

## Lista de Exercícios 4

Professor(a): Eduardo Mendes

Aluno: Franklin Alves de Oliveira

## Exercise 1 (Kalman Filter)

1.

Primeiramente, para  $f(x_t | x_{t-1})$ , temos:

$$\mathbb{E}[X_t | X_{t-1}] = \mathbb{E}[\phi X_{t-1} + V_t | X_{t-1}] = \phi x_{t-1} + \mathbb{E}[V_t | X_{t-1}] = \phi x_{t-1} \quad (1)$$

Uma vez que  $X_{t-1}$  é conhecido e igual a  $x_{t-1}$ . Agora, note que:

$$\mathbb{V}[X_t | X_{t-1}] = \mathbb{V}[\phi X_{t-1} + V_t | X_{t-1}] = \mathbb{V}[V_t | X_{t-1}] = \sigma_V^2 \quad (2)$$

Como, no tempo  $t$ ,  $X_{t-1}$  é conhecido, é fácil ver que  $X_t$  tem a mesma distribuição de  $V_t$ , com média e variâncias dada em 1 e 2, respectivamente. Isto é,

$$X_t | X_{t-1} \sim \mathcal{N}(\phi x_t, \sigma_V^2)$$

Agora, vamos fazer a mesma coisa para  $g(y_t | x_t)$ . Ora,

$$\mathbb{E}[Y_t | X_t] = \mathbb{E}[X_t + W_t | X_t] = x_t + \mathbb{E}[W_t | X_t] = x_t \quad (3)$$

e

$$\begin{aligned} \mathbb{V}[Y_t | X_t] &= \mathbb{V}[X_t + W_t | X_t] = \mathbb{V}[X_t | X_t] + \mathbb{V}[W_t | X_t] - 2\text{Cov}[X_t, W_t | X_t] = \mathbb{V}[W_t | X_t] = \\ &= \mathbb{V}[W_t] = \sigma_W^2 \end{aligned} \quad (4)$$

Logo, uma vez conhecido  $X_t$ , é fácil ver que  $Y_t$  segue a mesma distribuição de  $W_t$ , com média e variância dadas em 3 e 4, respectivamente. Isto é,

$$Y_t | X_t \sim \mathcal{N}(x_t, \sigma_W^2)$$

□

2.

Sabemos que

$$p(x_{t+1} | y_{1:t}) = \int f(x_{t+1} | x_t) p(x_t | y_{1:t}) dx_t = \int \mathcal{N}(x_{t+1}; \phi x_t, \sigma_V^2) \mathcal{N}(x_t; m_{t|t}, \sigma_{t|t}^2) dx_t. \quad (5)$$

Como  $X_{t+1} = \phi X_t + V_t$ , onde  $X_t \sim \mathcal{N}(m_{t|t}, \sigma_{t|t}^2)$ , temos:

$$\mathbb{E}[X_{t+1} | y_{1:t}] = \phi \mathbb{E}[X_t | y_{1:t}] + \mathbb{E}[V_t | y_{1:t}] = \phi m_{t|t} = m_{t+1|t} \quad (6)$$

$$\mathbb{V}[X_{t+1} | y_{1:t}] = \phi^2 \mathbb{V}[X_t | y_{1:t}] + \mathbb{V}[V_t | y_{1:t}] = \phi^2 \sigma_{t|t}^2 + \sigma_V^2 = \sigma_{t+1|t}^2 \quad (7)$$

Dado que  $X_t$  e  $V_t$  são independentes.

Assim, como  $p(x_{t+1} | y_{1:t})$  é dada pelo produto de duas normais, temos que:

$$p(x_{t+1} | y_{1:t}) = \mathcal{N}(m_{t+1|t}, \sigma_{t+1|t}^2)$$

□

**3.**

Pela regra de Bayes e Lei da probabilidade total, podemos escrever

$$p(x_{t+1} | y_{1:t+1}) = \frac{g(y_{t+1} | x_{t+1})p(x_{t+1} | y_{1:t})}{\int g(y_{t+1}|x_{t+1})p(x_{t+1} | y_{1:t}) dx_{t+1}}. \quad (8)$$

