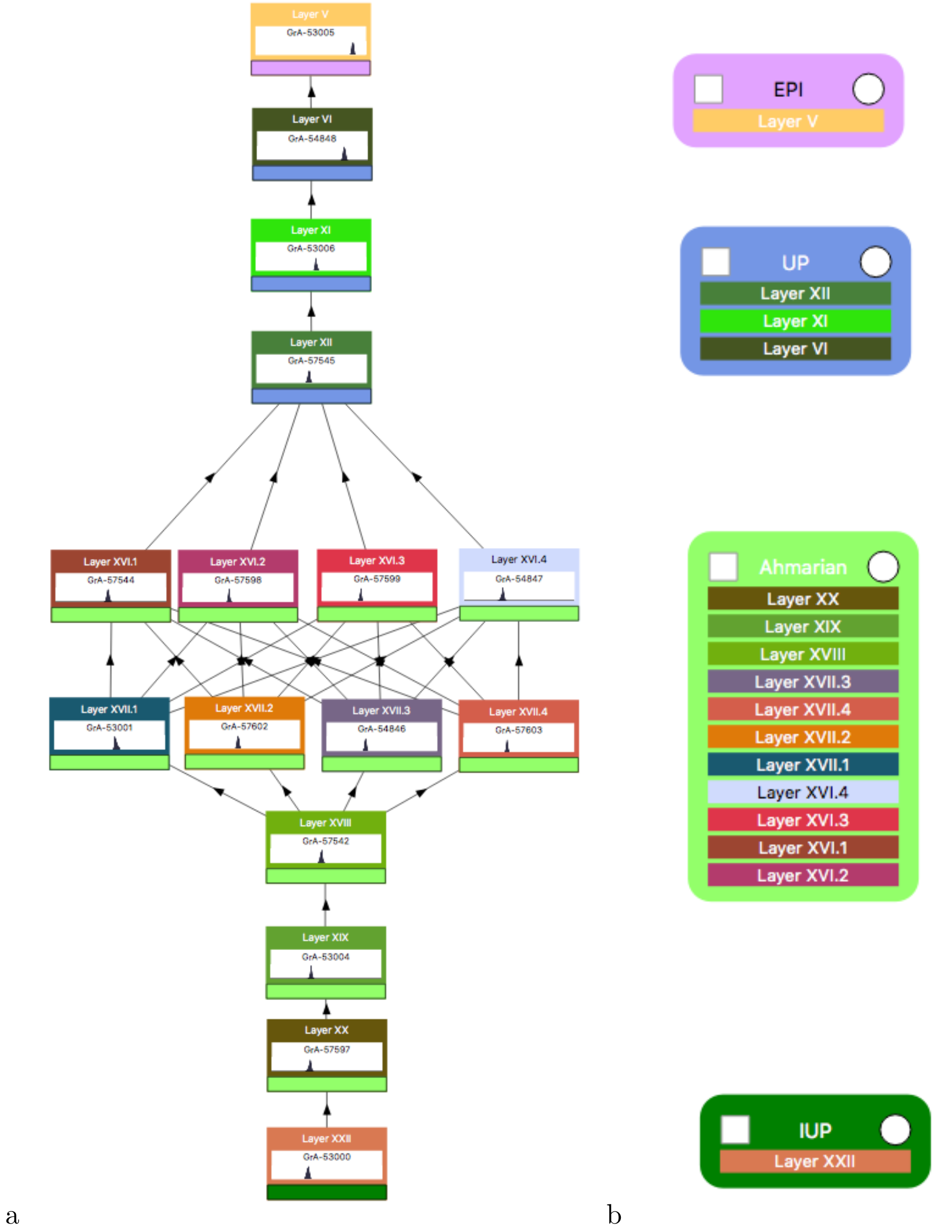
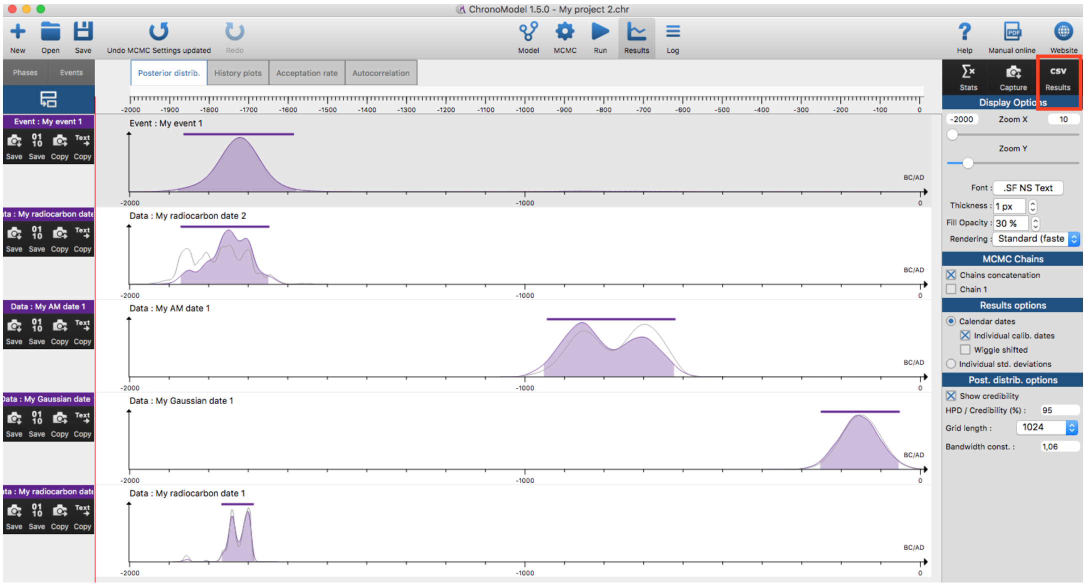


Figure 1: ChronoModel interface showing the modeling step. (a) Rectangular nodes represents the events of interest associated with one radiocarbone date. The color of the bottom of each node corresponds to the color of the phase in which the event is gathered. An arrow between two events symbolises a time order relationship. (b) Groups of events representing the four Paleolithic phases.



