

HW3 - de Charrin

Échantillonneur de Gibbs

November 29, 2023

01/12/2023 de Charrin Théotime - MVA

1 TP3 - Échantillonnage Hasting-Metropolis et Gibbs

```
[1]: import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import sklearn
from numba import jit
import numba as nb
import scipy.stats as stats
from numba.core import types
from numba.typed import Dict
from numba.typed import List
from plotly import graph_objs as go
import time
import sys
from alive_progress import alive_bar
from numbers import Number
import copy
```

```
[2]: plt.rcParams['figure.figsize'] = [12, 8]
plt.rcParams['figure.dpi'] = 100

def progressbar(it, prefix="", size=60, out=sys.stdout):
    count = len(it)
    start = time.time()
    def show(j):
        x = int(size*j/count)
        remaining = ((time.time() - start) / j) * (count - j)

        mins, sec = divmod(remaining, 60)
        time_str = f"{int(mins):02}:{sec:05.2f}"
```

```

print(f"{prefix}[{u'...' * x}{'.' * (size-x)}] {j}/{count} Est wait_
↪{time_str}", end='\r', file=out, flush=True)

for i, item in enumerate(it):
    yield item
    show(i+1)
print("\n", flush=True, file=out)

```

2 Exercise 1: Hasting-Metropolis within Gibbs – Stochastic Approximation EM

2.1 Qestion 1

Notations : $z_{pop} = (t_0, v_0)$, $z_i = (\alpha_i, \tau_i)$, $\theta = (\bar{t}_0, \bar{v}_0, \sigma_\xi^2, \sigma_\tau^2, \sigma^2)$

La log-likelihood du modèle complet est :

$$\begin{aligned}
\log p(\{y_{ij}\}_{ij}, \mathcal{Z}, \theta) &= \log p(\{y_{ij}\}_{ij}, z_{pop}, \{z_i\}_i, \theta) \\
&= \log p(\{y_{ij}\}_{ij} \mid z_{pop}, \{z_i\}_i, \theta) \quad (1) \\
&+ \log p(z_{pop}, \{z_i\}_i \mid \theta) \quad (2) \\
&+ \log p(\theta) \quad (3)
\end{aligned}$$

Si l'on raisonne à constante près, *i.e.* on enlève tout ce qui n'est ni dépendant de y, z ou θ on peut dire que :

$$\begin{aligned}
(1) &\propto \sum_{i=1}^N \sum_{j=1}^{k_i} \left\{ -\frac{1}{2} \log \sigma^2 - \frac{(y_{ij} - d_i(t_{ij}))^2}{2\sigma^2} \right\} && \text{car } y_{i,j} \sim \mathcal{N}(d_i(t_{i,j}), \sigma^2) \\
&\propto -\frac{Nk}{2} \log \sigma^2 - \sum_{i=1}^N \sum_{j=1}^k \frac{(y_{ij} - d_i(t_{ij}))^2}{2\sigma^2} && \text{on considère } k_i = k \quad \forall i \in [[1, N]] \\
(2) &= \log p(z_{pop} \mid \theta) + \log p(\{z_i\}_i \mid \theta) && \text{par indépendance des effets fixes } (t_0, v_0) \\
&\quad \cdot && \text{et aléatoires } (\alpha_i, \tau_i) \\
&\propto -\frac{(\bar{t}_0 - t_0)^2}{2\sigma_{t_0}^2} - \frac{(\bar{v}_0 - v_0)^2}{2\sigma_{v_0}^2} && t_0 \sim \mathcal{N}(\bar{t}_0, \sigma_{t_0}^2), \quad v_0 \sim \mathcal{N}(\bar{v}_0, \sigma_{v_0}^2) \\
&\quad -\frac{N}{2} \log \sigma_\xi^2 - \frac{\sum_1^N \xi_i^2}{2\sigma_\xi^2} && \xi_i \sim \mathcal{N}(0, \sigma_\xi^2) \text{ ou } \alpha_i \sim \log \mathcal{N}(0, \sigma_\xi^2) \\
&\quad -\frac{N}{2} \log \sigma_\tau^2 - \frac{\sum_1^N \tau_i^2}{2\sigma_\tau^2} && \tau_i \sim \mathcal{N}(0, \sigma_\tau^2) \\
(3) &\propto -\frac{(\bar{v}_0 - \bar{v}_0)^2}{2s_{v_0}^2} - \frac{(\bar{t}_0 - \bar{t}_0)^2}{2s_{t_0}^2} && \bar{t}_0 \sim \mathcal{N}(\bar{t}_0, s_{t_0}^2), \quad \bar{v}_0 \sim \mathcal{N}(\bar{v}_0, s_{v_0}^2) \\
&\quad -\left(\frac{m}{2} + 1\right)(\log \sigma_\xi^2 + \log \sigma_\tau^2 + \log \sigma^2) \\
&\quad -\frac{v^2}{2\sigma_\xi^2} - \frac{v^2}{2\sigma_\tau^2} - \frac{v^2}{2\sigma^2} && \sigma_\xi^2, \sigma_\tau^2, \sigma^2 \sim W^{-1}(v, m)
\end{aligned}$$

2.1.1 Passage en modèle exponentiel

On peut alors poser :

$$\begin{aligned}
\Phi(\theta) = & \frac{\bar{t}_0^2}{2\sigma_{t_0}^2} + \frac{\bar{t}_0^2}{2s_{t_0}^2} - \frac{2\bar{t}_0\bar{t}_0}{2s_{t_0}^2} \\
& + \frac{\bar{v}_0^2}{2\sigma_{v_0}^2} + \frac{\bar{v}_0^2}{2s_{v_0}^2} - \frac{2\bar{v}_0\bar{v}_0}{2s_{v_0}^2} \\
& + \left(\frac{N}{2} + \frac{m}{2} + 1\right)(\log \sigma_\xi^2 + \log \sigma_\tau^2) \\
& + \left(\frac{Nk}{2} + \frac{m}{2} + 1\right) \times \log \sigma^2 \\
& + \frac{v^2}{2\sigma_\xi^2} + \frac{v^2}{2\sigma_\tau^2} + \frac{v^2}{2\sigma^2}
\end{aligned}$$

Et :

$$S(y, z) = \begin{pmatrix} t_0 \\ v_0 \\ \frac{1}{N} \sum_1^N \log(\alpha_i)^2 = \frac{1}{N} \sum_1^N \xi_i^2 \\ \frac{1}{N} \sum_1^N \tau_i^2 \\ \frac{1}{Nk} \sum_1^N \sum_1^k (y_{ij} - d_i(t_{ij}))^2 \end{pmatrix} \quad \Psi(\theta) = \begin{pmatrix} \frac{2\bar{t}_0}{2\sigma_{t_0}^2} \\ \frac{2\bar{v}_0}{2\sigma_{v_0}^2} \\ \frac{N}{2\sigma_\xi^2} \\ \frac{N}{2\sigma_\tau^2} \\ \frac{Nk}{2\sigma^2} \end{pmatrix}$$

Et on retrouve :

$$\begin{aligned}
-\Phi(\theta) + \langle S(y, z) \mid \Psi(\theta) \rangle &= -\frac{\bar{t}_0^2}{2\sigma_{t_0}^2} - \frac{\bar{t}_0^2}{2s_{t_0}^2} + \frac{2\bar{t}_0\bar{\bar{t}}_0}{2s_{t_0}^2} - \frac{\bar{v}_0^2}{2\sigma_{v_0}^2} - \frac{\bar{v}_0^2}{2s_{v_0}^2} + \frac{2\bar{v}_0\bar{\bar{v}}_0}{2s_{v_0}^2} \\
&\quad - \left(\frac{N}{2} + \frac{m}{2} + 1\right)(\log \sigma_\xi^2 + \log \sigma_\tau^2) - \left(\frac{Nk}{2} + \frac{m}{2} + 1\right) \times \log \sigma^2 \\
&\quad - \frac{v^2}{2\sigma_\xi^2} - \frac{v^2}{2\sigma_\tau^2} - \frac{v^2}{2\sigma^2} \\
&\quad + t_0 \times \frac{2\bar{t}_0}{2\sigma_{t_0}^2} + v_0 \times \frac{2\bar{v}_0}{2\sigma_{v_0}^2} - \sum_1^N \xi_i^2 \times \frac{1}{2\sigma_\xi^2} \\
&\quad - \left(\sum_1^N \tau_i^2\right) \times \frac{1}{2\sigma_\tau^2} - \left(\sum_1^N \sum_1^k (y_{ij} - d_i(t_{ij}))^2\right) \times \frac{1}{2\sigma^2} \\
&= \\
(2) \quad &\left\{ \begin{aligned} &-\frac{(\bar{t}_0 - t_0)^2}{2\sigma_{t_0}^2} - \frac{(\bar{v}_0 - v_0)^2}{2\sigma_{v_0}^2} \\ &-\frac{N}{2} \log \sigma_\xi^2 - \frac{\sum_1^N \xi_i}{2\sigma_\xi^2} - \frac{N}{2} \log \sigma_\tau^2 - \frac{\sum_1^N \tau_i}{2\sigma_\tau^2} \end{aligned} \right. \\
&+ \\
(1) \quad &\left\{ \begin{aligned} &-\frac{Nk}{2} \log \sigma^2 - \frac{\sum_1^N \sum_1^k (y_{ij} - d_i(t_{ij}))^2}{2\sigma^2} \end{aligned} \right. \\
&+ \\
(3) \quad &\left\{ \begin{aligned} &-\frac{(\bar{v}_0 - \bar{\bar{v}}_0)^2}{2s_{v_0}^2} - \frac{(\bar{t}_0 - \bar{\bar{t}}_0)^2}{2s_{t_0}^2} \\ &-(\frac{m}{2} + 1)(\log \sigma_\xi^2 + \log \sigma_\tau^2 + \log \sigma^2) \\ &-\frac{v^2}{2\sigma_\xi^2} - \frac{v^2}{2\sigma_\tau^2} - \frac{v^2}{2\sigma^2} \end{aligned} \right. \\
\text{Fixé, indépendant de } \theta &\left\{ \begin{aligned} &+\frac{t_0^2}{2\sigma_{t_0}^2} + \frac{v_0^2}{2\sigma_{v_0}^2} + \frac{\bar{t}_0^2}{2s_{t_0}^2} + \frac{\bar{v}_0^2}{2s_{v_0}^2} \end{aligned} \right. \\
&\propto \log p(y, z, \theta) \text{ à des constantes près indépendantes de } \theta
\end{aligned}$$

Le modèle présenté ici appartient bien à la famille exponentielle courbe.

2.2 Question 2

[3]: #D'après l'énoncé

```

PRIOR={
  "v0_dbar":1,
  "s_v0":0.1**2,
  "t0_dbar":1,
  "s_t0":0.1**2,
  #m contrôle la variance des Wishart
  "m":5,
  "v":np.pi

```

```

}

#Paramètres fixés. On considère que sigma_x représente sigma^2_x
FIXED={
    "p0_bar":20,
    "sigma_p0":1**2,
    "sigma_v0":0.1**2,
    "sigma_t0":0.1**2,
}
FIXED["p0"]=np.random.normal(loc=FIXED['p0_bar'],scale=np.
    ↳sqrt(FIXED['sigma_p0']))
N=100

#À partir de la Q2 on considère que tous les k_i sont égaux entre individus et
    ↳égaux à 20.
#De même, tous les timepoints sont communs entre individus et répartis
    ↳uniformément sur [1,K]
k=20

##On génère notre ensemble de paramètres theta
THETA_TRUE={
    "t0_bar":np.random.normal(PRIOR["t0_dbar"],scale=np.sqrt(PRIOR['s_t0'])),
    "v0_bar":np.random.normal(PRIOR["v0_dbar"],scale=np.sqrt(PRIOR['s_v0'])),
    "sigma_xi":stats.invwishart.rvs(df=PRIOR['v'],scale=PRIOR['m'],size=1),
    "sigma_tau":stats.invwishart.rvs(df=PRIOR['v'],scale=PRIOR['m'],size=1),
    "sigma":stats.invwishart.rvs(df=PRIOR['v'],scale=PRIOR['m'],size=1)
}

THETA=copy.deepcopy(THETA_TRUE)

```

```

[4]: p0=FIXED['p0']
v0=np.random.normal(loc=THETA['v0_bar'],scale=np.sqrt(FIXED['sigma_v0']))
t0=np.random.normal(loc=THETA['t0_bar'],scale=np.sqrt(FIXED['sigma_t0']))

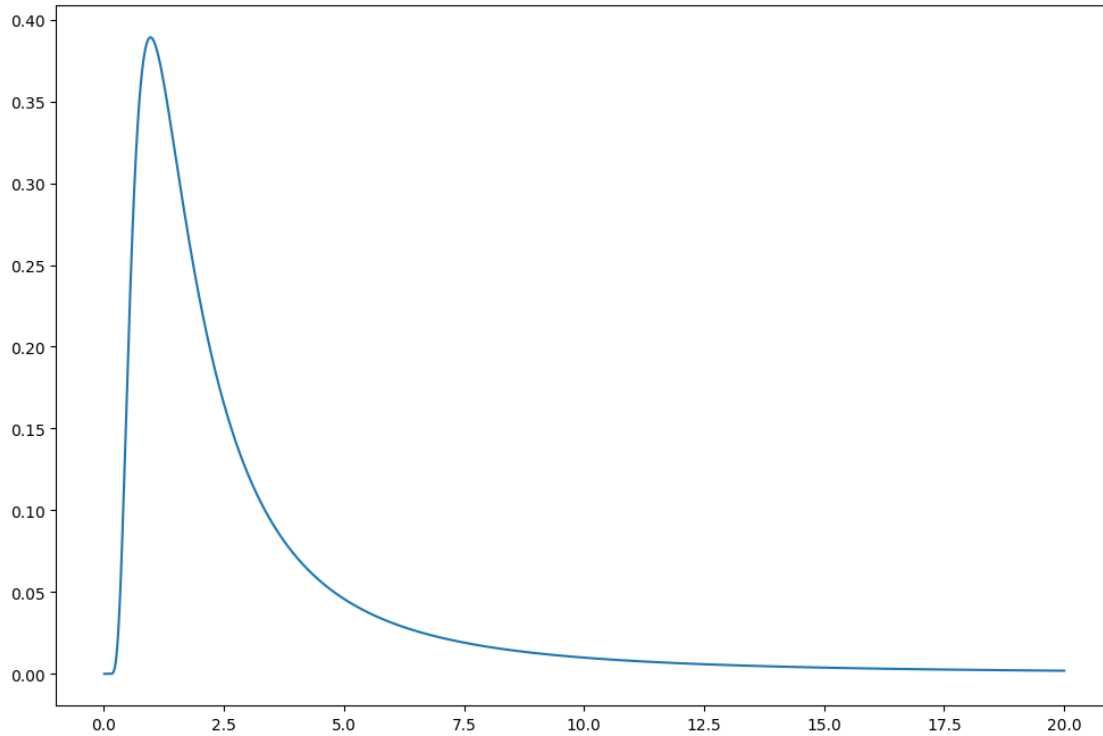
@jit(nopython=True)
def d(t,p0,t0,v0):
    return p0+v0*(t-t0)

```

```

[5]: x=np.linspace(0.01,20,10000)
iw=stats.invwishart.pdf(x,df=PRIOR['v'],scale=PRIOR['m'])
plt.plot(x,iw)
plt.show()

```



```
[6]: ##On génère un sample de N=100 individus avec nos paramètres
Y=np.zeros((N,k))
xi=np.random.normal(loc=0,scale=np.sqrt(THETA['sigma_xi']),size=N)
alpha=np.exp(xi)
eps=np.random.normal(loc=0,scale=np.sqrt(THETA['sigma']),size=(N,k))
tau=np.random.normal(loc=0, scale=np.sqrt(THETA['sigma_tau']),size=N)

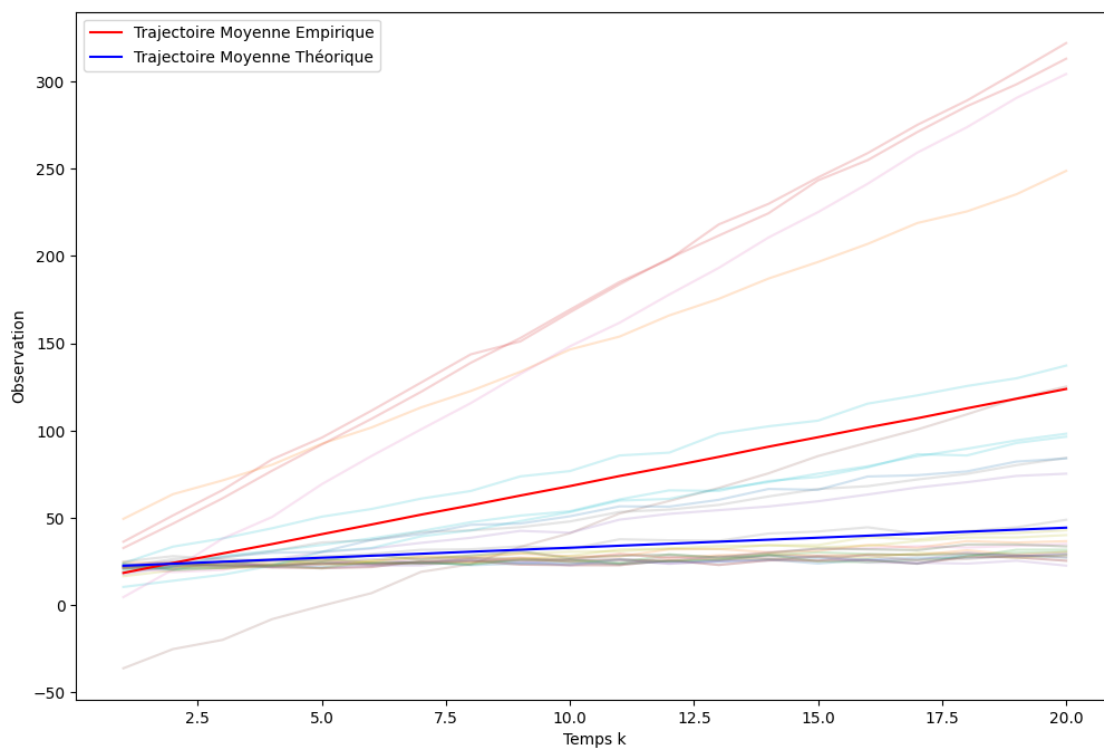
#On crée le dictionnaire associé
LATENT={
    "v0":v0,
    "t0":t0,
    "xi":xi,
    "tau":tau,
    "eps":eps,
    "alpha":alpha
}

column=np.ones(N)
for j in np.arange(0,k):
    indiv_trajectory=alpha*((j+1)*column-t0*column-tau)+t0*column
    Y[:,j]=d(indiv_trajectory,p0,t0=t0,v0=v0)+eps[:,j]
```

```

#Plot quelques trajectoires random et la trajectoire moyenne.
fig,ax=plt.subplots()
for j in np.random.choice(N,30):
    ax.plot(np.arange(1,k+1),Y[j,:],alpha=0.2)
ax.plot(np.arange(1,k+1),np.mean(Y,axis=0),label='Trajectoire Moyenne_
↳Empirique',c='red')
ax.plot(np.arange(1,k+1),d(np.arange(1,k+1),p0,t0=t0,v0=v0),label='Trajectoire_
↳Moyenne Théorique',c='blue')
ax.set_xlabel('Temps k')
ax.set_ylabel('Observation')
ax.legend()
plt.show()

```



La forte variance du paramètre de dilatation α qui est exponentiellement \propto à ξ (qui suit une Normale pouvant prendre des variances de 15-20) fait qu'on a des trajectoires très différentes sur une "petite" cohorte. Similairement, les moyennes théoriques et empiriques sont assez éloignées.

2.3 Question 3

```

[7]: @jit(nopython=True)
def d_ind(t, alpha, tau,
        v0,t0,p0=FIXED['p0']):
    row=np.ones(alpha.shape[0])

```

```

Y=np.zeros((alpha.shape[0],t))
for j in np.arange(0,t):
    indiv_trajectory=alpha*((j+1)*row-t0*row-tau)+t0*row
    Y[:,j]=d(indiv_trajectory, p0=p0, v0=v0, t0=t0)
return Y

#@jit(nopython=True)
def log_posterior_simple(Y,z,t0_bar,v0_bar,sigma_xi,sigma_tau,sigma,
                        sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0']):
    N=int(.5*(z.shape[0]-2))
    t0=z[0]
    v0=z[1]
    xi=np.array(z[2:N+2])
    tau=np.array(z[N+2:])
    if not isinstance(t0, Number) or not isinstance(v0,Number):
        raise ValueError('pas content : t_0 ou v_0 n\'est pas de la bonne forme')

    if xi.shape[0] != N or tau.shape[0] != N:
        raise ValueError('encore moins content')

    l_z=-.5*(np.square(t0_bar-t0)/sigma_t0
            + np.square(v0_bar-v0)/sigma_v0
            + N*np.log(sigma_tau)
            + N*np.log(sigma_xi)
            + np.sum(np.square(tau)/sigma_tau)
            #On remplace les log(alpha) par xi
            + np.sum(np.square(xi)/sigma_xi)
        )

    sq_sum=np.sum(np.square(Y-d_ind(t=Y.shape[1],alpha=np.
→exp(xi),tau=tau,t0=t0,v0=v0)))
    l_y=-.5*N*k*np.log(sigma) -.5* sq_sum/sigma

    return l_z+l_y

#@jit(nopython=True)
#random.normal_multivariate non supporté par numba.
def SRWHM_sampler(Y,z,sigma_prop,t0_bar=np.array(THETA['t0_bar']),v0_bar=np.
→array(THETA['v0_bar']),
                sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
                □
→sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma'],
                pi=log_posterior_simple,max_iter=20000,control=False,
                fenetre=5000,□
→increment_base=2,burnin=None,verbose=True,verbose_plus=False):
    burnin=0 if None else burnin

```



```

N=int(.5*(z.shape[0]-2))
z_list=[]
t0=[]
v0=[]
xi=[]
tau=[]
posterior_list=[]
accept=0
total_accept=0
#Sigma=sigma_prop*np.eye(z.shape[0])
zk=z
for k in range(max_iter):
    #On sample 2N+2 normales car on a un seul sigma_prop. Améliore
    ↳ significativement l'efficacité de l'algo.
    #Plus ou moins similaire à q5-6-7
    z_star=np.random.normal(loc=zk,scale=np.sqrt(sigma_prop))
    #z_star=np.random.multivariate_normal(mean=zk, cov=sigma_prop*np.eye(z.
    ↳ shape[0]))
    l_zstar=pi(Y=Y, z=z_star,t0_bar=t0_bar,v0_bar=v0_bar,
               sigma_t0=sigma_t0,sigma_v0=sigma_v0,
               sigma_xi=sigma_xi,sigma_tau=sigma_tau,sigma=sigma)

    l_z=pi(Y=Y,z=zk,t0_bar=t0_bar,v0_bar=v0_bar,
           sigma_t0=sigma_t0,sigma_v0=sigma_v0,
           sigma_xi=sigma_xi,sigma_tau=sigma_tau,sigma=sigma)
    #On calcule tout dans l'espace des log-probs
    prob_choose=0 if 0<= l_zstar-l_z else l_zstar-l_z
    u=np.random.rand()
    if prob_choose>=np.log(u):
        z_list.append(z_star)
        t0.append(z_star[0])
        v0.append(z_star[1])
        xi.append(z_star[2:N+2])
        tau.append(z_star[N+2:])
        if k>=burnin:
            accept+=1
            total_accept+=1
        posterior_list.append(l_zstar)
        zk=z_star
    else:
        z_list.append(zk)
        posterior_list.append(l_z)
        t0.append(zk[0])
        v0.append(zk[1])
        xi.append(zk[2:N+2])
        tau.append(zk[N+2:])
    #Permet de trouver notre hyperparamètre