Logo,

$$p(x_{t+1} | y_{1:t+1}) \propto \exp \left\{ -\frac{(y_{t+1} - x_{t+1})^2}{2\sigma_W^2} - \frac{(x_{t+1} - m_{t+1|t})^2}{2\sigma_{t+1|t}^2} \right\} \quad (9)$$

Abrindo o termo dentro da exponencial, é fácil ver que

$$\begin{aligned} \frac{(y_{t+1} - x_{t+1})^2}{2\sigma_W^2} + \frac{(x_{t+1} - m_{t+1|t})^2}{2\sigma_{t+1|t}^2} &= \frac{y_{t+1}^2 - 2y_{t+1}x_{t+1} + x_{t+1}^2}{2\sigma_W^2} + \frac{x_{t+1}^2 - 2x_{t+1}m_{t+1|t} + m_{t+1|t}^2}{2\sigma_{t+1|t}^2} = \\ &= \frac{y_{t+1}^2}{2\sigma_W^2} + x_{t+1}^2 \left( \frac{1}{2\sigma_{t+1|t}^2} + \frac{1}{2\sigma_W^2} \right) - 2x_{t+1} \left( \frac{m_{t+1|t}}{2\sigma_{t+1|t}^2} + \frac{y_{t+1}}{2\sigma_W^2} \right) + \frac{m_{t+1|t}^2}{2\sigma_{t+1|t}^2} = \\ &= \frac{y_{t+1}^2}{2\sigma_W^2} + \frac{x_{t+1}^2}{2\sigma_{t+1|t+1}^2} - 2x_{t+1} \frac{m_{t+1|t+1}}{2\sigma_{t+1|t+1}^2} + \frac{m_{t+1|t}^2}{2\sigma_{t+1|t}^2} = \\ &= \frac{y_{t+1}^2}{2\sigma_W^2} + \frac{(x_{t+1} - m_{t+1|t+1})^2}{2\sigma_{t+1|t+1}^2} - \frac{m_{t+1|t+1}}{2\sigma_{t+1|t+1}^2} + \frac{m_{t+1|t}^2}{2\sigma_{t+1|t}^2} \end{aligned} \quad (10)$$

Onde estamos denotando

$$\sigma_{t+1|t+1}^2 = \frac{\sigma_W^2 \sigma_{t+1|t}^2}{\sigma_W^2 + \sigma_{t+1|t}^2}$$

e

$$m_{t+1|t+1} = \sigma_{t+1|t+1}^2 \left( \frac{m_{t+1|t}}{\sigma_{t+1|t}^2} + \frac{y_{t+1}}{\sigma_W^2} \right).$$

□

**4.**

Usando a relação

$$p(y_{t+1} | y_{1:t}) = \int g(y_{t+1} | x_{t+1})p(x_{t+1} | y_{1:t}) dx_{t+1},$$

temos, do item 1:

$$g(y_{t+1} | x_{t+1}) = \mathcal{N}(x_{t+1}, \sigma_W^2) \quad (11)$$

e, do item 2,

$$p(x_{t+1} \mid y_{1:t}) = \mathcal{N}(m_{t+1|t}, \sigma_{t+1|t}^2) \quad (12)$$

Como  $p(y_{t+1} \mid y_{1:t})$  é o produto de duas normais, também é normal, com média e variância dadas por:

$$\mathbb{E}[Y_{t+1} \mid y_{1:t}] = \mathbb{E}[X_{t+1} \mid y_{1:t}] + \mathbb{E}[W_t \mid y_{1:t}] = \phi \mathbb{E}[X_t] = \phi m_{t|t} = m_{t+1|t} \quad (13)$$

$$\mathbb{V}[Y_{t+1} \mid y_{1:t}] = \mathbb{V}[X_{t+1} \mid y_{1:t}] + \mathbb{V}[W_t \mid y_{1:t}] = \sigma_{t+1|t}^2 + \sigma_W^2 \quad (14)$$

Onde  $W_t$  e  $X_t$  são independentes. □

## Exercise 2 (SIS Filter)

1.