-2000 Zoom X 10

Zoom Y

Font : .SF NS Text

Thickness : 1 px

Fill Opacity : 30 %

Rendering : Standard (faste

☒ Chains concatenation

□ Chain 1

● Calendar dates

☒ Individual calib. dates

□ Wiggle shifted

○ Individual std. deviations

☒ Show credibility

HPD / Credibility (%) : 95

Grid length : 1024

Bandwidth const. : 1,06

Figure 2: ChronoModel interface showing the results of a modeling.

4. First steps with ArchaeoPhases

4.1. Installing the package ArchaeoPhases

When R is launched, you need to install (only the first time) the package **ArchaeoPhases** using the following code :

```
R> install.packages('ArchaeoPhases', dependencies = TRUE)
```

Then the first step is to load the library:

```
R> library("ArchaeoPhases")
```

4.2. Importing data into R software

To import the data file into R, you may use `ImportCSV()` function. For CSV files extracted from ChronoModel sotfware, there is no need to specify any other parameters than the name of the file (and the path to it). Otherwise, you may change the specification after "sep=" and "dec=". The parameter "comment.char=" is used to define how comments are written in the file to be imported. Comments of all csv files generated by ChronoModel are specified by '#'.

4.3. Diagnostic tools

The output of any Bayesian modeling is simulated chains, the convergence of the MCMC samples should always be checked before any other analysis (for a more detail on the diagnostic of Markov chain, see Robert and Casella (2009)). ChronoModel has its own diagnostic tools, however it is not the case for Oxcal.

In order to use any function from package **coda**, data as to be transformed in a `mcmc.list` using the function `coda.mcmc()`. This function as two parameters: the data and the number of parallel chains. The dataset `KADatesChronoModel` contains the MCMC samples of the 16 dates of Ksar Akil's site (see Section 3.1). Three Markov chains have been generated by ChronoModel application.

```
R> data(KADatesChronoModel)
R> Dates_mcmc = coda.mcmc(data = KADatesChronoModel, numberChains = 3)
```

Now, we can trace the history plot of each parameter and see whether the chains have reach equilibrium before the Acquire period. The column 1 containing the iteration number is withdrawn.

```
R> plot(Dates_mcmc[,-1])
```

A part of the results is given in Figure 3. The 3 chains seem to have properly reach their equilibrium. The Gelman-Rubin diagnostic can also be tested :

```
R> gelman.diag(Dates_mcmc[,-1])
```

```
> gelman.diag(Dates_mcmc[,-1])
Potential scale reduction factors:

             Point est. Upper C.I.
Layer.V             1.0        1.0
Layer.XVI           1.0        1.0
Layer.VI            1.0        1.0
Layer.XVI.1         1.0        1.0
Layer.XI            1.0        1.0
Layer.XVI.2         1.0        1.0
Layer.XVI.3         1.0        1.0
Layer.XII           1.0        1.0
Layer.XVII          1.0        1.0
Layer.XVIII         1.0        1.0
Layer.XVII.1        1.0        1.0
Layer.XIX           1.0        1.0
Layer.XVII.2        1.0        1.0
Layer.XX            1.0        1.0
Layer.XVII.3        1.0        1.0
Layer.XXII          1.0        1.0

Multivariate psrf

1
```

The values of the Gelman-Rubin criterion close to 1 assess the convergence of the Markov chains.

4.4. Examining a series of dates

The data frame called "KADatesChronoModel.RData" contains the MCMC samples of each of the 16 dates corresponding to the death of the 16 shells found at Ksar Akil. The name given to thoses dates are the name of the layer in which they where found. In the following, the iterations of the three parallel chains are used as if there were only chain.
Using the function `MultiCredibleInterval`, we can estimate the 95% credible interval of each date.

```
R> MultiCredibleInterval(KADatesChronoModel, position = c(2:17), level = 0.95)
```

which gives the following results :

```
> MultiCredibleInterval(KADatesChronoModel, position = c(2:17), level = 0.95)
             Level CredibleIntervalInf CredibleIntervalSup
Layer.V       0.95              -29014              -27017
```