```

```

        if control:
            if k>=burnin:
                if k%fenetre==0:
                    if k==0:
                        continue
                    else:
                        if accept/fenetre>0.4:
                            print(f'Taux d\'acceptation trop haut : {accept/
→fenetre :.2%}\nOn augmente sigma_prop de {sigma_prop/increment_base * ((accept/
→fenetre)-0.4)*1:.2f}')
                            sigma_prop+=sigma_prop/increment_base * ((accept/
→fenetre)-0.4)
                            #Sigma=sigma_prop*np.eye(len(z))
                            accept=0
                        elif accept/fenetre<0.2:
                            print(f'Taux d\'acceptation trop bas : {accept/
→fenetre:.2f}\nOn diminue sigma_prop de {sigma_prop/increment_base * (0.
→2-(accept/fenetre)):.2f}')
                            sigma_prop-=sigma_prop/increment_base * (0.2-(accept/
→fenetre))
                            #Sigma=sigma_prop*np.eye(len(z))
                            accept=0
                        else:
                            print(f'Taux acceptation de {accept/fenetre:.2%}')
                            #Retour en direct sur notre taux d'acceptation
                if verbose_plus:
                    if k%fenetre==0 and k>burnin:
                        print(f'Moyenne actuelle ({k}): {total_accept/(k-burnin):.2%}')

            if control:
                print(f'Taux d\'acceptation sur la dernière fenêtre ({fenetre} pas) :
→{accept/fenetre:.2%}')
            elif verbose:
                print(f'Taux d\'acceptation moyen : {total_accept/(max_iter-burnin):.
→2%}')
        return z_list, t0, v0, xi, tau, posterior_list, sigma_prop

```

```

[11]: #Sanity check
#On part d'un z pas trop éloigné et du vrai theta, on regarde si on sample bien
→de la postérieure
burnin=500000
max_iter=1000000
sigma_prop=1*1e-7
z_around=np.concatenate([[LATENT['t0']+np.random.uniform(-1,1)], [LATENT['v0']+np.
→random.uniform(-1,1)]],

```

```

LATENT['xi']+np.random.uniform(-1,1,N),LATENT['tau']+np.
↳random.uniform(-1,1,N)])

sampled_around,t0_sampled_a, v0_sampled_a, xi_sampled_a, tau_sampled_a,␣
↳posterior_list_a, sigma_prop_a=SRWHM_sampler(
    Y,z_around,sigma_prop=sigma_prop,
    t0_bar=np.array(THETA_TRUE['t0_bar']),v0_bar=np.
↳array(THETA_TRUE['v0_bar']),
    sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
    sigma_xi=THETA_TRUE['sigma_xi'],␣
↳sigma_tau=THETA_TRUE['sigma_tau'],sigma=THETA_TRUE['sigma'],
    ␣
↳control=False,max_iter=max_iter,fenetre=7000,burnin=burnin,increment_base=2,verbose=True,␣
↳verbose_plus=True
)

```

```

Moyenne actuelle (k=504000):  40.15%
Moyenne actuelle (k=511000):  39.82%
Moyenne actuelle (k=518000):  39.25%
Moyenne actuelle (k=525000):  39.42%
Moyenne actuelle (k=532000):  39.22%
Moyenne actuelle (k=539000):  39.19%
Moyenne actuelle (k=546000):  39.27%
Moyenne actuelle (k=553000):  39.24%
Moyenne actuelle (k=560000):  39.25%
Moyenne actuelle (k=567000):  39.20%
Moyenne actuelle (k=574000):  39.16%
Moyenne actuelle (k=581000):  39.19%
Moyenne actuelle (k=588000):  39.19%
Moyenne actuelle (k=595000):  39.24%
Moyenne actuelle (k=602000):  39.23%
Moyenne actuelle (k=609000):  39.20%
Moyenne actuelle (k=616000):  39.22%
Moyenne actuelle (k=623000):  39.19%
Moyenne actuelle (k=630000):  39.20%
Moyenne actuelle (k=637000):  39.19%
Moyenne actuelle (k=644000):  39.21%
Moyenne actuelle (k=651000):  39.25%
Moyenne actuelle (k=658000):  39.25%
Moyenne actuelle (k=665000):  39.26%
Moyenne actuelle (k=672000):  39.27%
Moyenne actuelle (k=679000):  39.30%
Moyenne actuelle (k=686000):  39.28%
Moyenne actuelle (k=693000):  39.24%
Moyenne actuelle (k=700000):  39.24%
Moyenne actuelle (k=707000):  39.25%
Moyenne actuelle (k=714000):  39.29%

```

Moyenne actuelle (k=721000): 39.27%
 Moyenne actuelle (k=728000): 39.26%
 Moyenne actuelle (k=735000): 39.27%
 Moyenne actuelle (k=742000): 39.27%
 Moyenne actuelle (k=749000): 39.27%
 Moyenne actuelle (k=756000): 39.28%
 Moyenne actuelle (k=763000): 39.28%
 Moyenne actuelle (k=770000): 39.28%
 Moyenne actuelle (k=777000): 39.27%
 Moyenne actuelle (k=784000): 39.24%
 Moyenne actuelle (k=791000): 39.25%
 Moyenne actuelle (k=798000): 39.22%
 Moyenne actuelle (k=805000): 39.22%
 Moyenne actuelle (k=812000): 39.21%
 Moyenne actuelle (k=819000): 39.21%
 Moyenne actuelle (k=826000): 39.22%
 Moyenne actuelle (k=833000): 39.22%
 Moyenne actuelle (k=840000): 39.22%
 Moyenne actuelle (k=847000): 39.23%
 Moyenne actuelle (k=854000): 39.24%
 Moyenne actuelle (k=861000): 39.22%
 Moyenne actuelle (k=868000): 39.23%
 Moyenne actuelle (k=875000): 39.23%
 Moyenne actuelle (k=882000): 39.22%
 Moyenne actuelle (k=889000): 39.22%
 Moyenne actuelle (k=896000): 39.20%
 Moyenne actuelle (k=903000): 39.23%
 Moyenne actuelle (k=910000): 39.25%
 Moyenne actuelle (k=917000): 39.26%
 Moyenne actuelle (k=924000): 39.25%
 Moyenne actuelle (k=931000): 39.26%
 Moyenne actuelle (k=938000): 39.27%
 Moyenne actuelle (k=945000): 39.26%
 Moyenne actuelle (k=952000): 39.26%
 Moyenne actuelle (k=959000): 39.26%
 Moyenne actuelle (k=966000): 39.26%
 Moyenne actuelle (k=973000): 39.24%
 Moyenne actuelle (k=980000): 39.23%
 Moyenne actuelle (k=987000): 39.22%
 Moyenne actuelle (k=994000): 39.22%
 Taux d'acceptation moyen : 39.23%

```

[13]: #On part d'un z un peu plus éloigné
z_rand=np.random.uniform(-1,1,2*N+2)+5*np.ones(2*N+2)
sampled,t0_sampled, v0_sampled, xi_sampled, tau_sampled, posterior_list,
↳sigma_prop=SRWHM_sampler(
    Y,z_rand,sigma_prop=1*1e-8,
  
```

```

        t0_bar=np.array(THETA_TRUE['t0_bar']),v0_bar=np.
→array(THETA_TRUE['v0_bar']),
        sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
        sigma_xi=THETA_TRUE['sigma_xi'],␣
→sigma_tau=THETA_TRUE['sigma_tau'],sigma=THETA_TRUE['sigma'],
        ␣
→control=False,max_iter=max_iter,fenetre=40000,burnin=burnin,increment_base=2,verbose=True,␣
→verbose_plus=True)

```

```

Moyenne actuelle (k=520000):  24.52%
Moyenne actuelle (k=560000):  24.79%
Moyenne actuelle (k=600000):  24.63%
Moyenne actuelle (k=640000):  24.20%
Moyenne actuelle (k=680000):  23.48%
Moyenne actuelle (k=720000):  22.54%
Moyenne actuelle (k=760000):  21.59%
Moyenne actuelle (k=800000):  20.62%
Moyenne actuelle (k=840000):  19.76%
Moyenne actuelle (k=880000):  18.93%
Moyenne actuelle (k=920000):  18.22%
Moyenne actuelle (k=960000):  17.60%
Taux d'acceptation moyen : 17.04%

```

[16]: *#On part du vrai z*

```

z_true=np.concatenate([[LATENT['t0']], [LATENT['v0']], LATENT['xi'], LATENT['tau']])
sampled_true,t0_sampled_true, v0_sampled_true, xi_sampled_true,␣
→tau_sampled_true, posterior_list_true, sigma_prop_true=SRWHM_sampler(
        Y,z_true,sigma_prop=1e-7,
        t0_bar=np.array(THETA_TRUE['t0_bar']),v0_bar=np.
→array(THETA_TRUE['v0_bar']),
        sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
        sigma_xi=THETA_TRUE['sigma_xi'],␣
→sigma_tau=THETA_TRUE['sigma_tau'],sigma=THETA_TRUE['sigma'],
        ␣
→control=False,max_iter=max_iter,fenetre=40000,burnin=burnin,increment_base=2,verbose=True,␣
→verbose_plus=True)

```

```

Moyenne actuelle (k=520000):  38.06%
Moyenne actuelle (k=560000):  38.28%
Moyenne actuelle (k=600000):  38.50%
Moyenne actuelle (k=640000):  38.65%
Moyenne actuelle (k=680000):  38.62%
Moyenne actuelle (k=720000):  38.66%
Moyenne actuelle (k=760000):  38.65%
Moyenne actuelle (k=800000):  38.62%
Moyenne actuelle (k=840000):  38.61%
Moyenne actuelle (k=880000):  38.60%
Moyenne actuelle (k=920000):  38.57%

```

Moyenne actuelle (k=960000): 38.53%

Taux d'acceptation moyen : 38.48%

```
[17]: fig,(ax1,ax2,ax3)=plt.subplots(3)
      sampled_good=sampled_around[burnin:]
      idx=np.random.randint(1, int(.5*(max_iter-burnin)))
      sampled_good=sampled_good[-idx]
      ax1.plot(sampled_good-z_true)
      ax1.set_xlabel("$\mathcal{Z}$")
      ax1.set_ylabel('$z_i - \hat{z}_i$')
      ax1.set_title(f'Différence entre un $\mathcal{Z}$ échantillonné et réel en
      ↳partant du voisinage, en moyenne sur tous les $\mathcal{Z}$: {np.
      ↳mean(sampled_around[burnin:]-z_true):.2f}')

      sampled_rand=sampled[burnin:]
      idx=np.random.randint(1, int(.5*(max_iter-burnin)))
      sampled_rand=sampled_rand[-idx]
      ax2.plot(sampled_rand-z_true)
      ax2.set_xlabel("$\mathcal{Z}$")
      ax2.set_ylabel('$z_i - \hat{z}_i$')
      ax2.set_title(f'Différence entre un $\mathcal{Z}$ échantillonné et réel en
      ↳partant d\'une initialisation +/- aléatoire, en moyenne sur tous les
      ↳$\mathcal{Z}$: {np.mean(sampled[burnin:]-z_true):.2f}')

      sampled_trueburn=sampled_true[burnin:]
      idx=np.random.randint(1, int(.5*(max_iter-burnin)))
      sampled_trueburn=sampled_trueburn[-idx]
      ax3.plot(sampled_trueburn-z_true)
      ax3.set_xlabel("$\mathcal{Z}$")
      ax3.set_ylabel('$\hat{z}_i - z_i$')
      ax3.set_title(f'Différence entre un $\mathcal{Z}$ échantillonné et réel en
      ↳partant du vrai $\mathcal{Z}$, en moyenne sur tous les $\mathcal{Z}$: {np.
      ↳mean(sampled_true[burnin:]-z_true):.2f}')

      fig.tight_layout()
      plt.show()
```



```
[18]: objectif=-log_posterior_simple(Y,z_true,t0_bar=THETA_TRUE['t0_bar'],v0_bar=THETA_TRUE['v0_bar'],
    ↪sigma_xi=THETA_TRUE['v0_bar'],sigma_tau=THETA_TRUE['sigma_tau'],sigma=THETA_TRUE['sigma'],
    ↪sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'])
fig,(ax1,ax2,ax3)=plt.subplots(3)
ax1.plot(-1*np.array(posterior_list_a))
ax1.set_xlabel("Itérations")
ax1.set_ylabel('Log-likelihood')
ax1.hlines(objectif,0, max_iter, linestyle='dashdot',↪
    ↪label='Objectif',color='k')
ax1.semilogy()
ax1.legend()
ax1.set_title('Convergence de la log-vraisemblance')

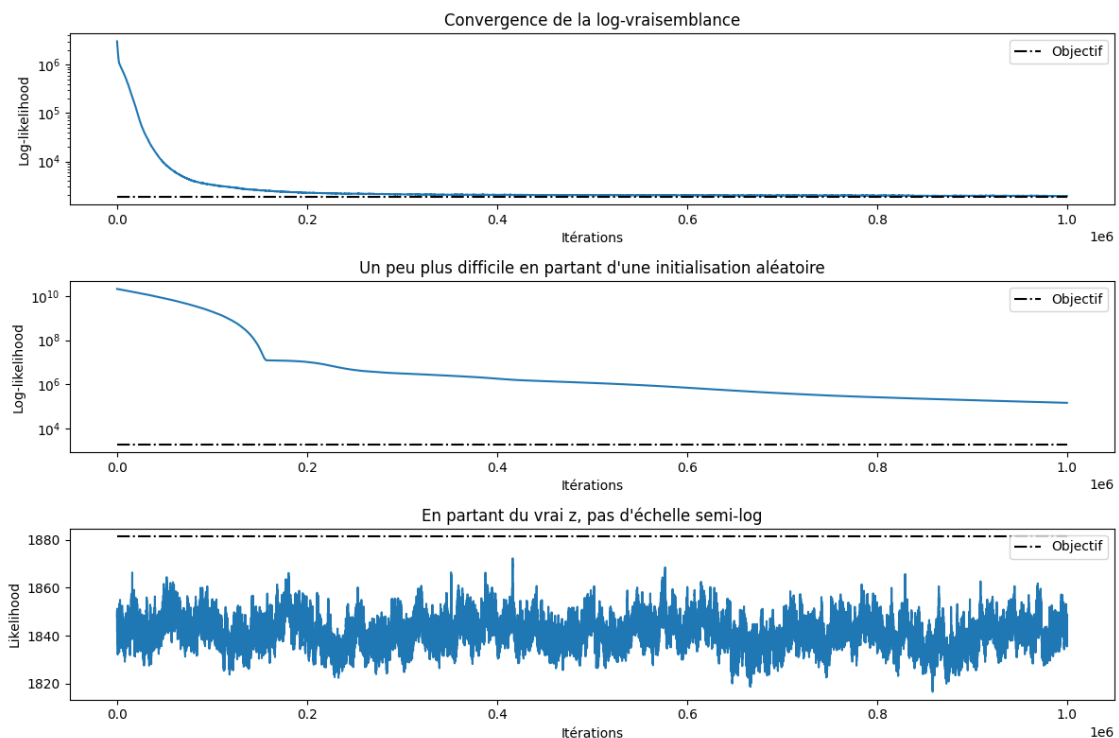
ax2.plot(-1*np.array(posterior_list))
ax2.set_xlabel("Itérations")
ax2.set_ylabel('Log-likelihood')
ax2.hlines(objectif,0, max_iter, linestyle='dashdot',↪
    ↪label='Objectif',color='k')
ax2.legend()
ax2.semilogy()
ax2.set_title('Un peu plus difficile en partant d\'une initialisation aléatoire')
```

```

ax3.plot(-1*np.array(posterior_list_true))
ax3.set_xlabel("Itérations")
ax3.set_ylabel('Likelihood')
ax3.hlines(objectif,0, max_iter, linestyle='dashdot',
↪label='Objectif',color='k')
ax3.legend()
#ax3.semilogy()
ax3.set_title('En partant du vrai z, pas d\'échelle semi-log')

fig.tight_layout()
plt.show()

```



```

[19]: def plot_chain(
        Y, samples, evaluated, burnin, THETA, FIXED, nsig=1, fmt='-',
↪y_range=None,
        width=3000, height=800, margins={'l':20, 'r':20, 't':50, 'b':20}):
    ##Adapté de exowanderer
    plasma = [
        'rgb(13, 8, 135, 1.0)',
        'rgb(70, 3, 159, 1.0)',
        'rgb(114, 1, 168, 1.0)',
        'rgb(156, 23, 158, 1.0)',
        'rgb(189, 55, 134, 1.0)',

```



```

        'rgb(216, 87, 107, 1.0)',
        'rgb(237, 121, 83, 1.0)',
        'rgb(251, 159, 58, 1.0)',
        'rgb(253, 202, 38, 1.0)',
        'rgb(240, 249, 33, 1.0)'
    ]

    if str(evaluated)=="xi" or str(evaluated)=="tau":
        raise ValueError('Ne peut pas plotter une chaîne multidimensionnelle')

    estimate = np.mean(samples[burnin:])
    stddev = np.std(samples[burnin:])
    title = f'On a une estimation de : {estimate:0.2f} ± {stddev:0.2f} (vrai_
    ↳{str(evaluated)}={np.mean(LATENT[evaluated]):.2f})'

    num_samples = len(samples)
    idx_burnin = burnin
    #On limite le nombre de points plottés à 100k
    if len(samples) > 100*1e3:
        denominateur=int(len(samples)/(100*1e3))
        samples=np.concatenate([[samples[denominateur*i]] for i in_
    ↳range(int(len(samples)/denominateur))])
        idx_burnin=int(burnin/denominateur)
        num_samples=len(samples)

    sample_steps = np.arange(num_samples)

    window = int(0.2*num_samples)
    df = pd.DataFrame(samples, columns=['samples'])
    df['low_q'] = df['samples'].rolling(window=window, center=True,
    ↳min_periods=0).quantile(q=0.05)
    df['high_q'] = df['samples'].rolling(window=window, center=True,
    ↳min_periods=0).quantile(q=0.95)

    t0=LATENT['t0']
    v0=LATENT['v0']
    xi=LATENT['xi']
    tau=LATENT['tau']

    samples_posterior = samples[idx_burnin:]
    samples_burnin = samples[:idx_burnin]

    estimate = np.mean(samples_posterior)
    stddev = np.std(samples_posterior)

```

```

if y_range is None:
    std_post = np.std(samples_posterior)
    y_range = min(samples) - nsig * std_post, max(samples) + nsig * std_post

if str(evaluated)=="t0":
    x_kde_posterior = np.
    ↪ linspace(min(samples_posterior),max(samples_posterior),len(samples_posterior))
    y_kde_posterior = np.array([np.exp(log_posterior_simple(Y,np.
    ↪ concatenate([[i],
                                                                [v0],xi,tau)),t0_bar=np.
    ↪ array(THETA['t0_bar']),v0_bar=np.array(THETA['v0_bar']),
        sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
        ↪
    ↪ sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma']))↪
    ↪ for i in sorted(samples_posterior))]
        #truth = np.asarray([stats.norm.pdf(x, loc=t0, scale=np.
    ↪ sqrt(FIXED['sigma_t0'])) for i in np.
    ↪ linspace(min(samples_posterior),max(samples_posterior),100)])

    x_kde_burnin = np.
    ↪ linspace(min(samples_burnin),max(samples_burnin),len(samples_burnin))
    y_kde_burnin = np.array([np.exp(log_posterior_simple(Y,np.
    ↪ concatenate([[i],
                                                                [v0],xi,tau)),t0_bar=np.
    ↪ array(THETA['t0_bar']),v0_bar=np.array(THETA['v0_bar']),
        sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
        ↪
    ↪ sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma']))↪
    ↪ for i in sorted(samples_burnin))]

elif str(evaluated)=="v0":
    x_kde_posterior = np.
    ↪ linspace(min(samples_posterior),max(samples_posterior),len(samples_posterior))
    y_kde_posterior = np.array([np.exp(log_posterior_simple(Y,np.
    ↪ concatenate([
                                                                [t0],[i],↪
    ↪ xi,tau]),t0_bar=np.array(THETA['t0_bar']),v0_bar=np.array(THETA['v0_bar']),
        sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
        ↪
    ↪ sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma']))↪
    ↪ for i in sorted(samples_posterior))]
        #truth = np.asarray([stats.norm.pdf(x, loc=v0, scale=np.
    ↪ sqrt(FIXED['sigma_v0'])) for i in np.
    ↪ linspace(min(samples_posterior),max(samples_posterior),100)])

```

```

x_kde_burnin = np.
↳ linspace(min(samples_burnin),max(samples_burnin),len(samples_burnin))
y_kde_burnin = np.array([np.exp(log_posterior_simple(Y,np.concatenate([
↳ [t0],[i],xi,tau])),t0_bar=np.array(THETA['t0_bar']),v0_bar=np.
↳ array(THETA['v0_bar']),
sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma']))
↳ for i in sorted(samples_burnin)])
else:
raise ValueError('Pas possible de plotter une chaîne de Markov
↳ multidimensionnelle.')
```

```

kde_trace_posterior = go.Histogram(
y=samples_posterior,
histnorm='probability density',
name='Distribution Postérieure',
xaxis="x2",
yaxis="y2",
)

kde_trace_burnin = go.Histogram(
y=samples_burnin,
histnorm='probability density',
name='Distribution Burnin',
xaxis="x2",
yaxis="y2",
)

plots = [
go.Scatter(
x=sample_steps,
y=df['low_q'],
line={'color':'rgba(255, 0, 0, 0.0)'},
showlegend=True
),

# fill between the endpoints of this trace and the endpoints of the
↳ trace before it
go.Scatter(
x=sample_steps,
y=df['high_q'],
line={'color':'rgba(255, 0, 0, 0.0)'},
fill="tonextx",

```

```

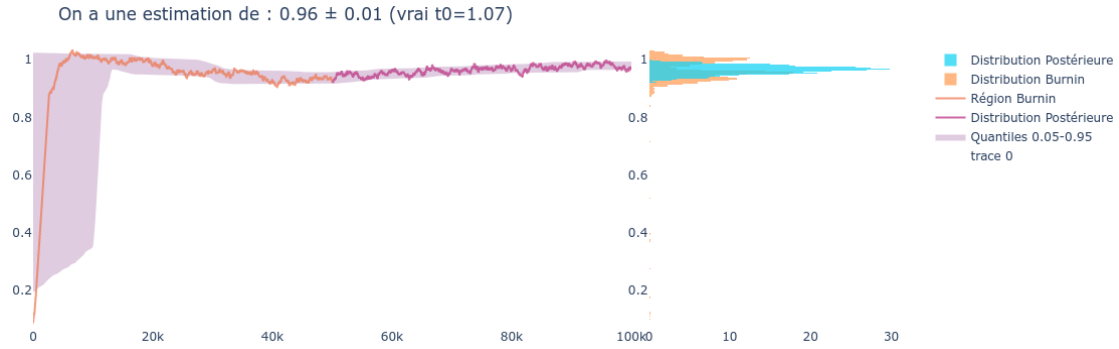
        fillcolor='rgba(100, 0, 100, 0.20)',
        name='Quantiles 0.05-0.95'
    ),
    go.Scatter(
        x=sample_steps[idx_burnin:],
        y=samples_posterior,
        name='Distribution Postérieure',
        line={'color':plasma[4]}
    ),
    go.Scatter(
        x=sample_steps[:idx_burnin],
        y=samples_burnin,
        name='Région Burnin',
        line={'color':plasma[6]}
    ),
]

layout = go.Layout(
    title=title,
    xaxis={'domain': [0, 0.68], 'showgrid': False},
    xaxis2={'domain': [0.7, 1], 'showgrid': False},
    yaxis={'range':y_range, 'showgrid': False},
    yaxis2={
        'anchor': 'x2',
        'range': y_range,
        'showgrid': False
    },
    #width=width,
    #height=height,
    margin=margins,
    plot_bgcolor='rgba(255, 255, 255, 1)',
    paper_bgcolor='rgba(255, 255, 255, 1)'
)

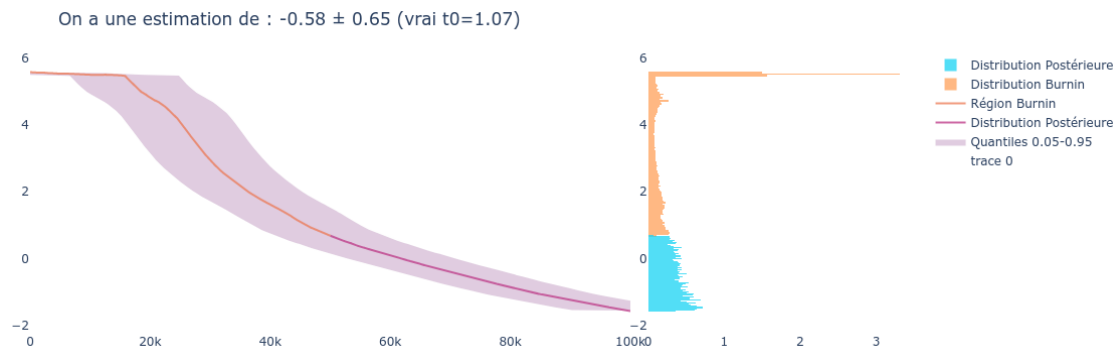
fig = go.Figure(plots, layout=layout)
fig.add_trace(kde_trace_burnin)
fig.add_trace(kde_trace_posterior)
fig.update_layout(barmode='overlay')
fig.update_traces(opacity=0.75)
fig.show()