$v_t$  é uma aproximação para a densidade de  $p(x_t | y_{0:t})$ . Por esse motivo é chamada de filtro.  $\square$

2.

Do algoritmo HMM, por independência, tem-se:

$$p_{Y_0}(y_0) = \int p_{Y_0, X_0}(y_0, x_0) dx_0 = \int p_{Y_0|X_0}(y_0 | x_0) p_{X_0}(x_0) dx_0 = \int g(y_0 | x_0) \mu(x_0) dx_0 \quad (15)$$

Ainda,

$$\begin{aligned} p_{Y_1, Y_0}(y_1, y_0) &= \int p_{Y_1, Y_0, X_1, X_0}(y_1, y_0, x_1, x_0) dx_0 = \\ &= \int p_{Y_1|Y_0, X_1, X_0}(y_1 | y_0, x_1, x_0) p_{X_1|Y_0, X_0}(x_1 | y_0, x_0) p_{Y_0|X_0}(y_0 | x_0) p_{X_0}(x_0) dx_{0:1} = \\ &= \int g(y_1 | x_1) f(x_1 | x_0) g(y_0 | x_0) \mu(x_0) dx_{0:1} \end{aligned} \quad (16)$$

$\square$

3.

Do item 2, é razoável pensar em aproximações para as distribuições  $p_{Y_0}$  e  $p_{Y_1, Y_0}(y_1, y_0)$ :

$$p_{Y_0}(y_0) \approx \hat{p}_{Y_0}(y_0) := \frac{1}{n} \sum_{i=1}^n g(y_0 | X_0^{(i)}) \quad (17)$$

$$p_{Y_1, Y_0}(y_1, y_0) \approx \hat{p}_{Y_1, Y_0}(y_1, y_0) := \frac{1}{n} \sum_{i=1}^n g(y_1 | X_1^{(i)}) g(y_0 | X_0^{(i)}) \quad (18)$$

Onde  $\{X_0^{(i)}\}_{i=1}^n$  e  $\{X_1^{(i)}\}_{i=1}^n$  são determinadas conforme o algoritmo SIS.  $\square$

4.

Para simplificar a notação, vamos denotar  $\Upsilon_t(x_{0:t}) = \prod_{j=0}^t g(y_j | x_j)$ . Assim,

$$\begin{aligned} \mathbb{E}[\Upsilon_t^2(x_{0:t})] &= \int \Upsilon_t^2(x_{0:t}) \prod_{i=0}^t \mu(x_i) dx_{0:t} = \int \Upsilon_{t-1}^2(x_{0:t-1}) \prod_{i=0}^{t-1} \mu(x_i) dx_{0:t-1} \int g^2(y_t | x_t) \mu(x_t) dx_t = \\ &= \mathbb{E}[\Upsilon_{t-1}^2(X_{0:t-1})] \cdot \mathbb{E}[g^2(y_t | X_t)] \end{aligned} \quad (19)$$

e

$$\mathbb{E}[\Upsilon_t(X_{0:t})] = \int \Upsilon_t(x_{0:t}) \prod_{i=0}^t \mu(x_i) dx_{0:t} = p_{Y_{0:t}}(y_{0:t}). \quad (20)$$

Ora,

$$\mathbb{V}\left[\frac{\prod_{k=0}^t g(y_k | X_k)}{p_{Y_{0:t}}(y_{0:t})}\right] - \mathbb{V}\left[\frac{\prod_{k=0}^{t-1} g(y_k | X_k)}{p_{Y_{0:t-1}}(y_{0:t-1})}\right] = \frac{\mathbb{E}[\Upsilon_t^2(X_{0:t})]}{p_{Y_{0:t}}^2(y_{0:t})} - \frac{\mathbb{E}[\Upsilon_{t-1}^2(X_{0:t-1})]}{p_{Y_{0:t-1}}^2(y_{0:t-1})} =$$

$$= \frac{\mathbb{E}[\Upsilon_{t-1}^2(X_{0:t-1})]}{p_{Y_{0:t-1}}^2(y_{0:t-1})} \left\{ \frac{\mathbb{E}(g^2(y_t | X_t))}{p_{Y_t}^2(y_t)} - 1 \right\} \quad (21)$$