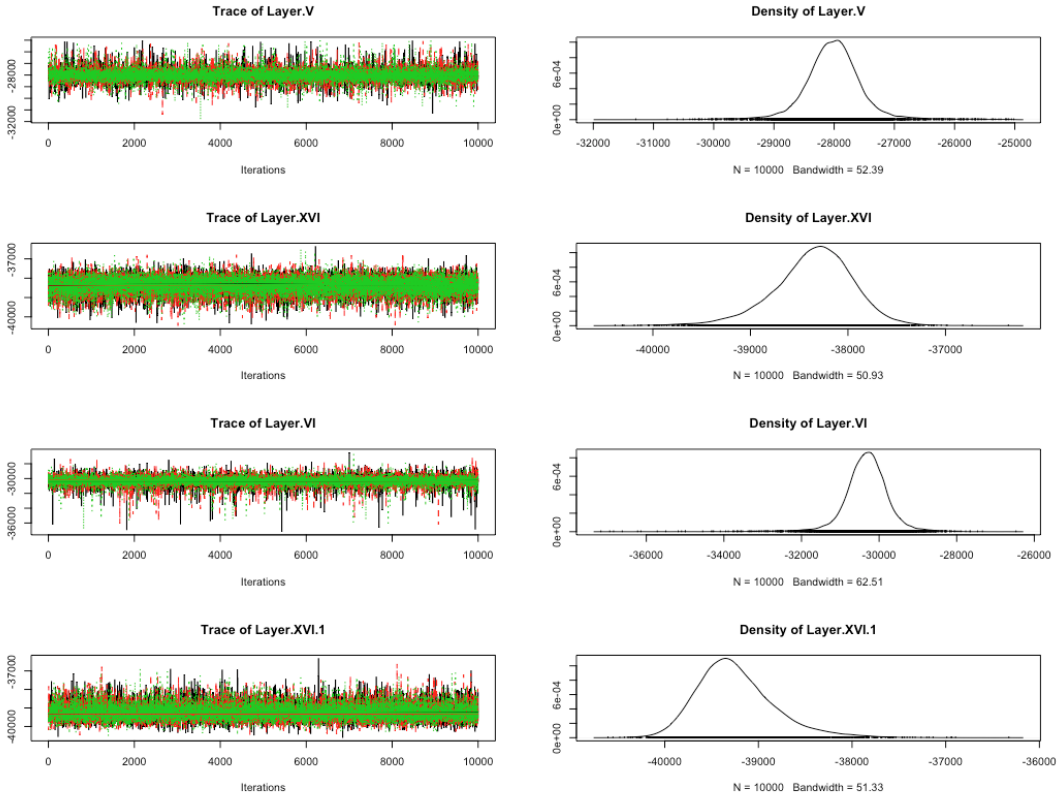


Figure 3: Traces and density of several parameters of the Bayesian modeling.

```
Layer.XVI     0.95            -39237              -37593
Layer.VI      0.95            -31486              -29168
Layer.XVI.1   0.95            -40028              -38384
Layer.XI      0.95            -37412              -35038
Layer.XVI.2   0.95            -40335              -37813
Layer.XVI.3   0.95            -40362              -37846
Layer.XII     0.95            -38524              -36763
Layer.XVII    0.95            -41017              -39083
Layer.XVIII   0.95            -41406              -40102
Layer.XVII.1  0.95            -40832              -39001
Layer.XIX     0.95            -41606              -40498
Layer.XVII.2  0.95            -40773              -39596
Layer.XX      0.95            -42136              -40864
Layer.XVII.3  0.95            -41229              -39592
Layer.XXII    0.95            -43297              -41161
```

The function `MultiHPD` gives an estimation of the 95% HPD regions of each date (results not shown).

```
R> MultiHPD(KADatesChronoModel, position = c(2:17), level = 0.95)
```

All those intervals may be drawn on a graph using the function `MultiDatesPlot`. The following lines show an example of its use, Figure 4 presents the graph of the credible intervals and Figure 5 displays the graph of a series of HPD regions.

```
R> MultiDatesPlot(KADatesChronoModel, position = c(2:17), level = 0.95,
intervals="CI", title=" 95% CI of Ksar Akil dates")
R> MultiDatesPlot(KADatesChronoModel, position = c(2:17), level = 0.95,
intervals="HPD", title=" 95% HPD regions of Ksar Akil dates")
```

Now, the rhythm of occurrence of the dates may be investigated using two functions `TempoPlot` and `TempoActivityPlot`.

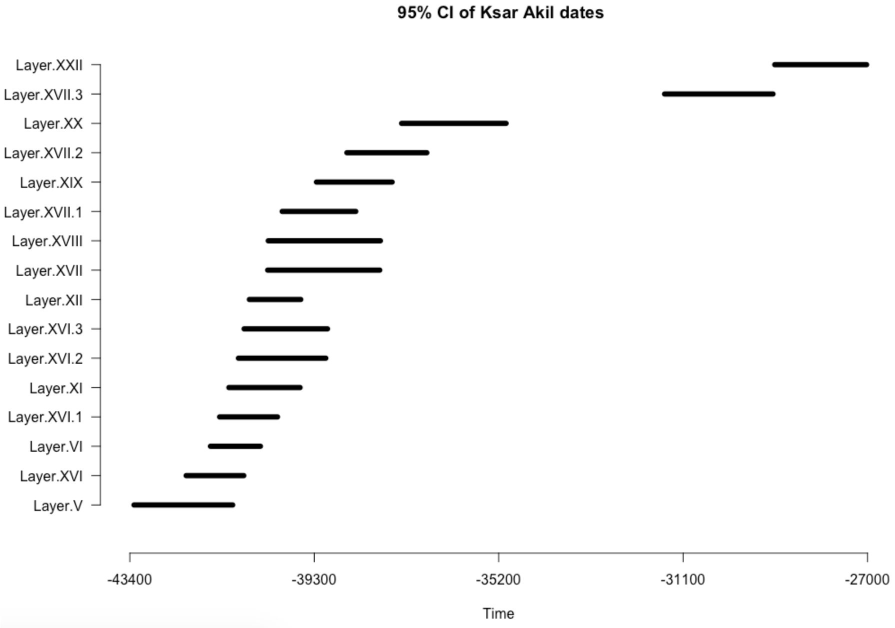


Figure 4: Graph of the credible interval of the series of dates included in Ksar Akil

```
R> TempoPlot(KADatesChronoModel, position = c(2:17), level = 0.95, title=" Tempo
plot")
R> TempoActivityPlot(KADatesChronoModel, position = c(2:17), level = 0.95)
```

Figure 6 displays the tempo plot and the activity plot of the site of Ksar Akil.

From these graphs, we can see that the highest part of the sampled activity is dated between -45 000 to -35 000 but two dates are younger, at about -32 000 and -28 000.