```

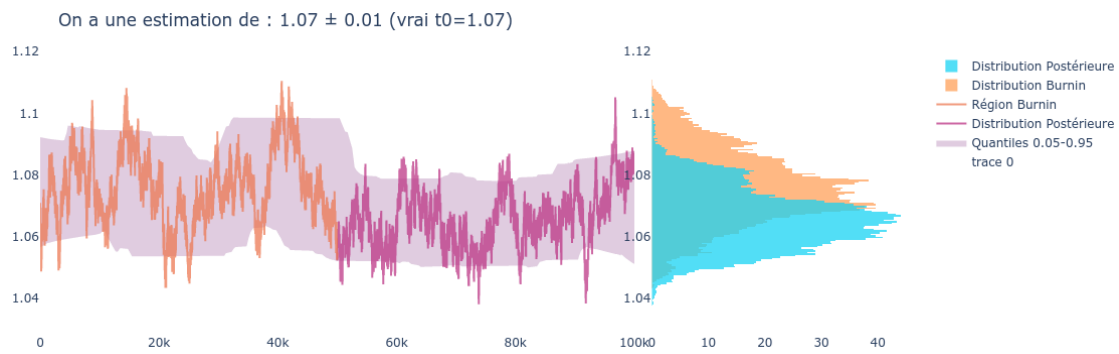
```
[20]: plot_chain(Y,np.array(t0_sampled_a),"t0", burnin,THETA_TRUE,FIXED)
```



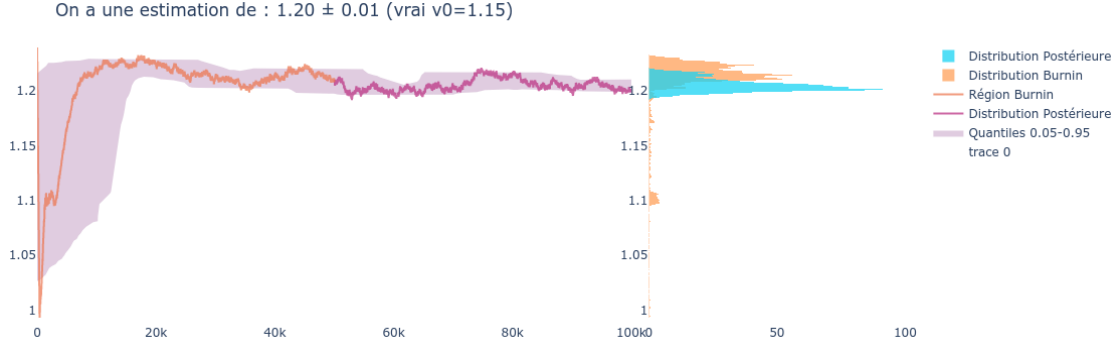
[24] : `plot_chain(Y,np.array(t0_sampled),"t0", burnin,THETA_TRUE,FIXED)`



[25] : `plot_chain(Y,np.array(t0_sampled_true),"t0", burnin,THETA_TRUE,FIXED)`



[26] : `plot_chain(Y,np.array(v0_sampled_a),"v0", burnin,THETA_TRUE,FIXED)`



On voit bien que l'échantillonneur Hastings-Metropolis vanilla est très sensible aux conditions de départ, et qu'il a du mal à récupérer les vrais z s'il part loin. Après 1 million d'itérations, notre t_0 shifté de $+5$ n'a toujours pas convergé (il diverge même).

L'échantillonneur fonctionne bien si on part de la bonne région ou aux alentours.

On est obligé de prendre des σ_{prop} ridiculement petits si on veut avoir un taux d'acceptation décent.

Ceci est dû au fait qu'on update tous les paramètres en même temps de manière aléatoire. Les chances d'aller dans la bonne direction pour $2N+2$ dimensions sont infimes, donc on reste sur place.

2.4 Question 4

D'après la question 1 on sait qu'on peut écrire la log-densité jointe :

$$\log p(y, z, \theta) = -\Phi(\theta) + \langle S(y, z) \mid \Psi(\theta) \rangle$$

C'est une fonction log-concave qu'on cherche à maximiser à chaque étape du MCMC-SAEM. Pour ce faire on résout l'équation

$$\begin{aligned} \theta^{(k+1)} &= \operatorname{argmax}_{\theta \in \Theta} -\Phi(\theta) + \langle S(y, z) \mid \Psi(\theta) \rangle \\ &\Leftrightarrow \nabla_{\theta} \log p(y, z, \theta) = 0 \\ &\Leftrightarrow \nabla_{\theta} \{-\Phi(\theta) + \langle S(y, z) \mid \Psi(\theta) \rangle\} = 0 \\ &\Leftrightarrow -\nabla_{\theta} \Phi(\theta) + \langle S(y, z) \mid \nabla_{\theta} \Psi(\theta) \rangle = 0 \quad \text{car } S \text{ est indépendant de } \theta \end{aligned}$$

On résoud indépendamment $-\nabla_{\theta} \Phi(\theta) + \langle S(y, z) \mid \nabla_{\theta} \Psi(\theta) \rangle = 0$ pour chaque composante de θ car elles sont indépendantes. On note :

$$\begin{aligned}
-\nabla_{\theta}\Phi(\theta) &= - \begin{pmatrix} \nabla_{\theta}\Phi_{\bar{t}_0}(\theta) \\ \nabla_{\theta}\Phi_{\bar{v}_0}(\theta) \\ \nabla_{\theta}\Phi_{\sigma_{\xi}^2}(\theta) \\ \nabla_{\theta}\Phi_{\sigma_{\tau}^2}(\theta) \\ \nabla_{\theta}\Phi_{\sigma^2}(\theta) \end{pmatrix} \\
&= -\nabla_{\theta} \begin{pmatrix} \frac{\bar{t}_0^2}{2\sigma_{t_0}^2} + \frac{\bar{t}_0^2}{2s_{t_0}^2} - \frac{2\bar{t}_0\bar{\bar{t}}_0}{2s_{t_0}^2} \\ \frac{\bar{v}_0^2}{2\sigma_{v_0}^2} + \frac{\bar{v}_0^2}{2s_{v_0}^2} - \frac{2\bar{v}_0\bar{\bar{v}}_0}{2s_{v_0}^2} \\ (\frac{N}{2} + \frac{m}{2} + 1) \log \sigma_{\xi}^2 + \frac{v^2}{2\sigma_{\xi}^2} \\ (\frac{N}{2} + \frac{m}{2} + 1) \log \sigma_{\tau}^2 + \frac{v^2}{2\sigma_{\tau}^2} \\ (\frac{Nk}{2} + \frac{m}{2} + 1) \log \sigma^2 + \frac{v^2}{2\sigma^2} \end{pmatrix} = - \begin{pmatrix} \bar{t}_0 \times \left(\frac{1}{\sigma_{t_0}^2} + \frac{1}{s_{t_0}^2} \right) - \frac{\bar{\bar{t}}_0}{s_{v_0}^2} \\ \bar{v}_0 \times \left(\frac{1}{\sigma_{v_0}^2} + \frac{1}{s_{v_0}^2} \right) - \frac{\bar{\bar{v}}_0}{s_{v_0}^2} \\ \frac{(\frac{N}{2} + \frac{m}{2} + 1)}{\sigma_{\xi}^2} - \frac{v^2}{2(\sigma_{\xi}^2)^2} \\ \frac{(\frac{N}{2} + \frac{m}{2} + 1)}{\sigma_{\tau}^2} - \frac{v^2}{2(\sigma_{\tau}^2)^2} \\ \frac{(\frac{Nk}{2} + \frac{m}{2} + 1)}{\sigma^2} - \frac{v^2}{2(\sigma^2)^2} \end{pmatrix}
\end{aligned}$$

Et pour le dot-product:

$$\begin{aligned}
\langle S(y, z) \mid \nabla_{\theta} \Psi(\theta) \rangle &= \begin{pmatrix} t_0 \\ v_0 \\ \frac{1}{N} \sum_1^N \log(\alpha_i)^2 = \frac{1}{N} \sum_1^N \xi_i^2 \\ \frac{1}{N} \sum_1^N \tau_i^2 \\ \frac{1}{Nk} \sum_1^N \sum_1^k (y_{ij} - d_i(t_{ij}))^2 \end{pmatrix} \cdot \nabla_{\theta} \begin{pmatrix} \frac{2\bar{t}_0}{2\sigma_{t_0}^2} \\ \frac{2\bar{v}_0}{2\sigma_{v_0}^2} \\ -\frac{N}{2\sigma_{\xi}^2} \\ -\frac{N}{2\sigma_{\tau}^2} \\ -\frac{Nk}{2\sigma^2} \end{pmatrix} \\
&= \begin{pmatrix} t_0 \\ v_0 \\ \frac{1}{N} \sum_1^N \log(\alpha_i)^2 = \frac{1}{N} \sum_1^N \xi_i^2 \\ \frac{1}{N} \sum_1^N \tau_i^2 \\ \frac{1}{Nk} \sum_1^N \sum_1^k (y_{ij} - d_i(t_{ij}))^2 \end{pmatrix} \cdot \begin{pmatrix} \frac{1}{\sigma_{t_0}^2} \\ \frac{1}{\sigma_{v_0}^2} \\ \frac{N}{2(\sigma_{\xi}^2)^2} \\ \frac{N}{2(\sigma_{\tau}^2)^2} \\ \frac{Nk}{2(\sigma^2)^2} \end{pmatrix}
\end{aligned}$$

On peut maintenant résoudre $-\nabla_{\theta} \Phi(\theta) + \langle S(y, z) \mid \nabla_{\theta} \Psi(\theta) \rangle = 0$:

$$\begin{aligned}
& \begin{pmatrix} -(\bar{t}_0 \times \left(\frac{1}{\sigma_{t_0}^2} + \frac{1}{s_{t_0}^2}\right) - \frac{\bar{\bar{t}}_0}{s_{v_0}^2}) \\ -(\bar{v}_0 \times \left(\frac{1}{\sigma_{v_0}^2} + \frac{1}{s_{v_0}^2}\right) - \frac{\bar{\bar{v}}_0}{s_{v_0}^2}) \\ -(\frac{(\frac{N}{2} + \frac{m}{2} + 1)}{\sigma_{\xi}^2} - \frac{v^2}{2(\sigma_{\xi}^2)^2}) \\ -(\frac{(\frac{N}{2} + \frac{m}{2} + 1)}{\sigma_{\tau}^2} - \frac{v^2}{2(\sigma_{\tau}^2)^2}) \\ -(\frac{(\frac{Nk}{2} + \frac{m}{2} + 1)}{\sigma^2} - \frac{v^2}{2(\sigma^2)^2}) \end{pmatrix} + \begin{pmatrix} \frac{t_0}{\sigma_{t_0}^2} \\ \frac{v_0}{\sigma_{v_0}^2} \\ \frac{1}{N} \sum_1^N \log(\alpha_i)^2 \times \frac{N}{2(\sigma_{\xi}^2)^2} \\ \frac{1}{N} \sum_1^N \tau_i^2 \times \frac{N}{2(\sigma_{\tau}^2)^2} \\ \frac{1}{Nk} \sum_1^N \sum_1^k (y_{ij} - d_i(t_{ij}))^2 \times \frac{Nk}{2(\sigma^2)^2} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \\
\Leftrightarrow & \begin{pmatrix} \bar{t}_0 = \left(\frac{t_0}{\sigma_{t_0}^2} + \frac{\bar{\bar{t}}_0}{s_{t_0}^2}\right) \times \left(\frac{1}{\sigma_{t_0}^2} + \frac{1}{s_{t_0}^2}\right)^{-1} \\ \bar{v}_0 = \left(\frac{v_0}{\sigma_{v_0}^2} + \frac{\bar{\bar{v}}_0}{s_{v_0}^2}\right) \times \left(\frac{1}{\sigma_{v_0}^2} + \frac{1}{s_{v_0}^2}\right)^{-1} \\ \sigma_{\xi}^2 = \frac{\sum_1^N \log(\alpha_i)^2 + v^2}{N + m + 2} \\ \sigma_{\tau}^2 = \frac{\sum_1^N \tau_i^2 + v^2}{N + m + 2} \\ \sigma^2 = \frac{\left(\sum_1^N \sum_1^k (y_{ij} - d_i(t_{ij}))^2\right) + v^2}{Nk + m + 2} \end{pmatrix}
\end{aligned}$$

On en conclut pour notre update des θ :

$$\theta^{(k+1)} = \begin{pmatrix} \left(\frac{S_{\bar{t}_0}^{(k+1)}}{\sigma_{t_0}^2} + \frac{\bar{t}_0}{s_{t_0}^2} \right) \times \left(\frac{1}{\sigma_{t_0}^2} + \frac{1}{s_{t_0}^2} \right)^{-1} \\ \left(\frac{S_{\bar{v}_0}^{(k+1)}}{\sigma_{v_0}^2} + \frac{\bar{v}_0}{s_{v_0}^2} \right) \times \left(\frac{1}{\sigma_{v_0}^2} + \frac{1}{s_{v_0}^2} \right)^{-1} \\ \frac{S_{\sigma_\xi}^{(k+1)} N + v^2}{N + m + 2} \\ \frac{S_{\sigma_\tau}^{(k+1)} N + v^2}{N + m + 2} \\ \frac{S_{\sigma^2}^{(k+1)} N k + v^2}{N k + m + 2} \end{pmatrix}$$

```
[32]: #On commence par définir les paramètres du modèle exponentiel
@jit(nopython=True)
def phi_theta(N,k,m=PRIOR['m'],v=PRIOR['v'],
    ↳
    ↳t0_bar=THETA['t0_bar'],sigma_t0=FIXED['sigma_t0'],s_t0=PRIOR['s_t0'],t0_dbar=PRIOR['t0_dbar']
    ↳
    ↳v0_bar=THETA['v0_bar'],sigma_v0=FIXED['sigma_v0'],s_v0=PRIOR['s_v0'],v0_dbar=PRIOR['v0_dbar']
    ↳
    ↳sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma']):

    phi_t=np.square(t0_bar)/sigma_t0 + np.square(t0_bar)/s_t0 - 2*t0_bar*t0_dbar/
    ↳s_t0
    phi_v=np.square(v0_bar)/sigma_v0 + np.square(v0_bar)/s_v0 - 2*v0_bar*v0_dbar/
    ↳s_v0
    phi_sig=(N+m+2)*(np.log(sigma_xi)+np.log(sigma_tau))+(N*k+m+2)*np.
    ↳log(sigma)+(np.square(v)/sigma_xi
    ↳
    ↳+np.square(v)/sigma_tau+np.square(v)/sigma)
    return .5*(phi_t+phi_v+phi_sigma)

@jit(nopython=True)
def s_yz(Y,t0, v0, xi,tau):
    N=Y.shape[0]
    k=Y.shape[1]
    S=np.zeros(5)

    S[0]=t0
```

```

S[1]=v0

#On peut remplacer les log(alpha) par xi pour alléger les calculs
S[2]=np.sum(np.square(xi))/N
S[3]=np.sum(np.square(tau))/N

S[4]=np.mean(np.square(Y-d_ind(t=Y.shape[1], t0=t0, v0=v0, alpha=np.exp(xi),
↳tau=tau)))
    return S

@jit(nopython=True)
def psi_theta(N,k,m=PRIOR['m'],v=PRIOR['v'],
    ↳
    ↳t0_bar=THETA['t0_bar'],sigma_t0=FIXED['sigma_t0'],s_t0=PRIOR['s_t0'],t0_dbar=PRIOR['t0_dbar']
    ↳
    ↳v0_bar=THETA['v0_bar'],sigma_v0=FIXED['sigma_v0'],s_v0=PRIOR['s_v0'],v0_dbar=PRIOR['v0_dbar']
    ↳
    ↳sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma']):
    psi=np.zeros(5)

    psi[0]= t0_bar/sigma_t0
    psi[1]=v0_bar/sigma_v0
    psi[2]=-0.5*N/sigma_xi
    psi[3]=-0.5*N/sigma_tau
    psi[4]=-0.5*N*k/sigma
    return psi

@jit(nopython=True)
def expmodel_loglikelihood(Y,z,N,k,m=PRIOR['m'],v=PRIOR['v'],
    ↳
    ↳t0_bar=THETA['t0_bar'],sigma_t0=FIXED['sigma_t0'],s_t0=PRIOR['s_t0'],t0_dbar=PRIOR['t0_dbar']
    ↳
    ↳v0_bar=THETA['v0_bar'],sigma_v0=FIXED['sigma_v0'],s_v0=PRIOR['s_v0'],v0_dbar=PRIOR['v0_dbar']
    ↳
    ↳sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma']):

    phi=psi_theta(N,k,m,v,t0_bar=t0_bar,v0_bar=v0_bar,sigma_xi=sigma_xi,
    ↳sigma_tau=sigma_tau, sigma=sigma)
    s=s_yz(Y, t0=z[0], v0=z[1], xi=z[2:N+2], tau= z[N+2:])
    psi=psi_theta(N,k,m,v,t0_bar=t0_bar,v0_bar=v0_bar,sigma_xi=sigma_xi,
    ↳sigma_tau=sigma_tau, sigma=sigma)
    return -phi + s@psi

#On calcule le theta optimal à partir de Sk
@jit(nopython=True)

```

```

def max_thetak(S,N,k,t0_bar, v0_bar, sigma_xi, sigma_tau, sigma,
    ↪m=PRIOR['m'],v=PRIOR['v'],
    ↪
    ↪sigma_t0=FIXED['sigma_t0'],s_t0=PRIOR['s_t0'],t0_dbar=PRIOR['t0_dbar'],
    ↪
    ↪sigma_v0=FIXED['sigma_v0'],s_v0=PRIOR['s_v0'],v0_dbar=PRIOR['v0_dbar'])):

    theta_dict=['t0_bar','v0_bar','sigma_xi','sigma_tau','sigma']
    t0_bar_update=(t0_dbar/s_t0 + S[0]/sigma_t0)/(1/sigma_t0 + 1/s_t0)
    v0_bar_update=(v0_dbar/s_v0 + S[1]/sigma_v0)/(1/sigma_v0 + 1/s_v0)

    sigma_xi_update=(S[2]*N+np.square(v))/(N+m+2)
    sigma_tau_update=(S[3]*N+np.square(v))/(N+m+2)

    sigma_update=(S[4]*N*k+np.square(v))/(N*k+m+2)

    #numba_list=List()
    #Pour que numba soit content
    update=[t0_bar_update,v0_bar_update, sigma_xi_update, sigma_tau_update,
    ↪sigma_update]
    update_array=np.array(update)
    THETA=dict(zip(list(theta_dict),update_array))
    #THETA=getNumbaDictFromDict(THETA)
    #print(THETA['sigma'])
    return THETA

#On peut maintenant implémenter le MCMC-SAEM
#@jit(nopython=True)
def MCMC_SAEM(Y,THETA0,FIXED,PRIOR,max_iter, burnin, alpha_burn,
    max_iter_hm=10000, burnin_hm=4000,sigma_hm=2*1e-2,fenetre_hm=4000,
    ↪THETA_TRUE=THETA_TRUE>window=200,verbose=True,verbose_plus=False):
    theta_true=np.array(list(THETA_TRUE.values()))
    N=Y.shape[0]
    k=Y.shape[1]
    zk=np.zeros(2*N+2)
    S=np.zeros(5)
    THETA=THETA0
    for i in range(max_iter):
        z,*_ = SRWHM_sampler(Y, zk, sigma_prop=sigma_hm,
            t0_bar=np.array(THETA['t0_bar']),v0_bar=np.
        ↪array(THETA['v0_bar']),
            sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
            sigma_xi=THETA['sigma_xi'],
        ↪sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma'],
            ↪
        ↪control=False,max_iter=max_iter_hm,fenetre=fenetre_hm,burnin=burnin_hm,increment_base=2,verbo

```

```

#On ne garde que le dernier sample - faire un tirage aléatoire?
idx=np.random.randint(1, int(.5*(max_iter-burnin)))
zk=z[-idx]
if i<=burnin:
    epsilon=1
else:
    epsilon=(i-burnin)**(-alpha_burn)
    if i%window==0:
        theta_values=np.array(list(THETA.values()))
        print(theta_true-theta_values)
S+=epsilon*(s_yz(Y,t0=zk[0],v0=zk[1], xi=zk[2:N+2],tau=zk[N+2:]) - S)
THETA_UPDATE=max_thetak(S,N,k,m=PRIOR['m'],v=PRIOR['v'],
    □
↳t0_bar=THETA['t0_bar'],sigma_t0=FIXED['sigma_t0'],s_t0=PRIOR['s_t0'],t0_dbar=PRIOR['t0_dbar']
    □
↳v0_bar=THETA['v0_bar'],sigma_v0=FIXED['sigma_v0'],s_v0=PRIOR['s_v0'],v0_dbar=PRIOR['v0_dbar']
    □
↳sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma'])

    #THETA_UPDATE=
    #print(isinstance(THETA_UPDATE,dict))
    THETA=THETA_UPDATE
    #print(THETA)
return THETA

```

```

[35]: #Pour estimer le nombre d'itérations de HM au début
burnin=500000
max_iter=1000000
sigma_prop=2*1e-7

z_zero=np.zeros(2*N+2)
sampled_zero,t0_sampled_zero, v0_sampled_zero, xi_sampled_zero,□
↳tau_sampled_zero, posterior_list_zero, sigma_prop_zero=SRWHM_sampler(
    Y,z_zero,sigma_prop=sigma_prop,
    t0_bar=np.array(THETA_TRUE['t0_bar']),v0_bar=np.
↳array(THETA_TRUE['v0_bar']),
    sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
    sigma_xi=THETA_TRUE['sigma_xi'],□
↳sigma_tau=THETA_TRUE['sigma_tau'],sigma=THETA_TRUE['sigma'],
    □
↳control=False,max_iter=max_iter,fenetre=7000,burnin=burnin,increment_base=2,verbose=True,□
↳verbose_plus=True
    )

```

Moyenne actuelle (k=504000): 28.50%

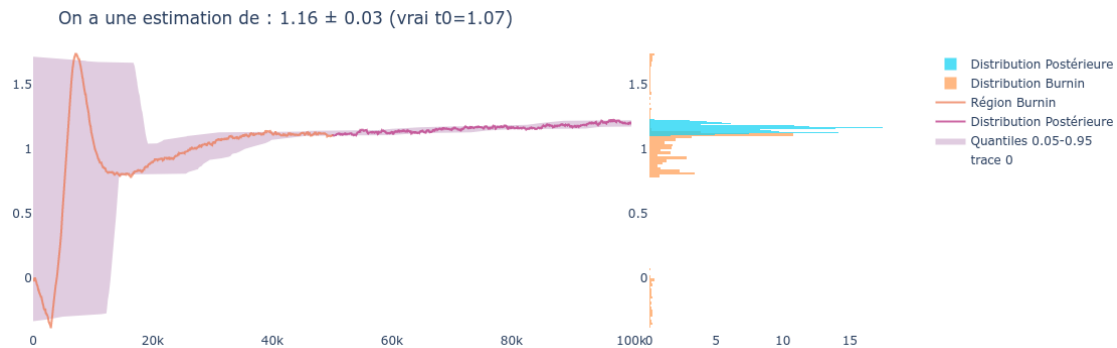
Moyenne actuelle (k=511000): 28.33%

Moyenne actuelle (k=518000): 28.56%

Moyenne actuelle (k=525000):	28.48%
Moyenne actuelle (k=532000):	28.34%
Moyenne actuelle (k=539000):	28.31%
Moyenne actuelle (k=546000):	28.23%
Moyenne actuelle (k=553000):	28.15%
Moyenne actuelle (k=560000):	28.22%
Moyenne actuelle (k=567000):	28.16%
Moyenne actuelle (k=574000):	28.18%
Moyenne actuelle (k=581000):	28.19%
Moyenne actuelle (k=588000):	28.12%
Moyenne actuelle (k=595000):	28.08%
Moyenne actuelle (k=602000):	28.02%
Moyenne actuelle (k=609000):	28.01%
Moyenne actuelle (k=616000):	28.07%
Moyenne actuelle (k=623000):	28.07%
Moyenne actuelle (k=630000):	28.09%
Moyenne actuelle (k=637000):	28.06%
Moyenne actuelle (k=644000):	28.03%
Moyenne actuelle (k=651000):	27.96%
Moyenne actuelle (k=658000):	27.99%
Moyenne actuelle (k=665000):	28.00%
Moyenne actuelle (k=672000):	28.04%
Moyenne actuelle (k=679000):	28.03%
Moyenne actuelle (k=686000):	28.06%
Moyenne actuelle (k=693000):	28.08%
Moyenne actuelle (k=700000):	28.10%
Moyenne actuelle (k=707000):	28.15%
Moyenne actuelle (k=714000):	28.13%
Moyenne actuelle (k=721000):	28.13%
Moyenne actuelle (k=728000):	28.12%
Moyenne actuelle (k=735000):	28.13%
Moyenne actuelle (k=742000):	28.11%
Moyenne actuelle (k=749000):	28.12%
Moyenne actuelle (k=756000):	28.13%
Moyenne actuelle (k=763000):	28.13%
Moyenne actuelle (k=770000):	28.12%
Moyenne actuelle (k=777000):	28.11%
Moyenne actuelle (k=784000):	28.11%
Moyenne actuelle (k=791000):	28.11%
Moyenne actuelle (k=798000):	28.12%
Moyenne actuelle (k=805000):	28.11%
Moyenne actuelle (k=812000):	28.08%
Moyenne actuelle (k=819000):	28.09%
Moyenne actuelle (k=826000):	28.10%
Moyenne actuelle (k=833000):	28.10%
Moyenne actuelle (k=840000):	28.11%
Moyenne actuelle (k=847000):	28.11%
Moyenne actuelle (k=854000):	28.10%

Moyenne actuelle (k=861000): 28.09%
 Moyenne actuelle (k=868000): 28.09%
 Moyenne actuelle (k=875000): 28.09%
 Moyenne actuelle (k=882000): 28.09%
 Moyenne actuelle (k=889000): 28.10%
 Moyenne actuelle (k=896000): 28.10%
 Moyenne actuelle (k=903000): 28.10%
 Moyenne actuelle (k=910000): 28.10%
 Moyenne actuelle (k=917000): 28.11%
 Moyenne actuelle (k=924000): 28.12%
 Moyenne actuelle (k=931000): 28.13%
 Moyenne actuelle (k=938000): 28.11%
 Moyenne actuelle (k=945000): 28.12%
 Moyenne actuelle (k=952000): 28.11%
 Moyenne actuelle (k=959000): 28.10%
 Moyenne actuelle (k=966000): 28.10%
 Moyenne actuelle (k=973000): 28.11%
 Moyenne actuelle (k=980000): 28.09%
 Moyenne actuelle (k=987000): 28.09%
 Moyenne actuelle (k=994000): 28.08%
 Taux d'acceptation moyen : 28.08%

```
[36]: plot_chain(Y, np.array(t0_sampled_zero), "t0", burnin, THETA_TRUE, FIXED)
```



```
[38]: #Sanity check. On part d'un theta_0 peu éloigné de theta_star
theta_list=['t0_bar', 'v0_bar', 'sigma_xi', 'sigma_tau', 'sigma']
THETA=copy.deepcopy(THETA_TRUE)
theta_values=np.array(list(THETA.values()))
theta_values=theta_values+np.random.uniform(-1,1,size=theta_values.shape)#+15*np.
    ↳ ones(theta_values.shape[0])
THETA0=dict(zip(list(theta_list), theta_values))

start_time = time.time()
```

```

THETA_FINAL_MCMC=MCMC_SAEM(Y,THETA0=THETA0,FIXED=FIXED,PRIOR=PRIOR,max_iter=1000,
↳burnin=500, alpha_burn=2.5/4,
        max_iter_hm=80000, burnin_hm=50000,sigma_hm=3*1e-7,
↳THETA_TRUE=THETA_TRUE,
        window=100,verbose=True,verbose_plus=False)

duree_mcmc=time.time() - start_time
print("--- %s seconds ---" % (duree_mcmc))

```

```

Taux d'acceptation moyen : 44.94%
Taux d'acceptation moyen : 81.95%
Taux d'acceptation moyen : 76.83%
Taux d'acceptation moyen : 72.28%
Taux d'acceptation moyen : 68.28%
Taux d'acceptation moyen : 65.22%
Taux d'acceptation moyen : 62.02%
Taux d'acceptation moyen : 59.81%
Taux d'acceptation moyen : 56.47%
Taux d'acceptation moyen : 51.73%
Taux d'acceptation moyen : 47.46%
Taux d'acceptation moyen : 45.23%
Taux d'acceptation moyen : 42.45%
Taux d'acceptation moyen : 41.26%
Taux d'acceptation moyen : 40.23%
Taux d'acceptation moyen : 38.98%
Taux d'acceptation moyen : 38.15%
Taux d'acceptation moyen : 36.45%
Taux d'acceptation moyen : 34.96%
Taux d'acceptation moyen : 33.76%
Taux d'acceptation moyen : 32.39%
Taux d'acceptation moyen : 31.49%
Taux d'acceptation moyen : 30.31%
Taux d'acceptation moyen : 30.14%
Taux d'acceptation moyen : 28.98%
Taux d'acceptation moyen : 29.23%
Taux d'acceptation moyen : 28.25%
Taux d'acceptation moyen : 28.32%
Taux d'acceptation moyen : 27.39%
Taux d'acceptation moyen : 26.77%
Taux d'acceptation moyen : 26.48%
Taux d'acceptation moyen : 26.13%
Taux d'acceptation moyen : 25.71%
Taux d'acceptation moyen : 25.65%
Taux d'acceptation moyen : 25.09%
Taux d'acceptation moyen : 24.57%
Taux d'acceptation moyen : 25.01%
Taux d'acceptation moyen : 24.62%

```


Taux d'acceptation moyen : 24.10%
Taux d'acceptation moyen : 24.06%
Taux d'acceptation moyen : 24.18%
Taux d'acceptation moyen : 23.87%
Taux d'acceptation moyen : 23.47%
Taux d'acceptation moyen : 22.97%
Taux d'acceptation moyen : 22.93%
Taux d'acceptation moyen : 23.26%
Taux d'acceptation moyen : 23.34%
Taux d'acceptation moyen : 23.23%
Taux d'acceptation moyen : 22.95%
Taux d'acceptation moyen : 22.38%
Taux d'acceptation moyen : 22.82%
Taux d'acceptation moyen : 22.86%
Taux d'acceptation moyen : 23.04%
Taux d'acceptation moyen : 22.10%
Taux d'acceptation moyen : 22.40%
Taux d'acceptation moyen : 22.03%
Taux d'acceptation moyen : 22.44%
Taux d'acceptation moyen : 22.18%
Taux d'acceptation moyen : 21.49%
Taux d'acceptation moyen : 21.77%
Taux d'acceptation moyen : 22.25%
Taux d'acceptation moyen : 22.32%
Taux d'acceptation moyen : 21.73%
Taux d'acceptation moyen : 21.32%
Taux d'acceptation moyen : 21.16%
Taux d'acceptation moyen : 21.76%
Taux d'acceptation moyen : 21.34%
Taux d'acceptation moyen : 22.16%
Taux d'acceptation moyen : 21.25%
Taux d'acceptation moyen : 21.68%
Taux d'acceptation moyen : 21.48%
Taux d'acceptation moyen : 21.39%
Taux d'acceptation moyen : 21.06%
Taux d'acceptation moyen : 21.72%
Taux d'acceptation moyen : 21.15%
Taux d'acceptation moyen : 21.63%
Taux d'acceptation moyen : 21.46%
Taux d'acceptation moyen : 21.64%
Taux d'acceptation moyen : 21.34%
Taux d'acceptation moyen : 21.29%
Taux d'acceptation moyen : 22.00%
Taux d'acceptation moyen : 21.62%
Taux d'acceptation moyen : 21.94%
Taux d'acceptation moyen : 21.55%
Taux d'acceptation moyen : 21.33%
Taux d'acceptation moyen : 21.68%

Taux d'acceptation moyen : 21.91%
Taux d'acceptation moyen : 21.60%
Taux d'acceptation moyen : 21.46%
Taux d'acceptation moyen : 21.54%
Taux d'acceptation moyen : 20.90%
Taux d'acceptation moyen : 20.93%
Taux d'acceptation moyen : 21.22%
Taux d'acceptation moyen : 20.82%
Taux d'acceptation moyen : 21.35%
Taux d'acceptation moyen : 21.20%
Taux d'acceptation moyen : 21.37%
Taux d'acceptation moyen : 21.41%
Taux d'acceptation moyen : 21.40%
Taux d'acceptation moyen : 21.01%
Taux d'acceptation moyen : 21.17%
Taux d'acceptation moyen : 21.44%
Taux d'acceptation moyen : 21.30%
Taux d'acceptation moyen : 21.71%
Taux d'acceptation moyen : 21.24%
Taux d'acceptation moyen : 21.73%
Taux d'acceptation moyen : 21.14%
Taux d'acceptation moyen : 21.20%
Taux d'acceptation moyen : 21.55%
Taux d'acceptation moyen : 21.63%
Taux d'acceptation moyen : 21.54%
Taux d'acceptation moyen : 21.28%
Taux d'acceptation moyen : 21.56%
Taux d'acceptation moyen : 21.62%
Taux d'acceptation moyen : 21.46%
Taux d'acceptation moyen : 21.89%
Taux d'acceptation moyen : 21.40%
Taux d'acceptation moyen : 21.73%
Taux d'acceptation moyen : 21.95%
Taux d'acceptation moyen : 21.27%
Taux d'acceptation moyen : 21.59%
Taux d'acceptation moyen : 21.95%
Taux d'acceptation moyen : 21.43%
Taux d'acceptation moyen : 20.80%
Taux d'acceptation moyen : 20.85%
Taux d'acceptation moyen : 20.77%
Taux d'acceptation moyen : 20.63%
Taux d'acceptation moyen : 20.50%
Taux d'acceptation moyen : 20.74%
Taux d'acceptation moyen : 21.05%
Taux d'acceptation moyen : 20.93%
Taux d'acceptation moyen : 20.63%
Taux d'acceptation moyen : 20.89%
Taux d'acceptation moyen : 21.35%

Taux d'acceptation moyen : 20.86%
Taux d'acceptation moyen : 20.94%
Taux d'acceptation moyen : 20.99%
Taux d'acceptation moyen : 20.85%
Taux d'acceptation moyen : 20.17%
Taux d'acceptation moyen : 20.56%
Taux d'acceptation moyen : 20.94%
Taux d'acceptation moyen : 20.84%
Taux d'acceptation moyen : 21.16%
Taux d'acceptation moyen : 20.60%
Taux d'acceptation moyen : 20.02%
Taux d'acceptation moyen : 20.22%
Taux d'acceptation moyen : 20.02%
Taux d'acceptation moyen : 20.66%
Taux d'acceptation moyen : 20.59%
Taux d'acceptation moyen : 20.64%
Taux d'acceptation moyen : 20.08%
Taux d'acceptation moyen : 20.49%
Taux d'acceptation moyen : 20.78%
Taux d'acceptation moyen : 20.31%
Taux d'acceptation moyen : 20.93%
Taux d'acceptation moyen : 20.60%
Taux d'acceptation moyen : 20.79%
Taux d'acceptation moyen : 20.93%
Taux d'acceptation moyen : 20.59%
Taux d'acceptation moyen : 20.61%
Taux d'acceptation moyen : 20.74%
Taux d'acceptation moyen : 20.34%
Taux d'acceptation moyen : 20.09%
Taux d'acceptation moyen : 20.03%
Taux d'acceptation moyen : 20.21%
Taux d'acceptation moyen : 19.82%
Taux d'acceptation moyen : 19.92%
Taux d'acceptation moyen : 20.48%
Taux d'acceptation moyen : 19.84%
Taux d'acceptation moyen : 20.76%
Taux d'acceptation moyen : 20.35%
Taux d'acceptation moyen : 20.95%
Taux d'acceptation moyen : 20.41%
Taux d'acceptation moyen : 20.31%
Taux d'acceptation moyen : 20.23%
Taux d'acceptation moyen : 20.35%
Taux d'acceptation moyen : 19.98%
Taux d'acceptation moyen : 19.94%
Taux d'acceptation moyen : 20.44%
Taux d'acceptation moyen : 20.23%
Taux d'acceptation moyen : 20.50%
Taux d'acceptation moyen : 21.23%

Taux d'acceptation moyen : 20.94%
Taux d'acceptation moyen : 20.66%
Taux d'acceptation moyen : 20.72%
Taux d'acceptation moyen : 20.92%
Taux d'acceptation moyen : 20.87%
Taux d'acceptation moyen : 20.31%
Taux d'acceptation moyen : 20.42%
Taux d'acceptation moyen : 20.55%
Taux d'acceptation moyen : 20.23%
Taux d'acceptation moyen : 20.68%
Taux d'acceptation moyen : 20.40%
Taux d'acceptation moyen : 20.77%
Taux d'acceptation moyen : 20.44%
Taux d'acceptation moyen : 20.03%
Taux d'acceptation moyen : 20.38%
Taux d'acceptation moyen : 20.54%
Taux d'acceptation moyen : 19.91%
Taux d'acceptation moyen : 20.92%
Taux d'acceptation moyen : 20.07%
Taux d'acceptation moyen : 19.86%
Taux d'acceptation moyen : 20.29%
Taux d'acceptation moyen : 20.71%
Taux d'acceptation moyen : 20.01%
Taux d'acceptation moyen : 20.54%
Taux d'acceptation moyen : 20.03%
Taux d'acceptation moyen : 20.33%
Taux d'acceptation moyen : 20.32%
Taux d'acceptation moyen : 19.42%
Taux d'acceptation moyen : 20.34%
Taux d'acceptation moyen : 19.98%
Taux d'acceptation moyen : 19.80%
Taux d'acceptation moyen : 20.05%
Taux d'acceptation moyen : 19.77%
Taux d'acceptation moyen : 20.11%
Taux d'acceptation moyen : 20.24%
Taux d'acceptation moyen : 19.76%
Taux d'acceptation moyen : 20.17%
Taux d'acceptation moyen : 20.08%
Taux d'acceptation moyen : 19.66%
Taux d'acceptation moyen : 20.48%
Taux d'acceptation moyen : 20.10%
Taux d'acceptation moyen : 19.83%
Taux d'acceptation moyen : 20.10%
Taux d'acceptation moyen : 19.66%
Taux d'acceptation moyen : 20.25%
Taux d'acceptation moyen : 20.19%
Taux d'acceptation moyen : 20.58%
Taux d'acceptation moyen : 20.42%

Taux d'acceptation moyen : 20.29%
Taux d'acceptation moyen : 20.08%
Taux d'acceptation moyen : 20.10%
Taux d'acceptation moyen : 20.02%
Taux d'acceptation moyen : 19.90%
Taux d'acceptation moyen : 20.67%
Taux d'acceptation moyen : 20.25%
Taux d'acceptation moyen : 20.18%
Taux d'acceptation moyen : 20.22%
Taux d'acceptation moyen : 20.50%
Taux d'acceptation moyen : 20.21%
Taux d'acceptation moyen : 20.44%
Taux d'acceptation moyen : 20.26%
Taux d'acceptation moyen : 20.56%
Taux d'acceptation moyen : 20.38%
Taux d'acceptation moyen : 20.30%
Taux d'acceptation moyen : 20.15%
Taux d'acceptation moyen : 19.96%
Taux d'acceptation moyen : 20.07%
Taux d'acceptation moyen : 20.06%
Taux d'acceptation moyen : 19.98%
Taux d'acceptation moyen : 19.92%
Taux d'acceptation moyen : 19.80%
Taux d'acceptation moyen : 19.81%
Taux d'acceptation moyen : 20.07%
Taux d'acceptation moyen : 19.98%
Taux d'acceptation moyen : 19.73%
Taux d'acceptation moyen : 19.91%
Taux d'acceptation moyen : 19.72%
Taux d'acceptation moyen : 19.71%
Taux d'acceptation moyen : 20.00%
Taux d'acceptation moyen : 20.28%
Taux d'acceptation moyen : 20.02%
Taux d'acceptation moyen : 20.08%
Taux d'acceptation moyen : 19.78%
Taux d'acceptation moyen : 19.92%
Taux d'acceptation moyen : 19.74%
Taux d'acceptation moyen : 20.77%
Taux d'acceptation moyen : 19.93%
Taux d'acceptation moyen : 19.97%
Taux d'acceptation moyen : 19.99%
Taux d'acceptation moyen : 20.20%
Taux d'acceptation moyen : 20.10%
Taux d'acceptation moyen : 20.22%
Taux d'acceptation moyen : 20.23%
Taux d'acceptation moyen : 20.22%
Taux d'acceptation moyen : 20.00%
Taux d'acceptation moyen : 20.24%

Taux d'acceptation moyen : 20.14%
Taux d'acceptation moyen : 19.77%
Taux d'acceptation moyen : 20.09%
Taux d'acceptation moyen : 20.02%
Taux d'acceptation moyen : 19.88%
Taux d'acceptation moyen : 19.77%
Taux d'acceptation moyen : 20.60%
Taux d'acceptation moyen : 20.54%
Taux d'acceptation moyen : 20.28%
Taux d'acceptation moyen : 19.94%
Taux d'acceptation moyen : 20.06%
Taux d'acceptation moyen : 20.08%
Taux d'acceptation moyen : 19.91%
Taux d'acceptation moyen : 19.74%
Taux d'acceptation moyen : 19.79%
Taux d'acceptation moyen : 20.06%
Taux d'acceptation moyen : 19.74%
Taux d'acceptation moyen : 20.26%
Taux d'acceptation moyen : 20.12%
Taux d'acceptation moyen : 20.63%
Taux d'acceptation moyen : 20.40%
Taux d'acceptation moyen : 19.68%
Taux d'acceptation moyen : 20.03%
Taux d'acceptation moyen : 19.72%
Taux d'acceptation moyen : 19.76%
Taux d'acceptation moyen : 19.78%
Taux d'acceptation moyen : 19.97%
Taux d'acceptation moyen : 20.26%
Taux d'acceptation moyen : 19.81%
Taux d'acceptation moyen : 20.31%
Taux d'acceptation moyen : 19.95%
Taux d'acceptation moyen : 20.00%
Taux d'acceptation moyen : 19.55%
Taux d'acceptation moyen : 20.09%
Taux d'acceptation moyen : 19.31%
Taux d'acceptation moyen : 19.94%
Taux d'acceptation moyen : 19.96%
Taux d'acceptation moyen : 19.72%
Taux d'acceptation moyen : 20.34%
Taux d'acceptation moyen : 19.68%
Taux d'acceptation moyen : 19.89%
Taux d'acceptation moyen : 19.95%
Taux d'acceptation moyen : 20.00%
Taux d'acceptation moyen : 20.32%
Taux d'acceptation moyen : 19.98%
Taux d'acceptation moyen : 19.63%
Taux d'acceptation moyen : 19.65%
Taux d'acceptation moyen : 19.48%

Taux d'acceptation moyen : 19.14%
Taux d'acceptation moyen : 19.39%
Taux d'acceptation moyen : 19.61%
Taux d'acceptation moyen : 19.67%
Taux d'acceptation moyen : 19.60%
Taux d'acceptation moyen : 19.44%
Taux d'acceptation moyen : 19.54%
Taux d'acceptation moyen : 19.28%
Taux d'acceptation moyen : 19.47%
Taux d'acceptation moyen : 19.69%
Taux d'acceptation moyen : 19.33%
Taux d'acceptation moyen : 19.65%
Taux d'acceptation moyen : 18.99%
Taux d'acceptation moyen : 19.09%
Taux d'acceptation moyen : 19.31%
Taux d'acceptation moyen : 19.22%
Taux d'acceptation moyen : 18.97%
Taux d'acceptation moyen : 19.32%
Taux d'acceptation moyen : 19.42%
Taux d'acceptation moyen : 19.61%
Taux d'acceptation moyen : 18.89%
Taux d'acceptation moyen : 18.56%
Taux d'acceptation moyen : 18.72%
Taux d'acceptation moyen : 19.39%
Taux d'acceptation moyen : 19.40%
Taux d'acceptation moyen : 19.62%
Taux d'acceptation moyen : 19.59%
Taux d'acceptation moyen : 19.52%
Taux d'acceptation moyen : 18.96%
Taux d'acceptation moyen : 19.34%
Taux d'acceptation moyen : 19.44%
Taux d'acceptation moyen : 19.02%
Taux d'acceptation moyen : 19.35%
Taux d'acceptation moyen : 19.62%
Taux d'acceptation moyen : 19.25%
Taux d'acceptation moyen : 19.23%
Taux d'acceptation moyen : 19.50%
Taux d'acceptation moyen : 19.91%
Taux d'acceptation moyen : 19.22%
Taux d'acceptation moyen : 19.72%
Taux d'acceptation moyen : 19.49%
Taux d'acceptation moyen : 19.21%
Taux d'acceptation moyen : 19.36%
Taux d'acceptation moyen : 19.43%
Taux d'acceptation moyen : 19.67%
Taux d'acceptation moyen : 19.08%
Taux d'acceptation moyen : 19.83%
Taux d'acceptation moyen : 19.28%