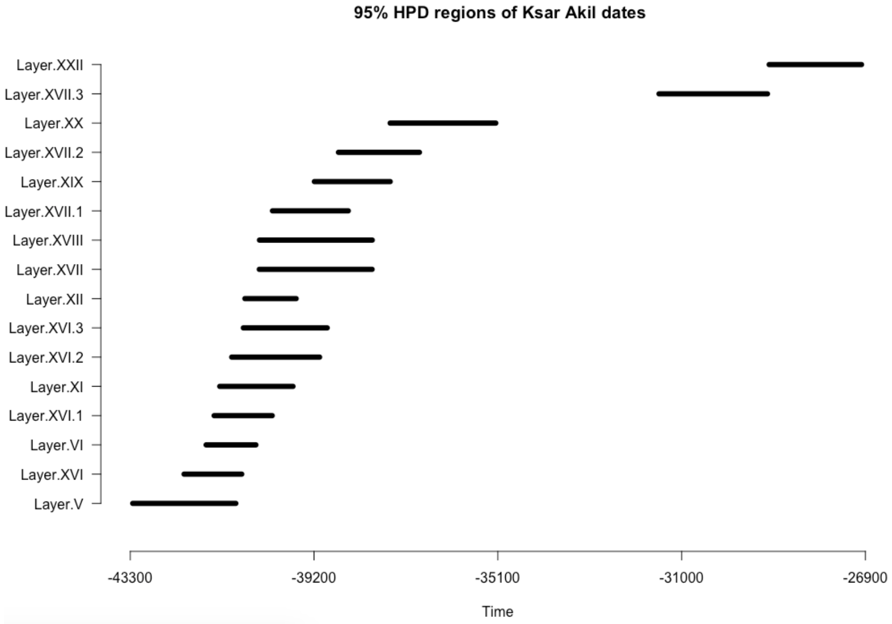


Figure 5: Graph of the HPD regions of the series of dates included in Ksar Akil

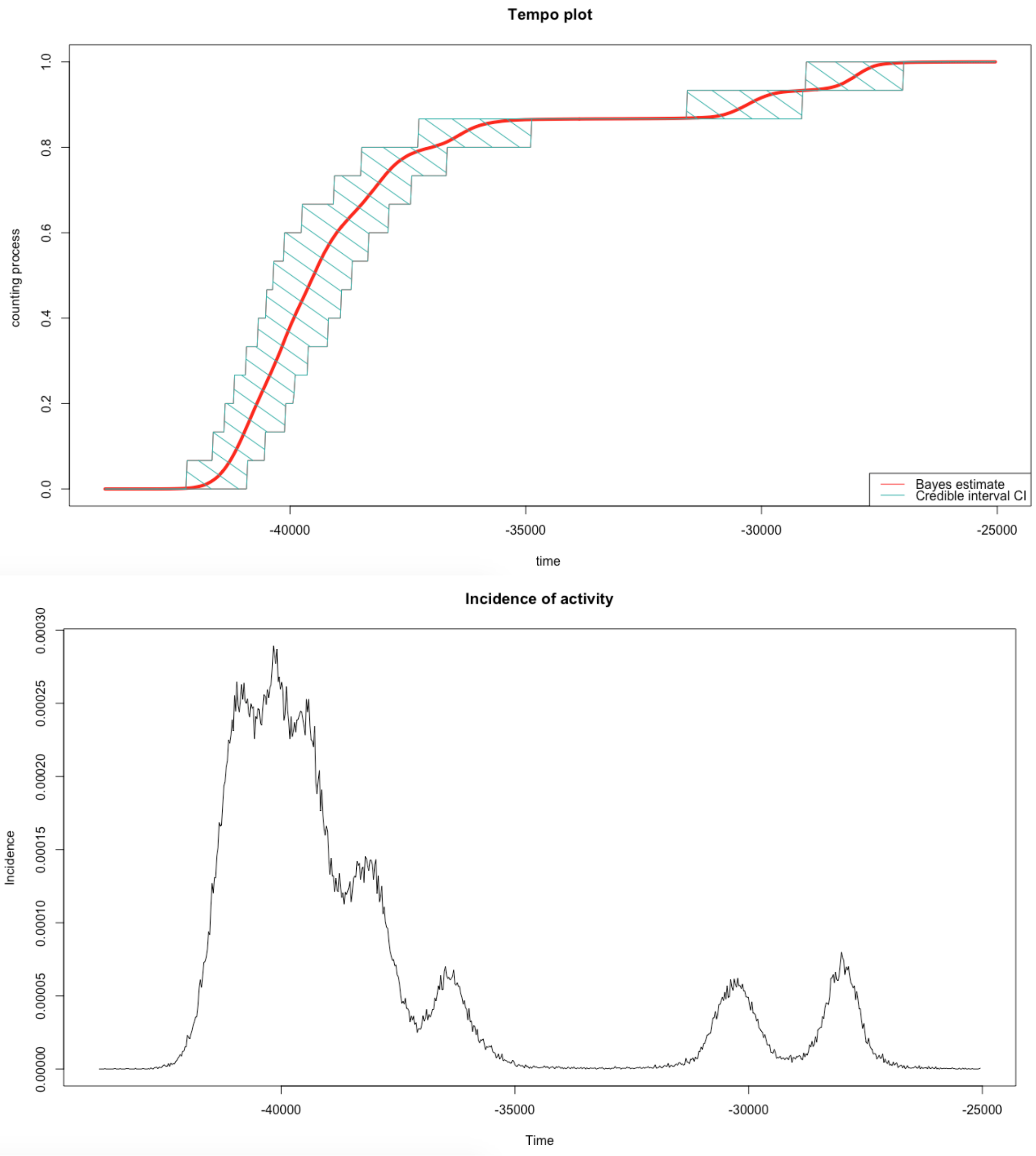


Figure 6: Tempo plot [top] and activity plot [bottom] of Ksar Akil

5. Analysis of archaeological phases

As said in Section 2, a phase or a group of dates is defined by the date of the minimum and the date of the maximum of the group. In this part, we will use the data containing these values for each group of dates.

```
R> data(KAPhasesChronoModel)
R> attach(KAPhasesChronoModel)
```

5.1. Examining an archaeological phase

From the output of the MCMC algorithm, we can estimate the duration of the groups as the value of the maximum - minimum at each iteration. We can also estimate the phase time range as the shortest interval that contains all the dates of the group at a given confidence level (see Section 2 for the statistical details).