Taux d'acceptation moyen : 19.55%
Taux d'acceptation moyen : 19.49%
Taux d'acceptation moyen : 19.90%
Taux d'acceptation moyen : 19.96%
Taux d'acceptation moyen : 19.54%
Taux d'acceptation moyen : 19.60%
Taux d'acceptation moyen : 19.40%
Taux d'acceptation moyen : 19.80%
Taux d'acceptation moyen : 19.81%
Taux d'acceptation moyen : 19.73%
Taux d'acceptation moyen : 19.59%
Taux d'acceptation moyen : 20.35%
Taux d'acceptation moyen : 20.36%
Taux d'acceptation moyen : 19.53%
Taux d'acceptation moyen : 19.92%
Taux d'acceptation moyen : 19.34%
Taux d'acceptation moyen : 19.16%
Taux d'acceptation moyen : 19.27%
Taux d'acceptation moyen : 19.56%
Taux d'acceptation moyen : 19.17%
Taux d'acceptation moyen : 19.20%
Taux d'acceptation moyen : 19.71%
Taux d'acceptation moyen : 19.55%
Taux d'acceptation moyen : 19.54%
Taux d'acceptation moyen : 19.39%
Taux d'acceptation moyen : 19.72%
Taux d'acceptation moyen : 19.75%
Taux d'acceptation moyen : 19.49%
Taux d'acceptation moyen : 19.87%
Taux d'acceptation moyen : 20.04%
Taux d'acceptation moyen : 19.70%
Taux d'acceptation moyen : 19.98%
Taux d'acceptation moyen : 19.83%
Taux d'acceptation moyen : 19.42%
Taux d'acceptation moyen : 19.68%
Taux d'acceptation moyen : 19.98%
Taux d'acceptation moyen : 19.60%
Taux d'acceptation moyen : 19.87%
Taux d'acceptation moyen : 20.18%
Taux d'acceptation moyen : 20.25%
Taux d'acceptation moyen : 19.38%
Taux d'acceptation moyen : 19.49%
Taux d'acceptation moyen : 19.34%
Taux d'acceptation moyen : 19.82%
Taux d'acceptation moyen : 19.54%
Taux d'acceptation moyen : 19.41%
Taux d'acceptation moyen : 20.35%
Taux d'acceptation moyen : 19.64%

Taux d'acceptation moyen : 19.42%
Taux d'acceptation moyen : 19.53%
Taux d'acceptation moyen : 19.76%
Taux d'acceptation moyen : 19.77%
Taux d'acceptation moyen : 19.97%
Taux d'acceptation moyen : 19.48%
Taux d'acceptation moyen : 19.66%
Taux d'acceptation moyen : 20.21%
Taux d'acceptation moyen : 19.69%
Taux d'acceptation moyen : 20.12%
Taux d'acceptation moyen : 19.94%
Taux d'acceptation moyen : 19.74%
Taux d'acceptation moyen : 19.48%
Taux d'acceptation moyen : 19.27%
Taux d'acceptation moyen : 18.88%
Taux d'acceptation moyen : 19.19%
Taux d'acceptation moyen : 19.23%
Taux d'acceptation moyen : 20.03%
Taux d'acceptation moyen : 19.02%
Taux d'acceptation moyen : 19.29%
Taux d'acceptation moyen : 19.55%
Taux d'acceptation moyen : 19.70%
Taux d'acceptation moyen : 19.74%
Taux d'acceptation moyen : 20.00%
Taux d'acceptation moyen : 19.98%
Taux d'acceptation moyen : 19.70%
Taux d'acceptation moyen : 19.80%
Taux d'acceptation moyen : 19.73%
Taux d'acceptation moyen : 19.40%
Taux d'acceptation moyen : 19.56%
Taux d'acceptation moyen : 19.04%
Taux d'acceptation moyen : 19.18%
Taux d'acceptation moyen : 19.59%
Taux d'acceptation moyen : 19.47%
Taux d'acceptation moyen : 19.36%
Taux d'acceptation moyen : 19.30%
Taux d'acceptation moyen : 19.27%
Taux d'acceptation moyen : 19.21%
Taux d'acceptation moyen : 19.91%
Taux d'acceptation moyen : 19.62%
Taux d'acceptation moyen : 19.53%
Taux d'acceptation moyen : 19.64%
Taux d'acceptation moyen : 19.96%
Taux d'acceptation moyen : 19.92%
Taux d'acceptation moyen : 19.89%
Taux d'acceptation moyen : 20.22%
Taux d'acceptation moyen : 19.89%
Taux d'acceptation moyen : 19.34%

Taux d'acceptation moyen : 19.32%
Taux d'acceptation moyen : 19.40%
Taux d'acceptation moyen : 19.85%
Taux d'acceptation moyen : 19.97%
Taux d'acceptation moyen : 18.90%
Taux d'acceptation moyen : 19.58%
Taux d'acceptation moyen : 19.81%
Taux d'acceptation moyen : 19.91%
Taux d'acceptation moyen : 19.11%
Taux d'acceptation moyen : 19.23%
Taux d'acceptation moyen : 19.27%
Taux d'acceptation moyen : 19.64%
Taux d'acceptation moyen : 19.76%
Taux d'acceptation moyen : 20.00%
Taux d'acceptation moyen : 18.85%
Taux d'acceptation moyen : 19.16%
Taux d'acceptation moyen : 19.16%
Taux d'acceptation moyen : 19.63%
Taux d'acceptation moyen : 19.76%
Taux d'acceptation moyen : 19.57%
Taux d'acceptation moyen : 19.48%
Taux d'acceptation moyen : 19.61%
Taux d'acceptation moyen : 19.51%
Taux d'acceptation moyen : 19.49%
Taux d'acceptation moyen : 19.27%
Taux d'acceptation moyen : 19.33%
Taux d'acceptation moyen : 19.96%
Taux d'acceptation moyen : 19.50%
Taux d'acceptation moyen : 19.46%
Taux d'acceptation moyen : 19.18%
Taux d'acceptation moyen : 19.76%
Taux d'acceptation moyen : 19.32%
Taux d'acceptation moyen : 19.19%
Taux d'acceptation moyen : 19.41%
Taux d'acceptation moyen : 19.00%
Taux d'acceptation moyen : 18.65%
Taux d'acceptation moyen : 18.88%
Taux d'acceptation moyen : 18.43%
Taux d'acceptation moyen : 18.93%
Taux d'acceptation moyen : 19.11%
Taux d'acceptation moyen : 19.64%
Taux d'acceptation moyen : 18.73%
Taux d'acceptation moyen : 19.37%
Taux d'acceptation moyen : 19.32%
Taux d'acceptation moyen : 18.88%
Taux d'acceptation moyen : 19.76%
Taux d'acceptation moyen : 18.76%
Taux d'acceptation moyen : 19.09%

Taux d'acceptation moyen : 19.60%
Taux d'acceptation moyen : 19.57%
Taux d'acceptation moyen : 19.45%
Taux d'acceptation moyen : 19.10%
Taux d'acceptation moyen : 18.88%
Taux d'acceptation moyen : 19.30%
Taux d'acceptation moyen : 19.15%
Taux d'acceptation moyen : 19.15%
Taux d'acceptation moyen : 19.53%
Taux d'acceptation moyen : 19.66%
Taux d'acceptation moyen : 18.72%
Taux d'acceptation moyen : 19.52%
Taux d'acceptation moyen : 19.18%
Taux d'acceptation moyen : 18.96%
Taux d'acceptation moyen : 19.26%
Taux d'acceptation moyen : 19.19%
Taux d'acceptation moyen : 18.76%
Taux d'acceptation moyen : 18.83%
Taux d'acceptation moyen : 18.59%
Taux d'acceptation moyen : 18.70%
Taux d'acceptation moyen : 18.66%
Taux d'acceptation moyen : 18.99%
Taux d'acceptation moyen : 19.05%
Taux d'acceptation moyen : 19.11%
Taux d'acceptation moyen : 18.14%
Taux d'acceptation moyen : 18.74%
Taux d'acceptation moyen : 18.60%
Taux d'acceptation moyen : 18.94%
Taux d'acceptation moyen : 18.38%
Taux d'acceptation moyen : 18.92%
Taux d'acceptation moyen : 19.15%
Taux d'acceptation moyen : 18.37%
Taux d'acceptation moyen : 18.82%
Taux d'acceptation moyen : 18.91%
Taux d'acceptation moyen : 18.37%
Taux d'acceptation moyen : 18.99%
Taux d'acceptation moyen : 18.68%
Taux d'acceptation moyen : 18.71%
Taux d'acceptation moyen : 18.54%
Taux d'acceptation moyen : 19.23%
Taux d'acceptation moyen : 18.72%
Taux d'acceptation moyen : 19.06%
Taux d'acceptation moyen : 18.90%
Taux d'acceptation moyen : 18.84%
Taux d'acceptation moyen : 18.94%
Taux d'acceptation moyen : 18.59%
Taux d'acceptation moyen : 18.98%
Taux d'acceptation moyen : 18.68%

Taux d'acceptation moyen : 19.12%
 Taux d'acceptation moyen : 18.67%
 Taux d'acceptation moyen : 18.73%
 Taux d'acceptation moyen : 18.65%
 Taux d'acceptation moyen : 18.74%
 Taux d'acceptation moyen : 18.99%
 Taux d'acceptation moyen : 19.17%
 Taux d'acceptation moyen : 18.92%
 Taux d'acceptation moyen : 18.81%
 Taux d'acceptation moyen : 19.44%
 Taux d'acceptation moyen : 19.47%
 Taux d'acceptation moyen : 18.33%
 Taux d'acceptation moyen : 19.37%
 Taux d'acceptation moyen : 19.04%
 Taux d'acceptation moyen : 19.20%
 Taux d'acceptation moyen : 19.51%
 Taux d'acceptation moyen : 18.98%
 Taux d'acceptation moyen : 19.35%
 Taux d'acceptation moyen : 18.90%
 Taux d'acceptation moyen : 19.39%
 Taux d'acceptation moyen : 19.28%
 Taux d'acceptation moyen : 18.75%
 Taux d'acceptation moyen : 19.32%
 Taux d'acceptation moyen : 19.30%
 Taux d'acceptation moyen : 18.84%
 Taux d'acceptation moyen : 18.92%
 Taux d'acceptation moyen : 19.10%
 Taux d'acceptation moyen : 18.58%
 Taux d'acceptation moyen : 19.20%
 Taux d'acceptation moyen : 18.65%
 Taux d'acceptation moyen : 19.14%
 Taux d'acceptation moyen : 18.65%
 Taux d'acceptation moyen : 19.13%
 Taux d'acceptation moyen : 19.14%
 Taux d'acceptation moyen : 18.47%
 [0.00207217 -0.11646004 -0.18984779 0.65151591 -0.01838864]
 Taux d'acceptation moyen : 18.82%
 Taux d'acceptation moyen : 18.74%
 Taux d'acceptation moyen : 18.98%
 Taux d'acceptation moyen : 18.80%
 Taux d'acceptation moyen : 18.75%
 Taux d'acceptation moyen : 19.18%
 Taux d'acceptation moyen : 18.81%
 Taux d'acceptation moyen : 19.17%
 Taux d'acceptation moyen : 19.11%
 Taux d'acceptation moyen : 18.53%
 Taux d'acceptation moyen : 18.63%
 Taux d'acceptation moyen : 18.45%

Taux d'acceptation moyen : 19.15%
Taux d'acceptation moyen : 18.52%
Taux d'acceptation moyen : 18.45%
Taux d'acceptation moyen : 18.34%
Taux d'acceptation moyen : 18.08%
Taux d'acceptation moyen : 18.67%
Taux d'acceptation moyen : 18.70%
Taux d'acceptation moyen : 18.79%
Taux d'acceptation moyen : 18.73%
Taux d'acceptation moyen : 18.89%
Taux d'acceptation moyen : 18.35%
Taux d'acceptation moyen : 18.58%
Taux d'acceptation moyen : 18.94%
Taux d'acceptation moyen : 19.51%
Taux d'acceptation moyen : 18.81%
Taux d'acceptation moyen : 18.78%
Taux d'acceptation moyen : 18.60%
Taux d'acceptation moyen : 18.47%
Taux d'acceptation moyen : 18.67%
Taux d'acceptation moyen : 18.51%
Taux d'acceptation moyen : 18.32%
Taux d'acceptation moyen : 18.38%
Taux d'acceptation moyen : 19.11%
Taux d'acceptation moyen : 19.00%
Taux d'acceptation moyen : 18.93%
Taux d'acceptation moyen : 18.47%
Taux d'acceptation moyen : 18.92%
Taux d'acceptation moyen : 18.87%
Taux d'acceptation moyen : 18.50%
Taux d'acceptation moyen : 18.31%
Taux d'acceptation moyen : 18.32%
Taux d'acceptation moyen : 19.23%
Taux d'acceptation moyen : 18.32%
Taux d'acceptation moyen : 18.67%
Taux d'acceptation moyen : 18.73%
Taux d'acceptation moyen : 18.55%
Taux d'acceptation moyen : 19.07%
Taux d'acceptation moyen : 18.03%
Taux d'acceptation moyen : 18.51%
Taux d'acceptation moyen : 18.22%
Taux d'acceptation moyen : 18.05%
Taux d'acceptation moyen : 18.34%
Taux d'acceptation moyen : 18.39%
Taux d'acceptation moyen : 17.98%
Taux d'acceptation moyen : 18.14%
Taux d'acceptation moyen : 18.30%
Taux d'acceptation moyen : 18.30%
Taux d'acceptation moyen : 18.25%

Taux d'acceptation moyen : 18.23%
 Taux d'acceptation moyen : 18.24%
 Taux d'acceptation moyen : 17.71%
 Taux d'acceptation moyen : 17.85%
 Taux d'acceptation moyen : 17.78%
 Taux d'acceptation moyen : 17.64%
 Taux d'acceptation moyen : 18.43%
 Taux d'acceptation moyen : 18.34%
 Taux d'acceptation moyen : 18.28%
 Taux d'acceptation moyen : 17.72%
 Taux d'acceptation moyen : 17.97%
 Taux d'acceptation moyen : 17.90%
 Taux d'acceptation moyen : 17.45%
 Taux d'acceptation moyen : 18.14%
 Taux d'acceptation moyen : 18.00%
 Taux d'acceptation moyen : 18.16%
 Taux d'acceptation moyen : 17.89%
 Taux d'acceptation moyen : 18.09%
 Taux d'acceptation moyen : 17.84%
 Taux d'acceptation moyen : 18.02%
 Taux d'acceptation moyen : 17.91%
 Taux d'acceptation moyen : 17.99%
 Taux d'acceptation moyen : 18.31%
 Taux d'acceptation moyen : 18.39%
 Taux d'acceptation moyen : 18.35%
 Taux d'acceptation moyen : 17.78%
 Taux d'acceptation moyen : 18.34%
 Taux d'acceptation moyen : 18.08%
 Taux d'acceptation moyen : 17.84%
 Taux d'acceptation moyen : 18.36%
 Taux d'acceptation moyen : 18.30%
 Taux d'acceptation moyen : 18.35%
 Taux d'acceptation moyen : 18.07%
 Taux d'acceptation moyen : 17.99%
 Taux d'acceptation moyen : 18.01%
 Taux d'acceptation moyen : 18.06%
 Taux d'acceptation moyen : 18.12%
 Taux d'acceptation moyen : 18.08%
 Taux d'acceptation moyen : 18.22%
 Taux d'acceptation moyen : 18.65%
 [0.01660448 -0.07467544 -0.24244111 0.58238864 -0.00249324]
 Taux d'acceptation moyen : 18.83%
 Taux d'acceptation moyen : 18.18%
 Taux d'acceptation moyen : 18.20%
 Taux d'acceptation moyen : 18.16%
 Taux d'acceptation moyen : 18.64%
 Taux d'acceptation moyen : 18.72%
 Taux d'acceptation moyen : 17.85%

Taux d'acceptation moyen : 18.60%
Taux d'acceptation moyen : 17.99%
Taux d'acceptation moyen : 18.76%
Taux d'acceptation moyen : 18.20%
Taux d'acceptation moyen : 18.70%
Taux d'acceptation moyen : 18.52%
Taux d'acceptation moyen : 18.83%
Taux d'acceptation moyen : 18.62%
Taux d'acceptation moyen : 18.52%
Taux d'acceptation moyen : 18.62%
Taux d'acceptation moyen : 18.41%
Taux d'acceptation moyen : 19.02%
Taux d'acceptation moyen : 18.67%
Taux d'acceptation moyen : 19.24%
Taux d'acceptation moyen : 19.10%
Taux d'acceptation moyen : 18.46%
Taux d'acceptation moyen : 18.56%
Taux d'acceptation moyen : 18.46%
Taux d'acceptation moyen : 18.92%
Taux d'acceptation moyen : 18.65%
Taux d'acceptation moyen : 18.83%
Taux d'acceptation moyen : 18.85%
Taux d'acceptation moyen : 18.20%
Taux d'acceptation moyen : 18.88%
Taux d'acceptation moyen : 18.34%
Taux d'acceptation moyen : 17.96%
Taux d'acceptation moyen : 18.64%
Taux d'acceptation moyen : 18.18%
Taux d'acceptation moyen : 18.38%
Taux d'acceptation moyen : 18.68%
Taux d'acceptation moyen : 18.69%
Taux d'acceptation moyen : 18.61%
Taux d'acceptation moyen : 18.67%
Taux d'acceptation moyen : 18.68%
Taux d'acceptation moyen : 18.75%
Taux d'acceptation moyen : 18.99%
Taux d'acceptation moyen : 19.05%
Taux d'acceptation moyen : 18.81%
Taux d'acceptation moyen : 18.77%
Taux d'acceptation moyen : 19.10%
Taux d'acceptation moyen : 18.77%
Taux d'acceptation moyen : 18.41%
Taux d'acceptation moyen : 19.13%
Taux d'acceptation moyen : 18.86%
Taux d'acceptation moyen : 18.62%
Taux d'acceptation moyen : 18.52%
Taux d'acceptation moyen : 18.62%
Taux d'acceptation moyen : 18.99%

Taux d'acceptation moyen : 18.71%
 Taux d'acceptation moyen : 18.82%
 Taux d'acceptation moyen : 18.30%
 Taux d'acceptation moyen : 18.83%
 Taux d'acceptation moyen : 18.35%
 Taux d'acceptation moyen : 18.41%
 Taux d'acceptation moyen : 18.78%
 Taux d'acceptation moyen : 18.29%
 Taux d'acceptation moyen : 18.58%
 Taux d'acceptation moyen : 18.86%
 Taux d'acceptation moyen : 19.08%
 Taux d'acceptation moyen : 19.19%
 Taux d'acceptation moyen : 18.90%
 Taux d'acceptation moyen : 18.70%
 Taux d'acceptation moyen : 19.45%
 Taux d'acceptation moyen : 18.52%
 Taux d'acceptation moyen : 19.27%
 Taux d'acceptation moyen : 18.68%
 Taux d'acceptation moyen : 18.94%
 Taux d'acceptation moyen : 19.41%
 Taux d'acceptation moyen : 18.81%
 Taux d'acceptation moyen : 18.88%
 Taux d'acceptation moyen : 18.86%
 Taux d'acceptation moyen : 18.52%
 Taux d'acceptation moyen : 19.20%
 Taux d'acceptation moyen : 18.50%
 Taux d'acceptation moyen : 19.29%
 Taux d'acceptation moyen : 19.23%
 Taux d'acceptation moyen : 19.13%
 Taux d'acceptation moyen : 18.83%
 Taux d'acceptation moyen : 18.81%
 Taux d'acceptation moyen : 18.62%
 Taux d'acceptation moyen : 18.89%
 Taux d'acceptation moyen : 18.67%
 Taux d'acceptation moyen : 19.16%
 Taux d'acceptation moyen : 19.09%
 Taux d'acceptation moyen : 19.18%
 Taux d'acceptation moyen : 18.52%
 Taux d'acceptation moyen : 18.60%
 Taux d'acceptation moyen : 19.05%
 Taux d'acceptation moyen : 19.52%
 Taux d'acceptation moyen : 19.00%
 Taux d'acceptation moyen : 18.98%
 Taux d'acceptation moyen : 18.86%
 Taux d'acceptation moyen : 19.05%
 [0.00934904 -0.11695477 -0.11131799 0.45645084 -0.00582209]
 Taux d'acceptation moyen : 18.43%
 Taux d'acceptation moyen : 19.03%

Taux d'acceptation moyen : 18.98%
Taux d'acceptation moyen : 18.93%
Taux d'acceptation moyen : 19.03%
Taux d'acceptation moyen : 18.57%
Taux d'acceptation moyen : 19.41%
Taux d'acceptation moyen : 18.69%
Taux d'acceptation moyen : 18.45%
Taux d'acceptation moyen : 18.89%
Taux d'acceptation moyen : 19.00%
Taux d'acceptation moyen : 19.14%
Taux d'acceptation moyen : 18.33%
Taux d'acceptation moyen : 18.62%
Taux d'acceptation moyen : 18.28%
Taux d'acceptation moyen : 18.38%
Taux d'acceptation moyen : 18.18%
Taux d'acceptation moyen : 17.76%
Taux d'acceptation moyen : 18.26%
Taux d'acceptation moyen : 18.53%
Taux d'acceptation moyen : 18.05%
Taux d'acceptation moyen : 17.94%
Taux d'acceptation moyen : 18.18%
Taux d'acceptation moyen : 18.43%
Taux d'acceptation moyen : 18.03%
Taux d'acceptation moyen : 18.46%
Taux d'acceptation moyen : 18.28%
Taux d'acceptation moyen : 18.38%
Taux d'acceptation moyen : 18.16%
Taux d'acceptation moyen : 18.17%
Taux d'acceptation moyen : 18.62%
Taux d'acceptation moyen : 18.92%
Taux d'acceptation moyen : 18.67%
Taux d'acceptation moyen : 18.20%
Taux d'acceptation moyen : 18.47%
Taux d'acceptation moyen : 18.55%
Taux d'acceptation moyen : 18.43%
Taux d'acceptation moyen : 17.73%
Taux d'acceptation moyen : 18.33%
Taux d'acceptation moyen : 18.49%
Taux d'acceptation moyen : 18.45%
Taux d'acceptation moyen : 18.06%
Taux d'acceptation moyen : 18.31%
Taux d'acceptation moyen : 18.46%
Taux d'acceptation moyen : 18.67%
Taux d'acceptation moyen : 18.36%
Taux d'acceptation moyen : 18.25%
Taux d'acceptation moyen : 18.88%
Taux d'acceptation moyen : 18.80%
Taux d'acceptation moyen : 18.78%

Taux d'acceptation moyen : 18.37%
Taux d'acceptation moyen : 18.91%
Taux d'acceptation moyen : 18.33%
Taux d'acceptation moyen : 18.98%
Taux d'acceptation moyen : 18.83%
Taux d'acceptation moyen : 19.05%
Taux d'acceptation moyen : 19.02%
Taux d'acceptation moyen : 18.63%
Taux d'acceptation moyen : 19.45%
Taux d'acceptation moyen : 18.67%
Taux d'acceptation moyen : 18.61%
Taux d'acceptation moyen : 18.54%
Taux d'acceptation moyen : 18.66%
Taux d'acceptation moyen : 19.45%
Taux d'acceptation moyen : 19.31%
Taux d'acceptation moyen : 19.02%
Taux d'acceptation moyen : 19.38%
Taux d'acceptation moyen : 19.29%
Taux d'acceptation moyen : 19.47%
Taux d'acceptation moyen : 18.80%
Taux d'acceptation moyen : 19.19%
Taux d'acceptation moyen : 19.56%
Taux d'acceptation moyen : 19.28%
Taux d'acceptation moyen : 19.31%
Taux d'acceptation moyen : 19.41%
Taux d'acceptation moyen : 19.14%
Taux d'acceptation moyen : 19.61%
Taux d'acceptation moyen : 19.62%
Taux d'acceptation moyen : 19.68%
Taux d'acceptation moyen : 19.22%
Taux d'acceptation moyen : 19.17%
Taux d'acceptation moyen : 18.90%
Taux d'acceptation moyen : 19.03%
Taux d'acceptation moyen : 19.82%
Taux d'acceptation moyen : 18.86%
Taux d'acceptation moyen : 19.43%
Taux d'acceptation moyen : 19.41%
Taux d'acceptation moyen : 19.25%
Taux d'acceptation moyen : 19.14%
Taux d'acceptation moyen : 19.88%
Taux d'acceptation moyen : 19.68%
Taux d'acceptation moyen : 19.11%
Taux d'acceptation moyen : 19.72%
Taux d'acceptation moyen : 19.31%
Taux d'acceptation moyen : 19.84%
Taux d'acceptation moyen : 19.78%
Taux d'acceptation moyen : 19.61%
Taux d'acceptation moyen : 19.35%

Taux d'acceptation moyen : 19.36%
 Taux d'acceptation moyen : 18.99%
 [-0.03097628 -0.13908938 -0.07230021 0.48971966 0.00631809]
 Taux d'acceptation moyen : 18.98%
 Taux d'acceptation moyen : 19.23%
 Taux d'acceptation moyen : 19.58%
 Taux d'acceptation moyen : 19.73%
 Taux d'acceptation moyen : 18.77%
 Taux d'acceptation moyen : 19.13%
 Taux d'acceptation moyen : 19.44%
 Taux d'acceptation moyen : 18.82%
 Taux d'acceptation moyen : 19.48%
 Taux d'acceptation moyen : 19.28%
 Taux d'acceptation moyen : 19.21%
 Taux d'acceptation moyen : 19.24%
 Taux d'acceptation moyen : 19.38%
 Taux d'acceptation moyen : 19.44%
 Taux d'acceptation moyen : 19.33%
 Taux d'acceptation moyen : 19.39%
 Taux d'acceptation moyen : 19.63%
 Taux d'acceptation moyen : 19.54%
 Taux d'acceptation moyen : 19.66%
 Taux d'acceptation moyen : 19.06%
 Taux d'acceptation moyen : 19.26%
 Taux d'acceptation moyen : 19.65%
 Taux d'acceptation moyen : 19.14%
 Taux d'acceptation moyen : 19.39%
 Taux d'acceptation moyen : 19.36%
 Taux d'acceptation moyen : 18.98%
 Taux d'acceptation moyen : 19.62%
 Taux d'acceptation moyen : 19.70%
 Taux d'acceptation moyen : 19.84%
 Taux d'acceptation moyen : 19.44%
 Taux d'acceptation moyen : 19.09%
 Taux d'acceptation moyen : 19.94%
 Taux d'acceptation moyen : 19.33%
 Taux d'acceptation moyen : 19.72%
 Taux d'acceptation moyen : 19.45%
 Taux d'acceptation moyen : 19.74%
 Taux d'acceptation moyen : 19.02%
 Taux d'acceptation moyen : 19.18%
 Taux d'acceptation moyen : 19.49%
 Taux d'acceptation moyen : 19.32%
 Taux d'acceptation moyen : 19.70%
 Taux d'acceptation moyen : 19.34%
 Taux d'acceptation moyen : 19.09%
 Taux d'acceptation moyen : 18.87%
 Taux d'acceptation moyen : 19.14%

Taux d'acceptation moyen : 19.39%
Taux d'acceptation moyen : 19.05%
Taux d'acceptation moyen : 19.48%
Taux d'acceptation moyen : 19.21%
Taux d'acceptation moyen : 19.42%
Taux d'acceptation moyen : 19.07%
Taux d'acceptation moyen : 19.18%
Taux d'acceptation moyen : 19.36%
Taux d'acceptation moyen : 19.02%
Taux d'acceptation moyen : 18.97%
Taux d'acceptation moyen : 19.17%
Taux d'acceptation moyen : 19.22%
Taux d'acceptation moyen : 18.78%
Taux d'acceptation moyen : 18.74%
Taux d'acceptation moyen : 18.96%
Taux d'acceptation moyen : 19.12%
Taux d'acceptation moyen : 19.16%
Taux d'acceptation moyen : 19.29%
Taux d'acceptation moyen : 19.29%
Taux d'acceptation moyen : 19.13%
Taux d'acceptation moyen : 18.62%
Taux d'acceptation moyen : 18.52%
Taux d'acceptation moyen : 19.04%
Taux d'acceptation moyen : 18.71%
Taux d'acceptation moyen : 18.80%
Taux d'acceptation moyen : 19.29%
Taux d'acceptation moyen : 19.35%
Taux d'acceptation moyen : 18.82%
Taux d'acceptation moyen : 18.61%
Taux d'acceptation moyen : 19.30%
Taux d'acceptation moyen : 18.99%
Taux d'acceptation moyen : 19.23%
Taux d'acceptation moyen : 18.45%
Taux d'acceptation moyen : 19.12%
Taux d'acceptation moyen : 19.55%
Taux d'acceptation moyen : 19.12%
Taux d'acceptation moyen : 18.78%
Taux d'acceptation moyen : 18.77%
Taux d'acceptation moyen : 19.42%
Taux d'acceptation moyen : 18.79%
Taux d'acceptation moyen : 19.08%
Taux d'acceptation moyen : 19.45%
Taux d'acceptation moyen : 19.45%
Taux d'acceptation moyen : 18.69%
Taux d'acceptation moyen : 19.08%
Taux d'acceptation moyen : 19.54%
Taux d'acceptation moyen : 19.28%
Taux d'acceptation moyen : 19.27%

```

Taux d'acceptation moyen : 19.24%
Taux d'acceptation moyen : 19.44%
Taux d'acceptation moyen : 19.39%
Taux d'acceptation moyen : 18.91%
Taux d'acceptation moyen : 19.68%
Taux d'acceptation moyen : 19.78%
--- 3618.164031982422 seconds ---

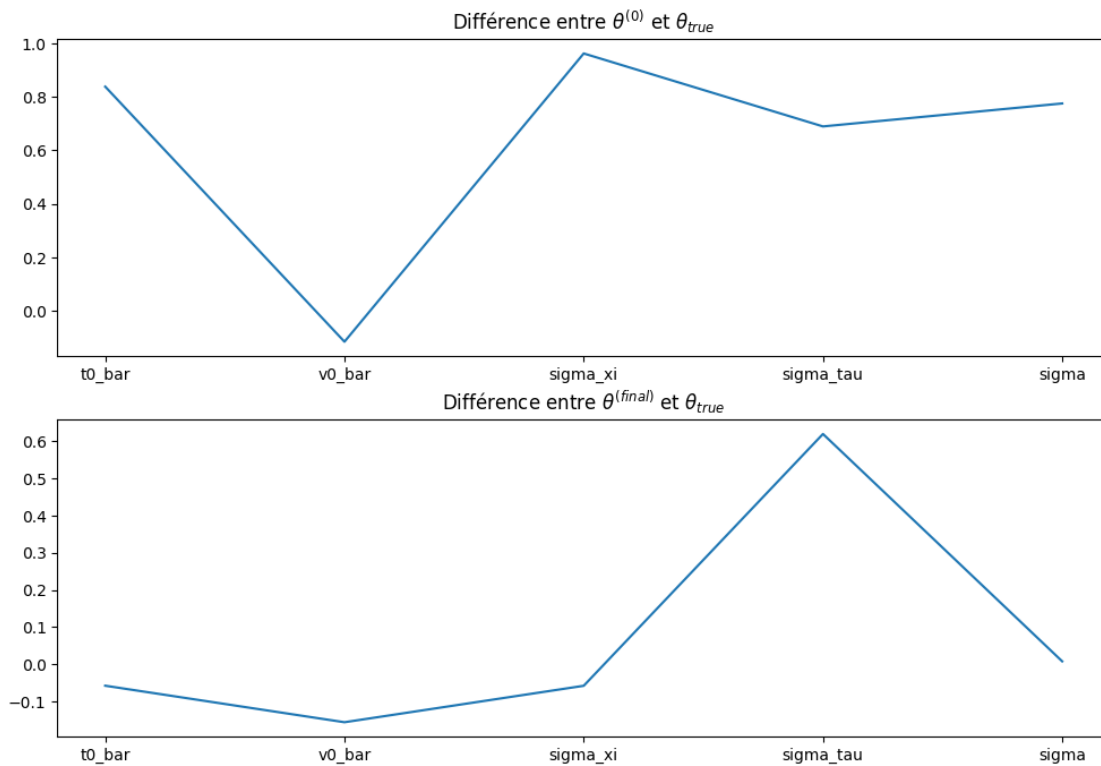
```

Dès 600 itérations le MCMC-SAEM semble plus ou moins avoir convergé vers les bons theta, puis il commence à osciller. Ici sur le sigma_tau.

```

[39]: fig, (ax1,ax2)=plt.subplots(2)
ax1.plot(THETA.keys(),np.array(list(THETA.values()))- theta_values)
ax1.set_title(r'Différence entre  $\theta^{(0)}$  et  $\theta_{true}$ ')
ax2.plot(THETA.keys(),np.array(list(THETA.values()))- np.
    ↳array(list(THETA_FINAL_MCMC.values()))
ax2.set_title(r'Différence entre  $\theta^{(final)}$  et  $\theta_{true}$ ')
plt.show()

```



Avec une variance σ_{prop} unique pour chaque variable, l'échantillonneur a du mal à converger précisément sur toutes les valeurs, notamment les σ_ξ et σ_τ, σ qui ont une forte variance.

2.5 HMwG-SAEM – Hasting-Metropolis within Gibbs sampler

```
[40]: import multiprocessing
from multiprocessing import Pool
import concurrent.futures
```

2.6 Questions 5-6-7

```
[41]: #On crée une fonction qui répond aux questions 5-6+/-7
#Prend en entrée nos (z_i) ou (z_pop) à sampler + un vecteur z dans  $R^{2N+2}$  qui
→ nous donne les valeurs restantes (ie z_i ou z_pop)

def HMwG_sampler(Y,z,z_true,sigma_prop,t0_bar=np.
    →array(THETA['t0_bar']),v0_bar=np.array(THETA['v0_bar']),
        sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
        ↵
    →sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma'],
        ↵
    →pi=log_posterior_simple,max_iter=1000,burnin=None,verbose=False):
    burnin=0 if None else burnin
    N=z.shape[0]
    z_list=[]
    accept=0
    total_accept=0
    #Sigma=sigma_prop*np.eye(z.shape[0])
    zk=z
    if z_true.shape==z.shape:
        raise ValueError('z_true doit être un vecteur complet, z doit être la ↵
    →partie à échantillonner')

    if N==2:
        def log_post_Gibbs(zk,zi,l):
            zk[l]=zi
            return pi(Y=Y, z=np.concatenate([zk, z_true[2:]]),
                t0_bar=t0_bar,v0_bar=v0_bar,
                sigma_t0=sigma_t0,sigma_v0=sigma_v0,
                sigma_xi=sigma_xi,sigma_tau=sigma_tau,sigma=sigma)
    else:
        def log_post_Gibbs(zk,zi,l):
            zk[l]=zi
            return pi(Y=Y, z=np.concatenate([z_true[:2],zk]),
                t0_bar=t0_bar,v0_bar=v0_bar,
                sigma_t0=sigma_t0,sigma_v0=sigma_v0,
                sigma_xi=sigma_xi,sigma_tau=sigma_tau,sigma=sigma)

    #Cdt à z_pop (resp. z_i), les z_i (resp. z_pop) sont indépendants donc on ↵
    →peut tout paralléliser
```

```

def update_indep(l):
    zi_list=[]
    current=z_k[l]
    accept=0
    for k in range(max_iter):
        z_star=np.random.normal(loc=current,scale=np.sqrt(sigma_prop))
        l_zstar=log_post_Gibbs(z_k,z_star,l)
        l_z=log_post_Gibbs(z_k,current,l)

        #On calcule tout dans l'espace des log-probs
        prob_choose=0 if 0<= l_zstar-l_z else l_zstar-l_z
        u=np.random.rand()
        if prob_choose>=np.log(u):
            zi_list.append(z_star)
            current=z_star
            if k>=burnin:
                accept+=1
        else:
            zi_list.append(current)
    idx=np.random.randint(1, int(.5*(max_iter-burnin)))
    z_sampled=zi_list[-idx]
    if verbose:
        taux=accept/max_iter
        return [z_sampled, taux]
    else:
        return z_sampled

#Partie MH sampling parallélisée
with concurrent.futures.ThreadPoolExecutor(max_workers=multiprocessing.
→cpu_count()) as executor:
    generator = executor.map(update_indep, range(N))
    #executor.shutdown(wait=True)
    results=list(generator)
    if verbose:
        sampled=[r[0] for r in results]
        taux_acceptation=[r[1] for r in results]
        for l in range(len(taux_acceptation)):
            print(f'Taux d\'acceptation moyen pour {l=} : {taux_acceptation[l]:.
→2%}')
        print(f'Taux moyen = {np.mean(taux_acceptation):.2%}')
    else:
        sampled=[r for r in results]
    return sampled

```

```

[44]: z_rand=np.random.rand(2*N)+15*np.ones(2*N)
z_true=np.concatenate([[LATENT['t0']], [LATENT['v0']], LATENT['xi'], LATENT['tau']])
burnin=2000

```

```

sigma_prop=9*1e-3
z_i_sampled =HMwG_sampler(Y,z_rand,z_true,sigma_prop=sigma_prop,
                           t0_bar=np.array(THETA_TRUE['t0_bar']),v0_bar=np.
↳array(THETA_TRUE['v0_bar']),
                           sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
                           sigma_xi=THETA_TRUE['sigma_xi'],
↳sigma_tau=THETA_TRUE['sigma_tau'],sigma=THETA_TRUE['sigma'],
                           max_iter=2500,burnin=burnin,verbose=True)

```

```

Taux d'acceptation moyen pour l=0 : 18.44%
Taux d'acceptation moyen pour l=1 : 19.04%
Taux d'acceptation moyen pour l=2 : 17.56%
Taux d'acceptation moyen pour l=3 : 17.60%
Taux d'acceptation moyen pour l=4 : 19.48%
Taux d'acceptation moyen pour l=5 : 18.48%
Taux d'acceptation moyen pour l=6 : 17.52%
Taux d'acceptation moyen pour l=7 : 17.60%
Taux d'acceptation moyen pour l=8 : 17.16%
Taux d'acceptation moyen pour l=9 : 16.96%
Taux d'acceptation moyen pour l=10 : 17.76%
Taux d'acceptation moyen pour l=11 : 17.12%
Taux d'acceptation moyen pour l=12 : 18.52%
Taux d'acceptation moyen pour l=13 : 16.60%
Taux d'acceptation moyen pour l=14 : 18.92%
Taux d'acceptation moyen pour l=15 : 16.40%
Taux d'acceptation moyen pour l=16 : 19.64%
Taux d'acceptation moyen pour l=17 : 19.72%
Taux d'acceptation moyen pour l=18 : 18.24%
Taux d'acceptation moyen pour l=19 : 17.72%
Taux d'acceptation moyen pour l=20 : 18.28%
Taux d'acceptation moyen pour l=21 : 17.92%
Taux d'acceptation moyen pour l=22 : 16.56%
Taux d'acceptation moyen pour l=23 : 17.24%
Taux d'acceptation moyen pour l=24 : 16.48%
Taux d'acceptation moyen pour l=25 : 16.24%
Taux d'acceptation moyen pour l=26 : 17.00%
Taux d'acceptation moyen pour l=27 : 16.92%
Taux d'acceptation moyen pour l=28 : 16.16%
Taux d'acceptation moyen pour l=29 : 15.64%
Taux d'acceptation moyen pour l=30 : 14.08%
Taux d'acceptation moyen pour l=31 : 16.32%
Taux d'acceptation moyen pour l=32 : 19.40%
Taux d'acceptation moyen pour l=33 : 19.40%
Taux d'acceptation moyen pour l=34 : 18.40%
Taux d'acceptation moyen pour l=35 : 18.08%
Taux d'acceptation moyen pour l=36 : 17.08%
Taux d'acceptation moyen pour l=37 : 18.52%

```


Taux d'acceptation moyen pour l=38 : 17.28%
 Taux d'acceptation moyen pour l=39 : 18.56%
 Taux d'acceptation moyen pour l=40 : 16.36%
 Taux d'acceptation moyen pour l=41 : 16.04%
 Taux d'acceptation moyen pour l=42 : 17.84%
 Taux d'acceptation moyen pour l=43 : 16.04%
 Taux d'acceptation moyen pour l=44 : 16.68%
 Taux d'acceptation moyen pour l=45 : 15.28%
 Taux d'acceptation moyen pour l=46 : 16.64%
 Taux d'acceptation moyen pour l=47 : 12.92%
 Taux d'acceptation moyen pour l=48 : 18.08%
 Taux d'acceptation moyen pour l=49 : 17.64%
 Taux d'acceptation moyen pour l=50 : 14.72%
 Taux d'acceptation moyen pour l=51 : 15.56%
 Taux d'acceptation moyen pour l=52 : 17.08%
 Taux d'acceptation moyen pour l=53 : 17.56%
 Taux d'acceptation moyen pour l=54 : 16.96%
 Taux d'acceptation moyen pour l=55 : 15.64%
 Taux d'acceptation moyen pour l=56 : 17.20%
 Taux d'acceptation moyen pour l=57 : 16.44%
 Taux d'acceptation moyen pour l=58 : 12.84%
 Taux d'acceptation moyen pour l=59 : 15.32%
 Taux d'acceptation moyen pour l=60 : 16.76%
 Taux d'acceptation moyen pour l=61 : 14.80%
 Taux d'acceptation moyen pour l=62 : 15.16%
 Taux d'acceptation moyen pour l=63 : 11.64%
 Taux d'acceptation moyen pour l=64 : 18.16%
 Taux d'acceptation moyen pour l=65 : 18.60%
 Taux d'acceptation moyen pour l=66 : 18.28%
 Taux d'acceptation moyen pour l=67 : 17.72%
 Taux d'acceptation moyen pour l=68 : 17.88%
 Taux d'acceptation moyen pour l=69 : 16.40%
 Taux d'acceptation moyen pour l=70 : 17.64%
 Taux d'acceptation moyen pour l=71 : 16.80%
 Taux d'acceptation moyen pour l=72 : 16.52%
 Taux d'acceptation moyen pour l=73 : 16.56%
 Taux d'acceptation moyen pour l=74 : 17.36%
 Taux d'acceptation moyen pour l=75 : 16.40%
 Taux d'acceptation moyen pour l=76 : 16.56%
 Taux d'acceptation moyen pour l=77 : 15.44%
 Taux d'acceptation moyen pour l=78 : 12.12%
 Taux d'acceptation moyen pour l=79 : 12.88%
 Taux d'acceptation moyen pour l=80 : 17.68%
 Taux d'acceptation moyen pour l=81 : 16.88%
 Taux d'acceptation moyen pour l=82 : 16.28%
 Taux d'acceptation moyen pour l=83 : 15.64%
 Taux d'acceptation moyen pour l=84 : 15.88%
 Taux d'acceptation moyen pour l=85 : 16.20%

Taux d'acceptation moyen pour l=86 : 15.00%
 Taux d'acceptation moyen pour l=87 : 14.08%
 Taux d'acceptation moyen pour l=88 : 14.00%
 Taux d'acceptation moyen pour l=89 : 16.00%
 Taux d'acceptation moyen pour l=90 : 14.76%
 Taux d'acceptation moyen pour l=91 : 15.16%
 Taux d'acceptation moyen pour l=92 : 14.44%
 Taux d'acceptation moyen pour l=93 : 13.40%
 Taux d'acceptation moyen pour l=94 : 11.68%
 Taux d'acceptation moyen pour l=95 : 11.36%
 Taux d'acceptation moyen pour l=96 : 10.88%
 Taux d'acceptation moyen pour l=97 : 11.44%
 Taux d'acceptation moyen pour l=98 : 11.64%
 Taux d'acceptation moyen pour l=99 : 10.72%
 Taux d'acceptation moyen pour l=100 : 9.20%
 Taux d'acceptation moyen pour l=101 : 10.64%
 Taux d'acceptation moyen pour l=102 : 11.24%
 Taux d'acceptation moyen pour l=103 : 10.28%
 Taux d'acceptation moyen pour l=104 : 11.56%
 Taux d'acceptation moyen pour l=105 : 10.00%
 Taux d'acceptation moyen pour l=106 : 11.36%
 Taux d'acceptation moyen pour l=107 : 11.80%
 Taux d'acceptation moyen pour l=108 : 10.08%
 Taux d'acceptation moyen pour l=109 : 9.68%
 Taux d'acceptation moyen pour l=110 : 10.44%
 Taux d'acceptation moyen pour l=111 : 12.36%
 Taux d'acceptation moyen pour l=112 : 15.52%
 Taux d'acceptation moyen pour l=113 : 14.64%
 Taux d'acceptation moyen pour l=114 : 15.28%
 Taux d'acceptation moyen pour l=115 : 14.36%
 Taux d'acceptation moyen pour l=116 : 13.20%
 Taux d'acceptation moyen pour l=117 : 13.72%
 Taux d'acceptation moyen pour l=118 : 14.12%
 Taux d'acceptation moyen pour l=119 : 15.52%
 Taux d'acceptation moyen pour l=120 : 14.16%
 Taux d'acceptation moyen pour l=121 : 14.60%
 Taux d'acceptation moyen pour l=122 : 15.44%
 Taux d'acceptation moyen pour l=123 : 15.32%
 Taux d'acceptation moyen pour l=124 : 13.56%
 Taux d'acceptation moyen pour l=125 : 14.80%
 Taux d'acceptation moyen pour l=126 : 13.56%
 Taux d'acceptation moyen pour l=127 : 13.92%
 Taux d'acceptation moyen pour l=128 : 11.76%
 Taux d'acceptation moyen pour l=129 : 11.80%
 Taux d'acceptation moyen pour l=130 : 12.68%
 Taux d'acceptation moyen pour l=131 : 13.16%
 Taux d'acceptation moyen pour l=132 : 11.88%
 Taux d'acceptation moyen pour l=133 : 12.40%