If interested in the summary statistics of the characteristics of a phase / group of dates, use the function `PhaseStatistics()`. For instance, let's examine the summary statistics of the phase Ahmarian.

```
R> PhaseStatistics(Ahmarian.alpha, Ahmarian.beta, level = 0.95)
```

The output of the **R** console is then :

```
> PhaseStatistics(Ahmarian.alpha, Ahmarian.beta, level = 0.95)
                        Minimum   Maximum Duration
mean                  -41463.00 -38249.00  3214.00
MAP                   -41441.00 -38250.00  3201.00
sd                       334.00    403.00   517.00
Q1                    -41659.00 -38496.00  2875.00
median                -41437.00 -38249.00  3202.00
Q2                    -41238.00 -38001.00  3535.00
level                      0.95      0.95     0.95
CredibleInterval Inf -42130.00 -39076.00  2200.00
CredibleInterval Sup -40851.00 -37456.00  4242.00
HPDRInf 1             -42092.00 -39063.00  2183.00
HPRDSup 1             -40801.00 -37437.00  4227.00
```

`PhaseStatistics()` results in a matrix of several summary statistics according to the marginal posterior density of the minimum, the maximum and the duration of the phase Ahmarian. The default confidence level is 0.95.

The following code gives the endpoints of the time range of the phase Ahmarian (the group of dates that constitute the phase Ahmarian) and recall the given confidence level.

```
R> PhaseTimeRange(Ahmarian.alpha, Ahmarian.beta, level = 0.95)
```

The output of the console is

```
> PhaseTimeRange(Ahmarian.alpha, Ahmarian.beta, level = 0.95)
       level TimeRangeInf TimeRangeSup
        0.95    -42189.00    -37461.00
```

Function PhasePlot may be used to draw a plot of the marginal posterior density of the minimum and the maximum of a phase on a same graph.

```
R> PhasePlot(Ahmarian.alpha, Ahmarian.beta, level = 0.95,
title ="Characterisation of phase Ahmarian")
```

The result is shown on Figure 7. Marginal posterior densities of the minimum and the maximum of the phase Ahmarian (curves) are presented with the shortest credible interval (solid line under the curve) at the desired level (default = 95 %) and their mean value (a dot under the curve) using the same color. In addition, the time interval of the phase at the desired level (default = 95 %) is also presented by a solid line above the curves.

PhaseDurationPlot draws the marginal posterior density of the duration of the phase with its shortest credible interval and its mean value. By default, the confidence level is fixed at 0.95 and graphs are in color but these details may be changed. Figure 8 displays the result.

```
R> PhaseDurationPlot(Ahmarian.alpha, Ahmarian.beta, level = 0.95,
title ="Duration of Phase Ahmarian")
```

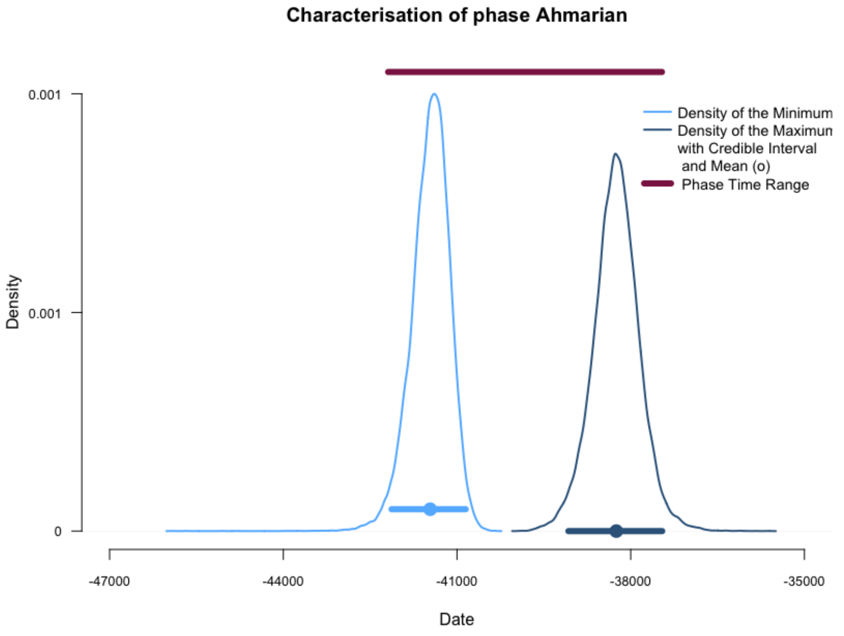


Figure 7: Plot of the characteristics of phase Ahmarian

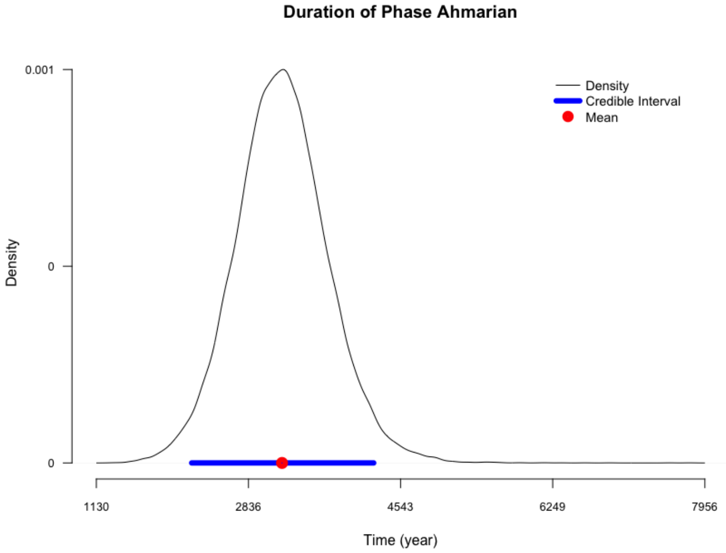


Figure 8: Marginal posterior density of the duration of phase Ahmarian

5.2. Examining the succession of two phases

Let's use for this example the succession of phases IUP and Ahmarian. As these phases are in temporal order, we can estimate, if it exists, the gap (or hiatus) and the transition interval between both phases (see Section 2 for more details). To test if there exists a gap at 95% between these successive phases, let's use the function PhasesGap.

```
R > PhasesGap(IUP.beta, Ahmarian.alpha, level = 0.95)
```