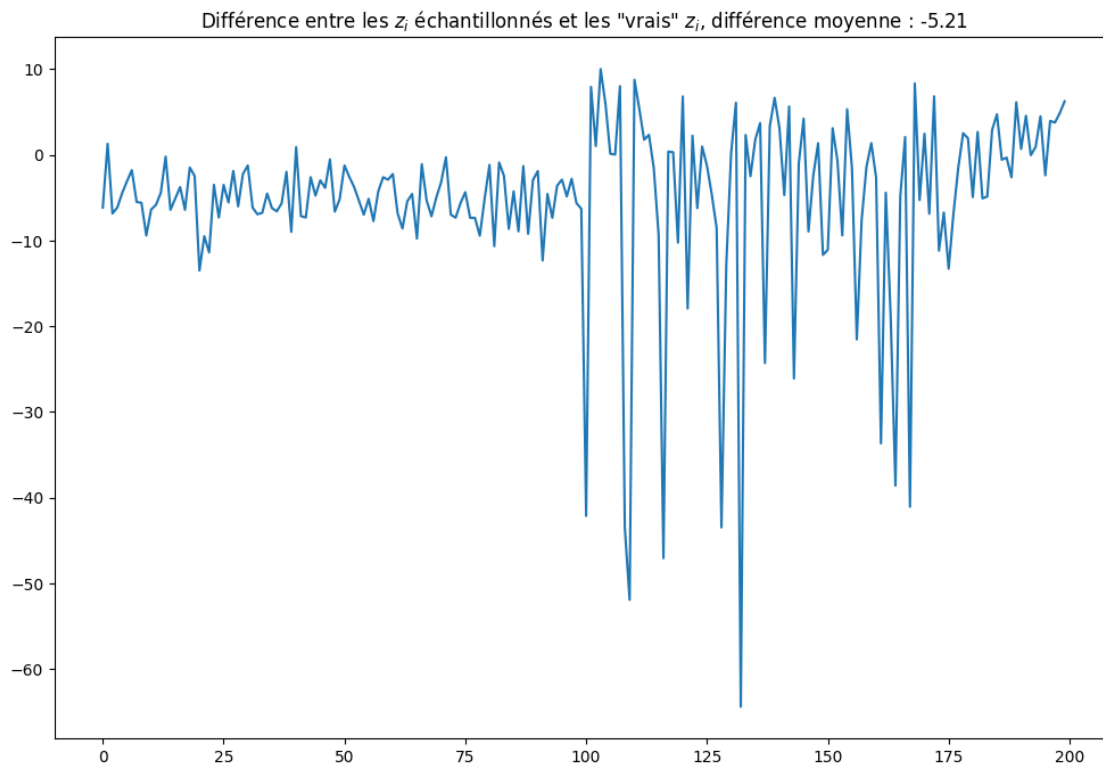
Taux d'acceptation moyen pour l=134 : 12.68%
 Taux d'acceptation moyen pour l=135 : 12.48%
 Taux d'acceptation moyen pour l=136 : 12.28%
 Taux d'acceptation moyen pour l=137 : 11.84%
 Taux d'acceptation moyen pour l=138 : 12.64%
 Taux d'acceptation moyen pour l=139 : 12.68%
 Taux d'acceptation moyen pour l=140 : 13.60%
 Taux d'acceptation moyen pour l=141 : 12.56%
 Taux d'acceptation moyen pour l=142 : 12.64%
 Taux d'acceptation moyen pour l=143 : 11.44%
 Taux d'acceptation moyen pour l=144 : 15.52%
 Taux d'acceptation moyen pour l=145 : 15.96%
 Taux d'acceptation moyen pour l=146 : 15.64%
 Taux d'acceptation moyen pour l=147 : 16.44%
 Taux d'acceptation moyen pour l=148 : 16.08%
 Taux d'acceptation moyen pour l=149 : 16.24%
 Taux d'acceptation moyen pour l=150 : 13.36%
 Taux d'acceptation moyen pour l=151 : 15.32%
 Taux d'acceptation moyen pour l=152 : 14.96%
 Taux d'acceptation moyen pour l=153 : 14.20%
 Taux d'acceptation moyen pour l=154 : 14.56%
 Taux d'acceptation moyen pour l=155 : 12.60%
 Taux d'acceptation moyen pour l=156 : 13.64%
 Taux d'acceptation moyen pour l=157 : 13.40%
 Taux d'acceptation moyen pour l=158 : 13.12%
 Taux d'acceptation moyen pour l=159 : 13.32%
 Taux d'acceptation moyen pour l=160 : 13.92%
 Taux d'acceptation moyen pour l=161 : 13.28%
 Taux d'acceptation moyen pour l=162 : 13.96%
 Taux d'acceptation moyen pour l=163 : 14.52%
 Taux d'acceptation moyen pour l=164 : 13.36%
 Taux d'acceptation moyen pour l=165 : 14.64%
 Taux d'acceptation moyen pour l=166 : 14.36%
 Taux d'acceptation moyen pour l=167 : 13.92%
 Taux d'acceptation moyen pour l=168 : 14.28%
 Taux d'acceptation moyen pour l=169 : 14.64%
 Taux d'acceptation moyen pour l=170 : 14.44%
 Taux d'acceptation moyen pour l=171 : 14.80%
 Taux d'acceptation moyen pour l=172 : 15.24%
 Taux d'acceptation moyen pour l=173 : 14.76%
 Taux d'acceptation moyen pour l=174 : 15.12%
 Taux d'acceptation moyen pour l=175 : 15.16%
 Taux d'acceptation moyen pour l=176 : 18.08%
 Taux d'acceptation moyen pour l=177 : 17.24%
 Taux d'acceptation moyen pour l=178 : 17.60%
 Taux d'acceptation moyen pour l=179 : 18.00%
 Taux d'acceptation moyen pour l=180 : 17.68%
 Taux d'acceptation moyen pour l=181 : 17.04%

Taux d'acceptation moyen pour l=182 : 17.48%
 Taux d'acceptation moyen pour l=183 : 17.44%
 Taux d'acceptation moyen pour l=184 : 17.48%
 Taux d'acceptation moyen pour l=185 : 16.76%
 Taux d'acceptation moyen pour l=186 : 17.44%
 Taux d'acceptation moyen pour l=187 : 17.32%
 Taux d'acceptation moyen pour l=188 : 17.08%
 Taux d'acceptation moyen pour l=189 : 17.20%
 Taux d'acceptation moyen pour l=190 : 16.96%
 Taux d'acceptation moyen pour l=191 : 16.44%
 Taux d'acceptation moyen pour l=192 : 18.56%
 Taux d'acceptation moyen pour l=193 : 18.72%
 Taux d'acceptation moyen pour l=194 : 18.68%
 Taux d'acceptation moyen pour l=195 : 18.72%
 Taux d'acceptation moyen pour l=196 : 18.76%
 Taux d'acceptation moyen pour l=197 : 18.52%
 Taux d'acceptation moyen pour l=198 : 18.48%
 Taux d'acceptation moyen pour l=199 : 18.88%
 Taux moyen = 15.43%

```

[45]: plt.plot(z_i_sampled-z_true[2:])
plt.title(f'Différence entre les $z_i$ échantillonnés et les "vrais" $z_i$,  

↪différence moyenne : {np.mean(z_i_sampled-z_true[2:]):.2f}')
plt.show()
  
```



Hormis quelques exceptions, le sampler Gibbs a bien échantillonné de la postérieure.

```
[47]: z_pop_rand=np.random.rand(2)+15*np.ones(2)
z_true=np.concatenate([[LATENT['t0']], [LATENT['v0']], LATENT['xi'], LATENT['tau']])
burnin=7500
z_pop_sampled=HMwG_sampler(Y,z_pop_rand,z_true,sigma_prop=1*1e-5,
                           t0_bar=np.array(THETA_TRUE['t0_bar']),v0_bar=np.
→array(THETA_TRUE['v0_bar']),
                           sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
                           sigma_xi=THETA_TRUE['sigma_xi'],
→sigma_tau=THETA_TRUE['sigma_tau'],sigma=THETA_TRUE['sigma'],
                           max_iter=15000,burnin=burnin,verbose=True)
```

Taux d'acceptation moyen pour l=0 : 22.08%

Taux d'acceptation moyen pour l=1 : 19.83%

Taux moyen = 20.96%

```
[50]: print(z_pop_sampled, z_true[:2])
np.mean(z_pop_sampled-z_true[:2])
```

```
[3.3492488687622206, 1.2250997004681266] [1.06808543 1.14689136]
```

```
[50]: 1.1796858907299483
```

L'échantillonnage des z_{pop} semble également correct, mais nécessite un σ_{prop} différent

```
[57]: #@jit(nopython=True)
def HMwG_SAEM(Y,THETA,FIXED,PRIOR,max_iter, burnin, alpha_burn,
              max_iter_hm=10000, burnin_hm=4000,sigma_hm=2*1e-2,fenetre_hm=4000,
→THETA_TRUE=THETA_TRUE>window=200,verbose=False):
    theta_true=np.array(list(THETA_TRUE.values()))
    N=Y.shape[0]
    k=Y.shape[1]
    zk=np.zeros(2*N+2)
    z_pop=zk[:2]
    z_i=zk[2:]
    S=np.zeros(5)
    for i in progressbar(range(max_iter),"En cours"):
        z_i = HMwG_sampler(Y,np.array(z_i),zk,sigma_prop=sigma_hm,t0_bar=np.
→array(THETA['t0_bar']),v0_bar=np.array(THETA['v0_bar']),
        sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
        sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma'],
        pi=log_posterior_simple,max_iter=max_iter_hm,
        burnin=burnin_hm,verbose=verbose)
```

```

        z_pop = HMwG_sampler(Y,np.array(z_pop),zk,sigma_prop=sigma_hm,t0_bar=np.
→array(THETA['t0_bar']),v0_bar=np.array(THETA['v0_bar']),
            sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
            ↵
→sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma'],
            pi=log_posterior_simple,max_iter=max_iter_hm,
            burnin=burnin_hm,verbose=verbose)
        #on concatène les 2 z indépendants updatés
        zk=np.concatenate([z_pop, z_i])
        if i<=burnin:
            epsilon=1
        else:
            epsilon=(i-burnin)**(-alpha_burn)
            if i%window==0:
                theta_values=np.array(list(THETA.values()))
                print(theta_true-theta_values)
            S=S+epsilon*(s_yz(Y,t0=zk[0],v0=zk[1],xi=zk[2:N+2],tau=zk[N+2:])-S)
            THETA_UPDATE=max_thetak(S,N,k,m=PRIOR['m'],v=PRIOR['v'],
            ↵
→t0_bar=THETA['t0_bar'],sigma_t0=FIXED['sigma_t0'],s_t0=PRIOR['s_t0'],t0_dbar=PRIOR['t0_dbar']
            ↵
→v0_bar=THETA['v0_bar'],sigma_v0=FIXED['sigma_v0'],s_v0=PRIOR['s_v0'],v0_dbar=PRIOR['v0_dbar']
            ↵
→sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma'])
            THETA=THETA_UPDATE
        return THETA

```

```

[61]: #Sanity check. On part d'un theta_0 peu éloigné de theta_star
theta_list=['t0_bar','v0_bar','sigma_xi','sigma_tau','sigma']
THETA=copy.deepcopy(THETA_TRUE)
theta_values=np.array(list(THETA.values()))
theta_values=theta_values+np.random.uniform(-1,1,size=theta_values.shape)+15*np.
→ones(theta_values.shape[0])
THETA0=dict(zip(list(theta_list), theta_values))
theta_true=np.array(THETA_TRUE.values())

#On a vu que les z_i et les z_pop ont besoin d'un hyperparamètre sigma_prop↵
→différent. On prend à la moitié environ
start_time=time.time()
THETA_FINAL_Gibbs=HMwG_SAEM(Y,THETA=THETA0, FIXED=FIXED, PRIOR=PRIOR,↵
→max_iter=1000, burnin=500, alpha_burn=2.5/4,
            max_iter_hm=100, burnin_hm=80,sigma_hm=5*1e-4,↵
→THETA_TRUE=THETA_TRUE>window=100,verbose=False)

duree_gibbs=time.time() - start_time

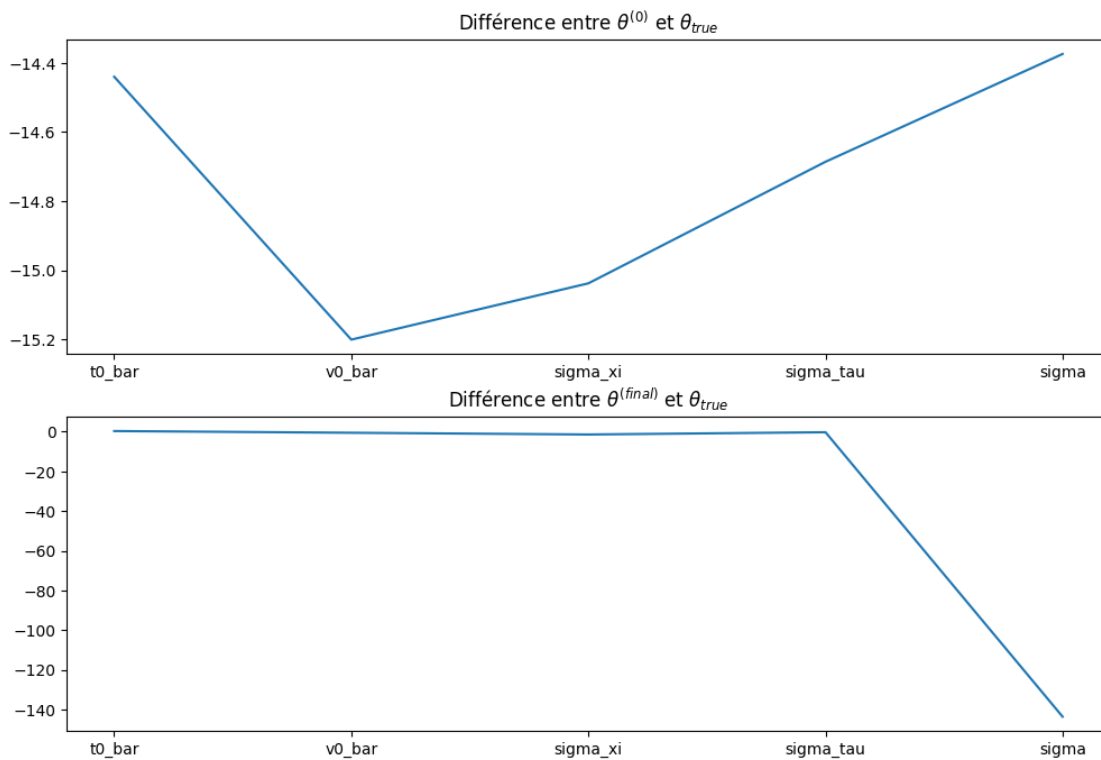
```

```
print("--- %s seconds ---" % (duree_gibbs))
```

```
[ 0.37085044 -0.55653377 -0.85321008 -0.27651161 -22.42611627]... 600/1000
Est wait 09:11.51
[ 0.33262675 -0.53589931 -1.01728131 -0.21941328 -143.59029677]700/1000
Est wait 06:53.61
[ 0.32317349 -0.52478677 -1.23341217 -0.59801441 -77.29568126]... 800/1000
Est wait 04:35.88
[ 0.34125491 -0.53270604 -1.4960923 -0.51319232 -37.17951101]... 900/1000
Est wait 02:17.88
En cours[...]1000/1000
Est wait 00:00.00
```

```
--- 1379.4267058372498 seconds ---
```

```
[63]: fig, (ax1,ax2)=plt.subplots(2)
ax1.plot(THETA.keys(),np.array(list(THETA.values()))- theta_values)
ax1.set_title(r'Différence entre  $\theta^{(0)}$  et  $\theta_{true}$ ')
ax2.plot(THETA.keys(),np.array(list(THETA.values()))- np.
↪array(list(THETA_FINAL_Gibbs.values()))))
ax2.set_title(r'Différence entre  $\theta^{(final)}$  et  $\theta_{true}$ ')
plt.show()
```



La convergence vers θ_{true} semble correcte et assez rapide quoiqu'incorrecte sur les sigma (répétitivement).

2.7 Question 8

Comme on l'a vu, le Hastings Metropolis within Gibbs sampler “vanilla” est intéressant mais il est assez long et a du mal à déterminer une dynamique “globale” pour le vecteur z . À l'inverse, le HM classique avait du mal à converger car on refusait trop souvent les propositions, toutes les variables étant échantillonnées sur la même échelle (alors que les ξ_i et t_0 ne sont pas forcément du même ordre de grandeur par exemple).

Choisir un échantillonneur de Gibbs par blocs permet de trouver le juste milieu. Si l'on choisit convenablement les blocs on peut alors ajouter de l'information sur ce qu'on doit échantillonner. On limite donc le nombre d'itérations de Gibbs nécessaires (nombre de blocs au lieu de $2N+2$), sans drastiquement diminuer le taux d'acceptation pour un même σ_{prop} .

```
[64]: #@jit(nopython=True)
def HMwG_blockssampler(Y,z,sigma_prop_i, sigma_prop_pop,t0_bar=np.
    ↳array(THETA_TRUE['t0_bar']),v0_bar=np.array(THETA_TRUE['v0_bar']),
        sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
            ↳
    ↳sigma_xi=THETA_TRUE['sigma_xi'],sigma_tau=THETA_TRUE['sigma_tau'],sigma=THETA_TRUE
        ['sigma'],
            pi=log_posterior_simple,max_iter=1000,
            fenetre=500, burnin=None,verbose=True,gibbs_iterations=1):
    burnin=0 if None else burnin
    zk=z
    z_pop=zk[:2]
    z_i=z[2:]
    def log_post_Gibbs_zpop(z_pop, zk):
        return pi(Y=Y, z=np.concatenate([z_pop, zk[2:]]),
            t0_bar=t0_bar,v0_bar=v0_bar,
            sigma_t0=sigma_t0,sigma_v0=sigma_v0,
            sigma_xi=sigma_xi,sigma_tau=sigma_tau,sigma=sigma)
    def log_post_Gibbs_zi(z_i, zk):
        return pi(Y=Y, z=np.concatenate([zk[:2],z_i]),
            t0_bar=t0_bar,v0_bar=v0_bar,
            sigma_t0=sigma_t0,sigma_v0=sigma_v0,
            sigma_xi=sigma_xi,sigma_tau=sigma_tau,sigma=sigma)

    #Partie MHwG sampling
    for g in range(gibbs_iterations):
        accept_pop=0
        accept_i=0
        z_i_list=[]
        z_pop_list=[]
        #Premier bloc
        for k in range(max_iter):
```



```

zi_star=np.random.normal(loc=z_i,scale=np.sqrt(sigma_prop_i))
l_zi_star=log_post_Gibbs_zi(zi_star,zk)
l_zi=log_post_Gibbs_zi(z_i,zk)
#On calcule tout dans l'espace des log-probs
prob_choose=0 if 0<= l_zi_star-l_zi else l_zi_star-l_zi
u=np.random.rand()
if prob_choose>=np.log(u):
    z_i_list.append(zi_star)
    zk=np.concatenate([zk[:2],zi_star])
    z_i=zi_star
    if k>=burnin:
        accept_i+=1
else:
    z_i_list.append(z_i)

#Second bloc
for k in range(max_iter):
    zpop_star=np.random.normal(loc=z_pop,scale=np.sqrt(sigma_prop_pop))
    l_zpop_star=log_post_Gibbs_zpop(zpop_star,zk)
    l_zpop=log_post_Gibbs_zpop(z_pop,zk)
    #On calcule tout dans l'espace des log-probs
    prob_choose=0 if 0<= l_zpop_star-l_zpop else l_zpop_star-l_zpop
    u=np.random.rand()
    if prob_choose>=np.log(u):
        z_pop_list.append(zpop_star)
        zk=np.concatenate([zpop_star,zk[2:]])
        z_pop=zpop_star
        if k>=burnin:
            accept_pop+=1
    else:
        z_pop_list.append(z_pop)

    if verbose:
        print(f'Taux d\'acceptation moyen pour z_pop (Itération de gibbs n°
→{g+1})= {accept_pop/(max_iter-burnin):.2%}')
        print(f'Taux d\'acceptation moyen pour z_i (Itération de gibbs n°
→{g+1})= {accept_i/(max_iter-burnin):.2%}')

    idx=np.random.randint(1, int(.5*(max_iter-burnin)))
    z_sampled=np.concatenate([z_pop_list[-idx],z_i_list[-idx]])
    return z_sampled

```

```

[66]: z_rand=np.random.rand(2*N+2)+15*np.ones(2*N+2)
z_true=np.concatenate([[LATENT['t0']], [LATENT['v0']], LATENT['xi'], LATENT['tau']])
burnin=10000

```

```

z_sampled = HMwG_block_sampler(Y,z_rand,sigma_prop_pop=1*1e-6,
    ↳sigma_prop_i=5*1e-5,
        t0_bar=np.array(THETA_TRUE['t0_bar']),v0_bar=np.
    ↳array(THETA_TRUE['v0_bar']),
        sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
        sigma_xi=THETA_TRUE['sigma_xi'],
    ↳sigma_tau=THETA_TRUE['sigma_tau'],sigma=THETA_TRUE['sigma'],
        max_iter=20000,burnin=burnin,verbose=True, gibbs_iterations=5)

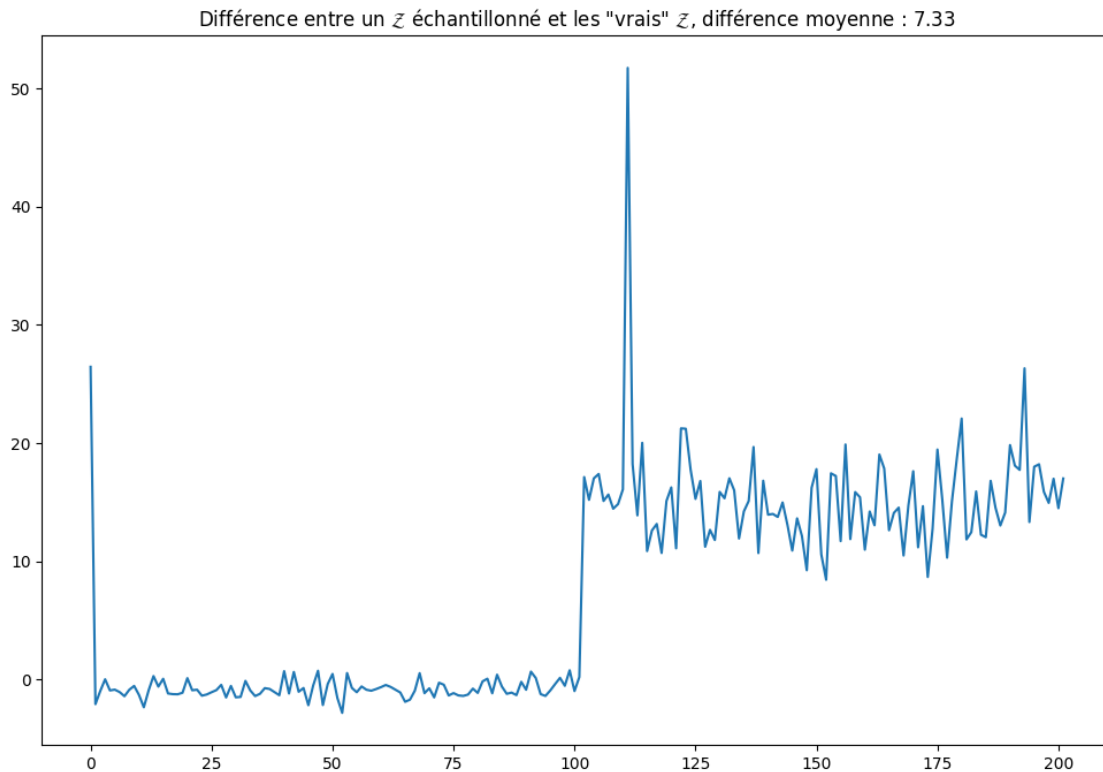
```

Taux d'acceptation moyen pour z_{pop} (Itération de gibbs n° 1)= 50.07%
 Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 48.67%
 Taux d'acceptation moyen pour z_{pop} (Itération de gibbs n° 2)= 49.57%
 Taux d'acceptation moyen pour z_i (Itération de gibbs n° 2)= 47.64%
 Taux d'acceptation moyen pour z_{pop} (Itération de gibbs n° 3)= 56.58%
 Taux d'acceptation moyen pour z_i (Itération de gibbs n° 3)= 46.81%
 Taux d'acceptation moyen pour z_{pop} (Itération de gibbs n° 4)= 47.20%
 Taux d'acceptation moyen pour z_i (Itération de gibbs n° 4)= 30.16%
 Taux d'acceptation moyen pour z_{pop} (Itération de gibbs n° 5)= 41.66%
 Taux d'acceptation moyen pour z_i (Itération de gibbs n° 5)= 13.80%

```

[67]: plt.plot(z_sampled-z_true)
plt.title(f'Différence entre un  $\mathcal{Z}$  échantillonné et les "vrais"
    ↳ $\mathcal{Z}$ , différence moyenne : {np.mean(z_sampled-z_true):.2f}')
plt.show()