Then the output is

```
> PhasesGap(IUP.beta, Ahmarian.alpha, level = 0.95)
            level HiatusIntervalInf HiatusIntervalSup
           "0.95"               "NA"               "NA"
```

This code gives the endpoints of the gap interval between both phases. The default confidence level is 0.95. For this confidence level, no hiatus exists between phases IUP and Ahmarian. However, for a confidence level fixed at 0.50, there exists a gap of almost 100 years between both phases. To test if there exists a gap at 50% between these successive phases, let's use the following line:

```
R> PhasesGap(IUP.beta, Ahmarian.alpha, level = 0.50)
```

Now the output is

```
> PhasesGap(IUP.beta, Ahmarian.alpha, level = 0.50)
            level HiatusIntervalInf HiatusIntervalSup
```

```
              0.5          -41759.0           -41666.0
```

Now, to estimate the transition interval at 95% between these successive phases, let's use the function `PhasesTransition()`:

```
R> PhasesTransition(IUP.beta, Ahmarian.alpha, level = 0.95)
```

And the result is:

```
> PhasesTransition(IUP.beta, Ahmarian.alpha, level = 0.95)
              0.95 TransitionRangeInf TransitionRangeSup
              0.95          -43241.00          -40728.00
```

At 95%, the transition between these two phases starts at -43 241 and ends at -40 728.
All these pieces of information may be illustrated on a graphic using the function `SuccessionPlot()`. Figure 9 presents the resulting graph.

```
R> SuccessionPlot(IUP.alpha, IUP.beta, Ahmarian.alpha, Ahmarian.beta,
R> level = 0.95, title="Characterisation of a succession of phases")
```

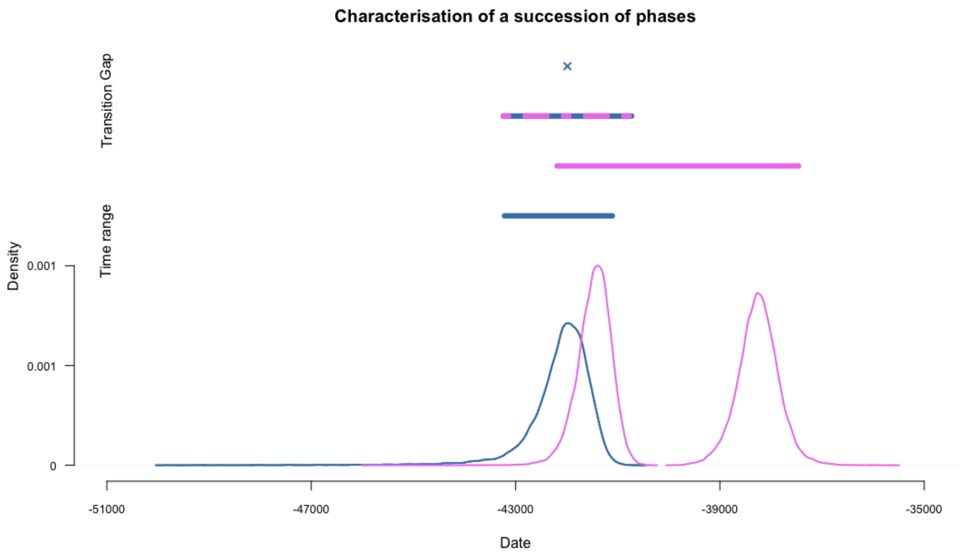


Figure 9: Plot of the succession of phases IUP and Ahmarian and 95% intervals. Curves represent the densities of the minimum and the maximum of each phase. The oldest phase is drawn in blue, the youngest phase in violet. When a group is defined by only one date, then the minimum date equals the maximum date of the group. Hence, in that case, only one curve is presented. Segments correspond to time range of the phase of the same color, two-coloured segments correspond to transition interval or to the gap range. A cross instead of a two-coloured segment means that there is no gap range at the desired level of confidence.

5.3. Examining a series of phases

We may also be interested in a series of phases, as for instance the following phases of Ksar Akil: IUP, Ahmarian, UP and EPI. We could, for example, wish to know the time range of these different phases, or we could want to draw the different minimums and maximums on a same graph. We could also wish to know the credible interval of all minimums.

To do that, the ”Multi” functions are available. For data extracted from ChronoModel software or using the function `CreateMinMaxGroup()`, only the vector of positions of all phases' minimums are needed, as the maximums are the column next to the minimums'. By default, the argument of `position_maximum` is equals to `position_minimum + 1`. Otherwise, the vector of positions of the phases' maximums should be specified.
For example, let's estimate the time range of phases IUP, Ahmarian, UP and EPI, whose minimums are respectively in position 2, 4, 6 and 8.

```
R> MultiPhaseTimeRange(KAPhasesChronoModel, position_minimum = c(2,4,6,8))
```

```
> MultiPhaseTimeRange(KAPhasesChronoModel, position_minimum = c(2,4,6,8))
                             Level TimeRangeInf TimeRangeSup
EPI.alpha EPI.beta            0.95       -29071       -27102
UP.alpha UP.beta              0.95       -38559       -29335
Ahmarian.alpha Ahmarian.beta  0.95       -42189       -37461
IUP.alpha IUP.beta            0.95       -43217       -41106
```

Phase ”EPI” whose minimum output is in column 2 has a time range = [-29 071, -27 102] at a confidence level of 95% and the phase ”IUP” whose minimum output is in column 8 has a time range = [-43 217, -41 106].
The function `MultiPhasePlot()` generates a plot presented in Figure 10 that illustrates the characteristics of these phases.

```
R> MultiPhasePlot(KAPhasesChronoModel, position_minimum = c(2,4,6,8),
title="Phases EPI, UP, Ahmarian et IUP")
```

Curves represent the densities of the minimum or the maximum of each phase. Segments above the curves correspond to the time range of the phases. Characteristics of a phase (minimum and maximum densities, time range) are drawn using the same color. When a group is defined by only one date, then the minimum date equals the maximum date of the group. Hence, in that case, only one curve is presented.

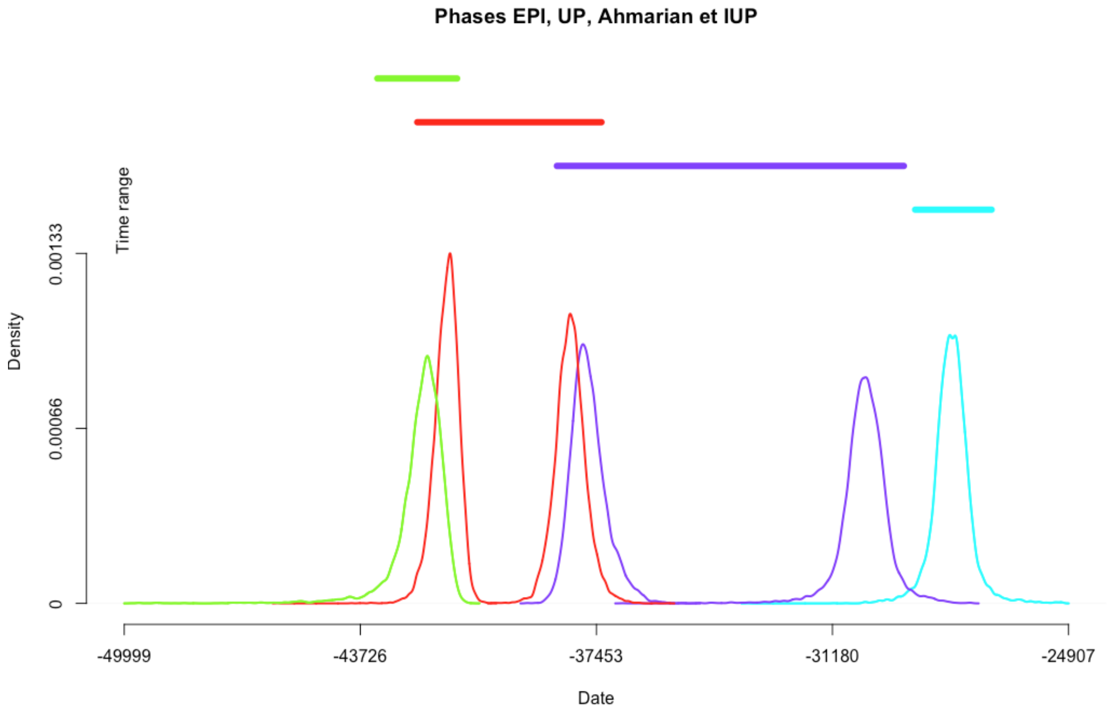


Figure 10: Plot of the series of Ksar Akil's phases and their associated time range at 95%. The marginal posterior densities of phase IUP are drawn in green, those of phase Ahmarian are in red, those of phase UP are in purple and those of phase EPI are in light blue. As there is only one date in the phases EPI and IUP, the minimum and the maximum of these phases have the same value at each iteration. Hence, we can see only one curve for each of these phases. Time ranges are displayed by segments above the curves of the same color as the densities of the phase.

5.4. Examining a succession of phases in temporal order

We may also be interested in a succession of phases. This is actually the case of the succession of IUP, Ahmarian, UP and EPI that are in stratigraphic order. Hence, we can estimate the transition interval and, if it exists, the gap between these successive phases. This may be done using the following code :

```
R> MultiPhasesGap(KAPhasesChronoModel, position_minimum=c(8,6,4,2))
R> MultiPhasesTransition(KAPhasesChronoModel, position_minimum=c(8,6,4,2))
```

For these functions, the order of the phases is important. The vector of positions of the minimums should start with the minimum of the oldest phase and end with the one of the youngest phase. For data extracted from ChronoModel or using the function `CreateMinMaxGroup()`, the vector of positions of the phases' maximums is deduced from the vector of the minimum. For other data, this vector should be specified.

```
> MultiPhasesGap(KAPhasesChronoModel, position_minimum=c(8,6,4,2))
                          Level  HiatusIntervalInf HiatusIntervalSup
IUP.beta & Ahmarian.alpha "0.95" "NA"              "NA"
Ahmarian.beta & UP.alpha  "0.95" "NA"              "NA"
UP.beta & EPI.alpha       "0.95" "-29180"          "-28977"
> MultiPhasesTransition(KAPhasesChronoModel, position_minimum=c(8,6,4,2))
                          0.95 TransitionRangeInf TransitionRangeSup
IUP.beta & Ahmarian.alpha 0.95             -43241             -40728
Ahmarian.beta & UP.alpha  0.95             -39075             -36686
UP.beta & EPI.alpha       0.95             -31504             -26960
```

At a confidence level of 95%, there is no gap between the succession of phases IUP, Ahmarian and UP, but there exists one of 203 years between phase UP and phase EPI. The function `MultiSuccessionPlot()` draws the characteristics of such a succession. Figure 11 presents the resulting plot.

```
R> MultiSuccessionPlot(Phases, position_minimum =c(8,6,4,2), title = "Ksar
Akil - Succession of phases:  IUP, Ahmarian, UP, EPI ")
```

In that plot, curves represent the densities of the minimum or the maximum of each phase. When a group is defined by one only date, the minimum equals the maximum date of the group. Hence, in that case, only one curve is presented. Segments above the curves correspond to time range of the phase associated to a level confidence of 95%. Two-coloured segments correspond to transition interval or to the gap range associated to a level confidence of 95%. A cross instead of a two-coloured segment means that there is no gap between the succession of phases at the desired level of confidence.