```



```

[68]: #@jit(nopython=True)
def HMw_blockG_SAEM(Y,THETA,FIXED,PRIOR,max_iter, burnin, alpha_burn,
                    max_iter_hm=10000, burnin_hm=4000,sigma_hm_pop=2*1e-2,
    ↪sigma_hm_i=2*1e-2,fenetre_hm=4000, THETA_TRUE=THETA_TRUE>window=200,
                    verbose=False,gibbs_iterations=1):
    theta_true=np.array(list(THETA_TRUE.values()))
    N=Y.shape[0]
    k=Y.shape[1]
    zk=np.zeros(2*N+2)
    S=np.zeros(5)
    verbose_tmp=verbose
    for i in progressbar(range(max_iter),"En cours"):
    ↪#for i in range(max_iter):
        if i%window==0:
            verbose=True
        else:
            verbose=verbose_tmp
        z =
    ↪HMwG_block_sampler(Y,zk,sigma_prop_pop=sigma_hm_pop,sigma_prop_i=sigma_hm_i,t0_bar=np.
    ↪array(THETA['t0_bar']),v0_bar=np.array(THETA['v0_bar']),
            sigma_t0=FIXED['sigma_t0'],sigma_v0=FIXED['sigma_v0'],
            ↪
    ↪sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma'],
            pi=log_posterior_simple,max_iter=max_iter_hm,
            ↪
    ↪burnin=burnin_hm,verbose=verbose,gibbs_iterations=gibbs_iterations)
        if i<=burnin:
            epsilon=1
        else:
            epsilon=(i-burnin)**(-alpha_burn)
            if i%window==0:
                theta_values=np.array(list(THETA.values()))
                print(theta_true-theta_values)

        S+=epsilon*(s_yz(Y,t0=z[0],v0=z[1], xi=z[2:N+2],tau=z[N+2:]) - S)
        THETA_UPDATE=max_thetak(S,N,k,m=PRIOR['m'],v=PRIOR['v'],
            ↪
    ↪t0_bar=THETA['t0_bar'],sigma_t0=FIXED['sigma_t0'],s_t0=PRIOR['s_t0'],t0_dbar=PRIOR['t0_dbar']
            ↪
    ↪v0_bar=THETA['v0_bar'],sigma_v0=FIXED['sigma_v0'],s_v0=PRIOR['s_v0'],v0_dbar=PRIOR['v0_dbar']
            ↪
    ↪sigma_xi=THETA['sigma_xi'],sigma_tau=THETA['sigma_tau'],sigma=THETA['sigma'])
        THETA=THETA_UPDATE
        zk=z

```

```
return THETA
```

```
[69]: #Sanity check. On part d'un theta_0 peu éloigné de theta_star
theta_list=['t0_bar','v0_bar','sigma_xi','sigma_tau','sigma']
THETA=copy.deepcopy(THETA_TRUE)
theta_values_far=np.array(list(THETA.values()))
theta_values_far=theta_values_far+np.random.uniform(-1,1,size=theta_values_far.
↳shape) + 15*np.ones(theta_values_far.shape)
THETA0=dict(zip(list(theta_list), theta_values_far))

start_time=time.time()
THETA_FINAL_block_far=HMw_blockG_SAEM(Y,THETA0,FIXED,PRIOR,max_iter=15000,↳
↳burnin=7500, alpha_burn=2.5/4,
max_iter_hm=2000,↳
↳burnin_hm=1800,sigma_hm_pop=1*1e-4,sigma_hm_i=1*1e-4,↳
↳THETA_TRUE=THETA_TRUE>window=1000,verbose=False,gibbs_iterations=1)

duree_gibbsblock_far=time.time() - start_time
print("--- %s seconds ---" % (duree_gibbsblock_far))
```

```
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 71.00%
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 100.00%
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 14.00%]
1000/15000 Est wait 41:14.88
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 12.00%]
2000/15000 Est wait 38:28.23
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 9.50%.]
3000/15000 Est wait 35:42.73
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 10.50%]
4000/15000 Est wait 32:51.03
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 9.00%.]
5000/15000 Est wait 29:53.85
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 10.50%]
6000/15000 Est wait 26:48.25
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 12.00%]
7000/15000 Est wait 23:43.34
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 12.00%]
8000/15000 Est wait 20:46.76
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
```

```

[ 0.1353437 -0.75316703  0.1348917   3.58835641 -8.15557731]
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 11.00%
9000/15000 Est wait 17:48.63
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
[ 0.12642368 -0.75699192  0.13088516  3.59009635 -8.09621796]
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 5.50%.]
10000/15000 Est wait 14:52.25
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
[ 0.12135612 -0.75630825  0.13206738  3.58693525 -8.08570628]
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 11.00%]
11000/15000 Est wait 11:54.66
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
[ 0.12343649 -0.75212777  0.13351766  3.56430134 -7.8208349 ]
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 13.00%]
12000/15000 Est wait 08:56.52
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
[ 0.11001384 -0.7599745   0.13310106  3.54608976 -7.51597215]
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 9.50%.]
13000/15000 Est wait 05:58.06
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
[ 0.10442453 -0.76195686  0.12788593  3.53402993 -7.32778104]
Taux d'acceptation moyen pour z_pop (Itération de gibbs n° 1)= 11.50%]
14000/15000 Est wait 02:59.22
Taux d'acceptation moyen pour z_i (Itération de gibbs n° 1)= 0.00%
[ 0.11433636 -0.76379004  0.12926206  3.53137509 -7.27622191]
En cours[...]
15000/15000 Est wait 00:00.00

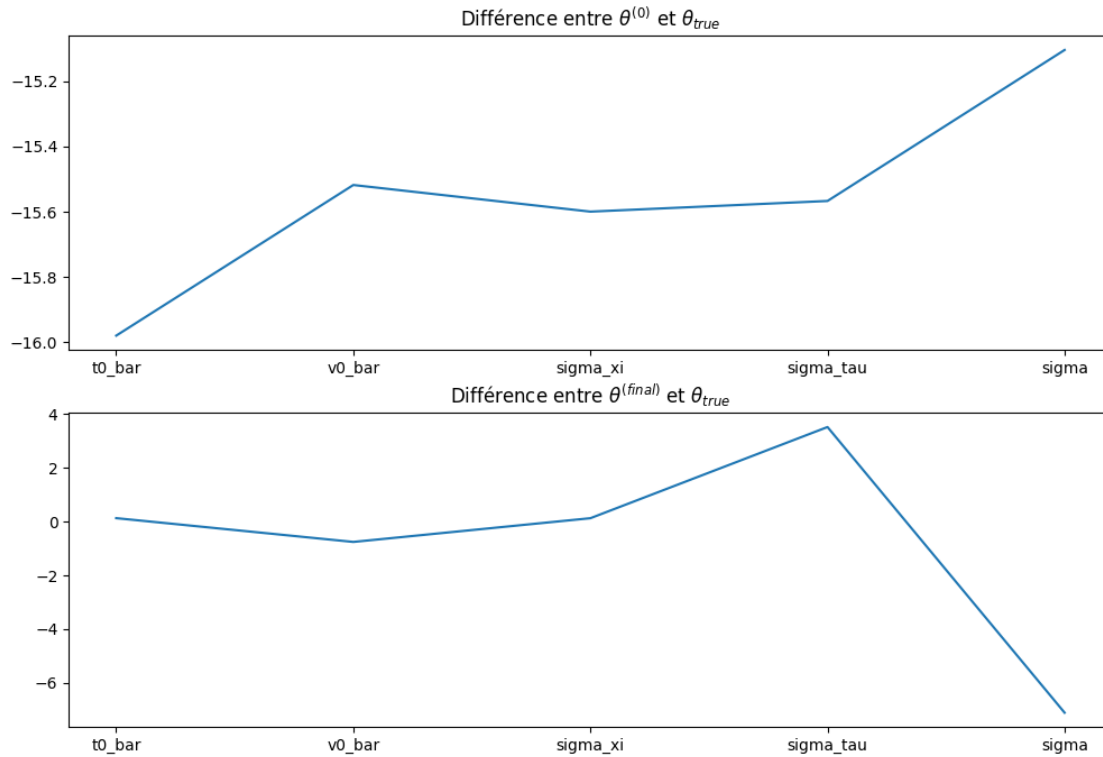
--- 2685.628041267395 seconds ---

```

```

[70]: fig, (ax1,ax2)=plt.subplots(2)
      ax1.plot(THETA.keys(),np.array(list(THETA.values()))- theta_values_far)
      ax1.set_title(r'Différence entre $\theta^{\{0\}}$ et $\theta_{\{true\}}$')
      ax2.plot(THETA.keys(),np.array(list(THETA.values()))- np.
      ↪array(list(THETA_FINAL_block_far.values()))))
      ax2.set_title(r'Différence entre $ \theta^{\{final\}} $ et $ \theta_{\{true\}}$')
      plt.show()

```



L'échantillonnage par Gibbs en bloc semble avoir du mal à converger très précisément

```
[103]: THETA_FINAL_MCMC
nain=dict(THETA_FINAL_MCMC)
np.mean(list(nain.values()))
theta_true.mean
```

```
[103]: <function ndarray.mean>
```

```
[106]: print(f'Durée de MH classique : {duree_mcmc}\nPrécision finale :{np.mean(np.
    ↳array(list(dict(THETA_FINAL_MCMC).values()))-np.array(list(THETA.
    ↳values()))})\n')
print(f'Durée de Gibbs classique : {duree_gibbs}\nPrécision finale {np.mean(np.
    ↳array(list(dict(THETA_FINAL_Gibbs).values()))-np.array(list(THETA.
    ↳values()))})\n')
print(f'Durée de Block-Gibbs classique : {duree_gibbsblock_far}\nPrécision_
    ↳finale {np.mean(np.array(list(dict(THETA_FINAL_block_far).values()))-np.
    ↳array(list(THETA.values()))})\n')
```

Durée de MH classique : 3618.164031982422

Précision finale :-0.07147702306917636

Durée de Gibbs classique : 1379.4267058372498

Précision finale 29.106131385698724

Durée de Block-Gibbs classique : 2685.628041267395
Précision finale 0.8251083099039089

Le Block Gibbs semble être le meilleur compromis, plutôt robuste aux conditions initiales.

3 Exercice 2 : Multiplicative Hasting-Metropolis

3.1 Question 1

On va utiliser la méthode de la fonction muette. Soit h une fonction mesurable continue bornée.

On sait que $\epsilon \sim \pi_f$ et $\mathcal{B} \sim \text{Bernoulli}(\frac{1}{2})$.

On a :

$$\begin{aligned}\mathbb{E}[h(Y)] &= \mathbb{E}[h(Y) \mid \mathcal{B} = 1]P(\mathcal{B} = 1) + \mathbb{E}[h(Y) \mid \mathcal{B} = 0]P(\mathcal{B} = 0) \\ &= \frac{1}{2} (\mathbb{E}[h(Y) \mid \mathcal{B} = 1] + \mathbb{E}[h(Y) \mid \mathcal{B} = 0]) \\ &= \frac{1}{2} \left(\mathbb{E}[h(\epsilon X) \mid \mathcal{B} = 1] + \mathbb{E}[h(\frac{X}{\epsilon}) \mid \mathcal{B} = 0] \right) \\ &= \frac{1}{2} \left(\underbrace{\int_{-1}^1 h(\epsilon X) f(\epsilon) d\epsilon}_{\text{On pose } Y=\epsilon X, \Phi_1} + \underbrace{\int_{-1}^0 h(\frac{X}{\epsilon}) f(\epsilon) d\epsilon + \int_0^1 h(\frac{X}{\epsilon}) f(\epsilon) d\epsilon}_{\text{On pose } Y=\frac{X}{\epsilon}, \Phi_2} \right)\end{aligned}$$

Φ_1 et Φ_2 sont C^1 , bijectives si $X < 0$ ou $X > 0$.

- $\Phi_1 : \epsilon \rightarrow \frac{Y}{X} \quad d\epsilon = \frac{dY}{X} \quad \Phi_1^{-1}(-1) = -X, \quad \Phi_1^{-1}(1) = X$
- $\Phi_2 : \epsilon \rightarrow \frac{X}{Y} \quad d\epsilon = -\frac{X}{Y^2} dY \quad \Phi_2^{-1}(-1) = -X, \quad \Phi_2^{-1}(0) = \pm\infty, \quad \Phi_2^{-1}(1) = X$

On a donc :

- Si $X > 0$:

$$\begin{aligned}\mathbb{E}(h(Y)) &= \frac{1}{2} \left(\int_{-x}^x h(Y) f\left(\frac{Y}{X}\right) \frac{dY}{X} - \int_{-x}^{-\infty} h(Y) f\left(\frac{X}{Y}\right) \frac{X}{Y^2} dY - \int_{+\infty}^x h(Y) f\left(\frac{X}{Y}\right) \frac{X}{Y^2} dY \right) \\ &= \frac{1}{2} \left(\int_{-x}^x h(Y) f\left(\frac{Y}{X}\right) \frac{dY}{X} + \int_{-\infty}^{-x} h(Y) f\left(\frac{X}{Y}\right) \frac{X}{Y^2} dY + \int_x^{+\infty} h(Y) f\left(\frac{X}{Y}\right) \frac{X}{Y^2} dY \right)\end{aligned}$$

- Si $X < 0$:

$$\begin{aligned}\mathbb{E}(h(Y)) &= \frac{1}{2} \left(\int_{-x}^x h(Y) f\left(\frac{Y}{X}\right) \frac{dY}{X} - \int_{-x}^{+\infty} h(Y) f\left(\frac{X}{Y}\right) \frac{X}{Y^2} dY - \int_{-\infty}^x h(Y) f\left(\frac{X}{Y}\right) \frac{X}{Y^2} dY \right) \\ &= \frac{1}{2} \left(\int_{-x}^x h(Y) f\left(\frac{Y}{X}\right) \frac{dY}{-X} + \int_{-\infty}^{-x} h(Y) f\left(\frac{X}{Y}\right) \frac{-X}{Y^2} dY + \int_x^{+\infty} h(Y) f\left(\frac{X}{Y}\right) \frac{-X}{Y^2} dY \right)\end{aligned}$$

On peut voir qu'on a simplement inversé les signes dans l'intégrale pour tomber sur la valeur absolue de x .

Si $X = 0$, pas besoin de faire de changement de variable : il vient directement que

$$\begin{aligned}\mathbb{E}(h(\epsilon \times 0)) &= \frac{1}{2} \left(\int_{-1}^1 h(\epsilon \times 0) f(\epsilon) d\epsilon + \int_{-1}^0 h\left(\frac{0}{\epsilon}\right) f(\epsilon) d\epsilon + \int_0^1 h\left(\frac{0}{\epsilon}\right) f(\epsilon) d\epsilon \right) \\ &= \frac{1}{2} \left(2 \times \int_{-1}^1 h(0) f(\epsilon) d\epsilon \right) \quad \text{f est densité de } \epsilon \\ &= h(0)\end{aligned}$$

Comme on sait que $y \sim q(x, dy)$, *i.e.* $\mathbb{E}(Y) = \int q(x, dy)$, on peut identifier $f_Y(y) = q(x, dy)$ avec la méthode de la fonction muette et écrire $\forall x \neq 0$:

$$\begin{aligned}q(x, dy) &= \frac{1}{2} \left(f\left(\frac{dy}{x}\right) \times \frac{1}{|x|} \mathbb{1}_{|x| < |y|} + f\left(\frac{x}{dy}\right) \frac{|x|}{dy^2} \mathbb{1}_{|x| > |y|} + f\left(\frac{x}{dy}\right) \frac{|x|}{dy^2} \mathbb{1}_{|x| > |y|} \right) \\ &= \frac{1}{2} \left(f\left(\frac{dy}{x}\right) \times \frac{1}{|x|} \mathbb{1}_{|dy| < |x|} + f\left(\frac{x}{dy}\right) \frac{|x|}{dy^2} \mathbb{1}_{|dy| > |x|} \right)\end{aligned}$$

Pour que le noyau q soit utilisé dans l'étape MCMC il faut qu'il soit densité de probabilité. On se concentre donc sur $x_0 \neq 0$ et $f(\epsilon) = 0$ si $\epsilon = 0$

3.2 Question 2

Le taux d'acceptation dans l'algorithme de Hastings-Metropolis est $\alpha(x, y) = 1 \wedge \frac{q(Y, X)\pi(Y)}{q(X, Y)\pi(X)}$ pour pouvoir répondre à l'équation de balance détaillée, et donc que π soit mesure invariante.

On a donc :

$$\begin{aligned}\alpha(X, Y) &= 1 \wedge \frac{q(Y, X)\pi(Y)}{q(X, Y)\pi(X)} \\ &= 1 \wedge \frac{\frac{1}{2} \left(f\left(\frac{X}{Y}\right) \times \frac{1}{|Y|} \mathbb{1}_{|X| < |Y|} + f\left(\frac{Y}{X}\right) \frac{|Y|}{X^2} \mathbb{1}_{|X| > |Y|} \right) \pi(Y)}{\frac{1}{2} \left(f\left(\frac{Y}{X}\right) \times \frac{1}{|X|} \mathbb{1}_{|Y| < |X|} + f\left(\frac{X}{Y}\right) \frac{|X|}{Y^2} \mathbb{1}_{|Y| > |X|} \right) \pi(X)} \\ &= 1 \wedge \frac{\frac{1}{2} \left(f\left(\frac{X}{Y}\right) \times \frac{1}{|Y|} \mathbb{1}_{|X| < |Y|} + f\left(\frac{Y}{X}\right) \frac{|Y|}{X^2} \mathbb{1}_{|Y| < |X|} \right) \pi(Y)}{\frac{1}{2} \left(f\left(\frac{X}{Y}\right) \frac{|X|}{Y^2} \mathbb{1}_{|X| < |Y|} + f\left(\frac{Y}{X}\right) \times \frac{1}{|X|} \mathbb{1}_{|Y| < |X|} \right) \pi(X)} \\ &= 1 \wedge \frac{\frac{|Y|}{|X|} \times \frac{1}{2} \left(f\left(\frac{X}{Y}\right) \times \frac{|X|}{Y^2} \mathbb{1}_{|X| < |Y|} + f\left(\frac{Y}{X}\right) \frac{1}{|X|} \mathbb{1}_{|X| > |Y|} \right) \pi(Y)}{\frac{1}{2} \left(f\left(\frac{X}{Y}\right) \frac{|X|}{Y^2} \mathbb{1}_{|Y| > |X|} + f\left(\frac{Y}{X}\right) \times \frac{1}{|X|} \mathbb{1}_{|Y| < |X|} \right) \pi(X)} \\ &= 1 \wedge \frac{|Y|\pi(Y)}{|X|\pi(X)} \\ &= \begin{cases} 1 \wedge \frac{\pi(Y)}{\pi(X)} \times |\epsilon| & \text{si } \mathcal{B} = 1 \\ 1 \wedge \frac{\pi(Y)}{\pi(X)} \times \frac{1}{|\epsilon|} & \text{sinon} \end{cases}\end{aligned}$$

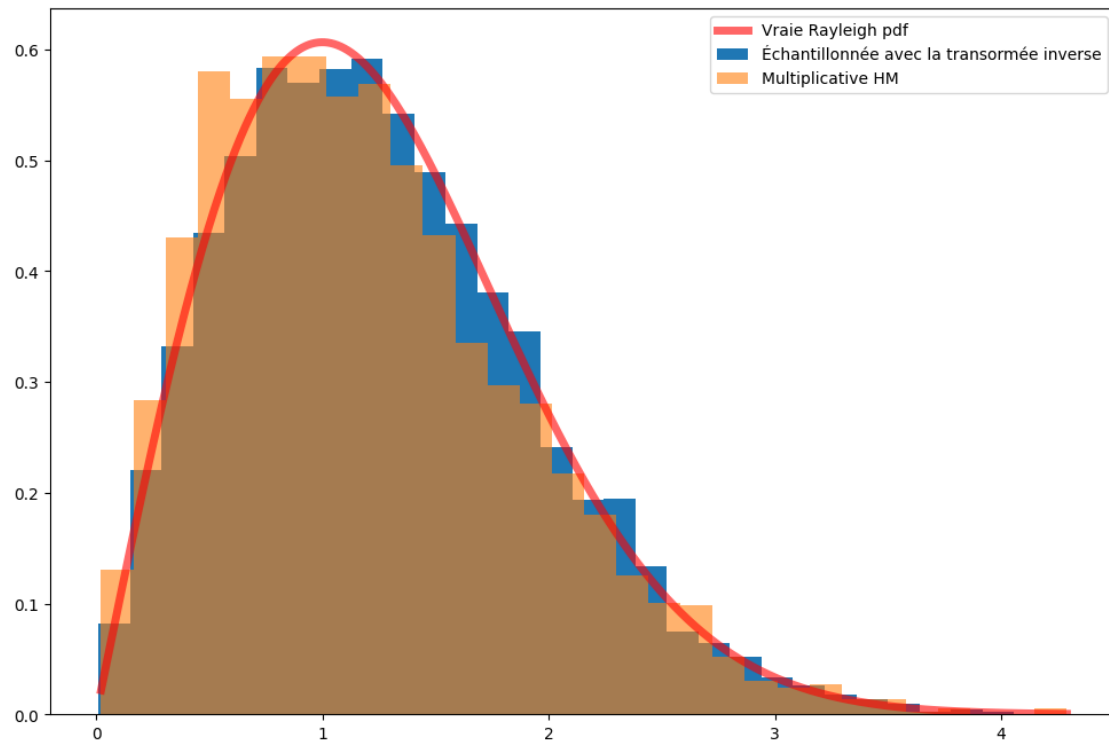
3.3 Question 3

On prend f densité de π_f distribution uniforme sur $] - 1, 1[$ et 2 densités cibles différentes : - Une densité de Rayleigh qu'on sait sampler avec la méthode de la transformée inverse (cf TP1) - Une densité normale

```
[107]: #On va utiliser une Rayleigh(1) du TP1 pour la première partie, et une
        ↪gaussienne pour la seconde.
        #\pi_f est une loi uniforme sur ]-1,1[
def multiplicative_HM(x, N, distribution, mu=None, sigma=None):
    x_list=[x]
    if distribution=='rayleigh':
        pi= lambda x: stats.rayleigh.pdf(x)
    if distribution=='gauss':
        pi= lambda x: stats.norm.pdf(x, loc=mu, scale=sigma)
    for i in range(N):
        epsilon=np.random.uniform(-1,1)
        b=np.random.binomial(1,.5)
        if b==1:
            x_star=epsilon*x
        elif b==0:
            x_star=x/epsilon
        u=np.random.rand()
        alpha=1 if 1 <= pi(x_star)/pi(x) * np.abs(x_star/x) else pi(x_star)/
        ↪pi(x) * np.abs(x_star/x)
        if u < alpha:
            x_list.append(x_star)
            x=x_star
    return x_list

[108]: ## Pour la Rayleigh:
N= 10000
u=np.random.rand(N)
r=np.sqrt(- 2*np.log(1-u))

sampled=multiplicative_HM(1, N, 'rayleigh')
x=np.linspace(min(sampled), max(sampled),N)
plt.plot(x, stats.rayleigh.pdf(x),'r-', lw=5, alpha=0.6, label='Vraie Rayleigh
        ↪pdf')
plt.hist(r, bins=30,density=True,label='Échantillonnée avec la transormée
        ↪inverse')
plt.hist(sampled, bins=30, density=True,label='Multiplicative HM',alpha=0.6)
plt.legend()
plt.show()
```



```
[109]: ## Pour la Gaussienne:
N= 100000
mu=5
sigma=3
u=np.random.rand(N)
sampled=multiplicative_HM(1, N, 'gauss', mu=mu, sigma= sigma)
x=np.linspace(min(sampled), max(sampled),N)
plt.plot(x, stats.norm.pdf(x,loc=mu, scale=sigma),'r-', lw=5, label='Vraie_
↪Gaussienne')
plt.hist(sampled, bins=50, density=True,label='Multiplicative HM')
plt.legend()
plt.show()
```