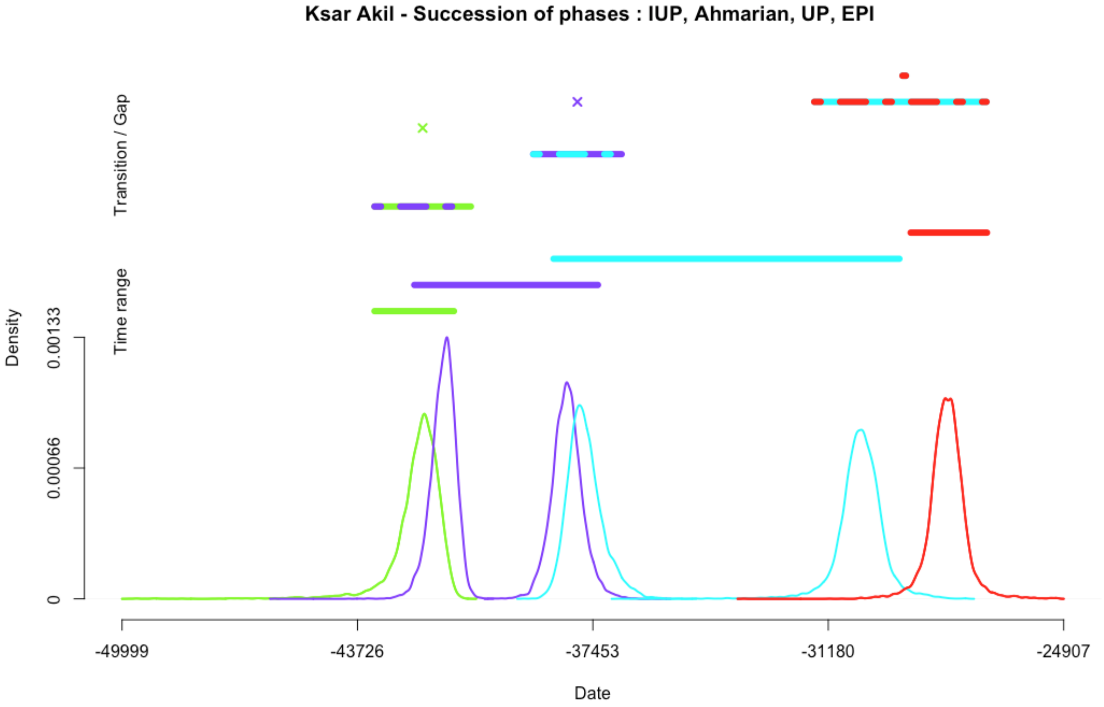


Figure 11: Plot of the succession of phases from Ksar Akil. The characteristics of phase IUP are drawn in green, those of phase Ahmarian are in purple, those of phase UP are in light blue and those of phase EPI are in red. Again, as there is only one event in the phases EPI and IUP, the minimum and the maximum of these phases have the same values at each iteration. Hence, we can only see one curve for each of these phases. Time range are displayed by segments above the curves. Two-coloured segments correspond to transition interval or to the gap range associated to a level confidence of 95%. As there are no gaps at 95% between phases IUP and Ahmarian, and Ahmarian and UP, a cross is drawn instead.

5.5. Summary

We can summary the characteristics of the different phases of Ksar Akil, using the data published by Bosch *et al.* (2015) and the modeling done with ChronoModel, by the following table :

Phase	**Time range**		**Transition range**		**Gap range**	
	Inf	**Sup**	**Inf**	**Sup**	**Inf**	**Sup**
IUP	-43 217	-41 106	NA	NA	NA	NA
Ahmarian	-42 189	-37 461	-43 241	-40 728	NA	NA
UP	-38 559	-29 335	-39 075	-36 686	NA	NA
EPI	-29 071	-27 102	-31 504	-26 960	-29 180	-28 977

Table 1: Endpoints (Inf and Sup values) are given for each interval associated with a level of confidence at 95%.

Acknowledgments

Marie-Anne Vibet thanks the DefiMaths Program and Region Pays de La Loire for financial supports. Correspondence should be sent to Anne Philipe (anne.philippe@univ-nantes.fr).

References

Bosch MD, Mannino MA, Prendergast AL, O'Connell TC, Demarchi B, Taylor SM, Niven L, van der Plicht J, Hublin JJ (2015). "New Chronology for Ksâr Akil (Lebanon) Supports Levantine Route of Modern Human Dispersal into Europe." *Proceedings of the National Academy of Sciences*, **112**(25), 7683–7688. doi:10.1073/pnas.1501529112.

Dye TS (2016). "Long-term Rhythms in the Development of Hawaiian Social Stratification." *Journal of Archaeological Science*, **71**, 1–9. doi:10.1016/j.jas.2016.05.006.

Lanos P, Philippe A (2017). "Hierarchical Bayesian Modeling for Combining Dates in Archaeological Context." *Journal de la SFdS*. To appear.

Lanos P, Philippe A, Lanos H, Dufresne P (2016). *Chronomodel : Chronological Modelling of Archaeological Data using Bayesian Statistics (Version 1.5)*. URL http://www.chronomodel.fr.

Ramsey CB (2009). "Bayesian Analysis of Radiocarbon Dates." *Radiocarbon*, **51**(1), 337–360.

Ramsey CB (2016). *Oxcal 4.2*. URL http://c14.arch.ox.ac.uk/oxcalhelp/hlp_contents.html.

Robert C, Casella G (2009). *Introducing Monte Carlo Methods with R*. Springer Science & Business Media.

Vibet MA, Philippe A, Lanos P, Dufresne P (2016). *ChronoModel V1.5 User's Manual*. URL http://www.chronomodel.fr.

Affiliation:

Anne Philippe
Université de Nantes, Laboratoire de mathé matiques Jean Leray
2, Rue de la Houssiniere, 44000 Nantes, FRANCE
E-mail: anne.philippe@univ-nantes.fr
URL: http://www.math.sciences.univ-nantes.fr/˜philippe/info.html

Marie-Anne Vibet
Université de Nantes, Laboratoire de mathé matiques Jean Leray
2, Rue de la Houssiniere, 44000 Nantes, FRANCE
E-mail: marie-anne.vibet@univ-nantes.fr