Note que, todos os termos em 21 são positivos, pois estão elevados ao quadrado. Já o termo entre chaves, é fácil ver, da desigualdade de Jensen, que

$$\frac{\mathbb{E}(g^2(y_t | X_t))}{p_{Y_t}^2(y_t)} \geq 1. \quad (22)$$

Portanto, como queríamos mostrar, segue que

$$\mathbb{V} \left[ \frac{\prod_{k=0}^t g(y_k | X_k)}{p_{Y_{0:t}}(y_{0:t})} \right] - \mathbb{V} \left[ \frac{\prod_{k=0}^{t-1} g(y_k | X_k)}{p_{Y_{0:t-1}}(y_{0:t-1})} \right] = \frac{\mathbb{E}[\Upsilon_{t-1}^2(X_{0:t-1})]}{p_{Y_{0:t-1}}^2(y_{0:t-1})} \left\{ \frac{\mathbb{E}(g^2(y_t | X_t))}{p_{Y_t}^2(y_t)} - 1 \right\} \geq 0 \quad (23)$$

□

5.

(Não consegui fazer.)

6.

Do item 5, sabemos que a variância com respeito à distribuição conjunta de  $X_0, \dots, X_{t-1}$  cresce exponencialmente com o valor de  $t$ . Isso significa que, em poucos passos, a variância pode assumir valores muito grandes (explodir), o que eleva grandemente o custo computacional em se usar Monte Carlo, nesse caso. Para  $t$  grande, o custo computacional de uma iteração (i.e., tempo de execução) pode chegar a ser proibitivo. □

## Simulation question (Reversible jump MCMC)

- Verificando se os Kernels  $K_1$  e  $K_2$  do Metropolis-Hastings tradicional funcionam.

Sabemos que, para  $k = 1$ , o modelo que queremos amostrar é

$$\pi(\theta \mid k = 1) = \exp \left\{ \frac{1}{2} \theta^2 \right\}$$

e, para  $k = 2$ ,

$$\pi(\theta \mid k = 2) = \exp \left\{ \frac{1}{2} (\theta_1^2 + \theta_2^2) \right\}.$$

Então, inicialmente, foram criadas funções para amostrar dessas duas distribuições. Abaixo, é exibida uma descrição para 100 amostragens:

Distribuição	Min.	q25	Média	Mediana	q75	Máximo
$\pi(\theta \mid k = 1)$	2.028	2.1705	45.78143	2.277	2.444	4336.41
$\pi(\theta \mid k = 2)$	2.503	3.0595	76.40542	4.5170	4.9525	7141.137

Tabela 1: Estatísticas descritivas para 100 amostragens.

Nesse passo, também foram criadas funções para executar um *Random Walk Metropolis Hastings* em cada um dos modelos.

- Vamos propor uma variável auxiliar  $u \in \mathbb{R}$  tal que  $u \sim \text{Cauchy}(0, 1)$  (caudas mais pesadas que a Normal). Como função de *jump* do modelo 1 para o modelo 2, tomamos a transformação  $G_{1 \rightarrow 2}(\theta, u) = (\theta, u)$ , onde  $\mathcal{J}_{G_{1 \rightarrow 2}} = 1$ .
- Analogamente, do modelo 2 para o modelo 1, definimos  $G_{2 \rightarrow 1}(\theta_1, \theta_2) = (\theta_1, \theta_2)$ , onde  $\mathcal{J}_{G_{2 \rightarrow 1}} = 1$ .
- Como probabilidade de *jump* entre modelos, tomamos o valor 0.25, e probabilidade  $1 - 0.25$  de dar um passo no Metropolis-Hastings tradicional. Ainda, calculamos a proporção de visitas a cada modelo.

Como valores inicial para  $\theta$ , tomamos  $\theta = 0$  e começamos amostrando do modelo 1. Em 100 mil iterações, os resultados foram os seguintes:

Taxa de aceitação: 45.35% (nº de jumps / nº de tentativas = 11343/25013)

Proporção de visitas em cada modelo: 2.506628

A seguir, são apresentados alguns gráficos das cadeias geradas.

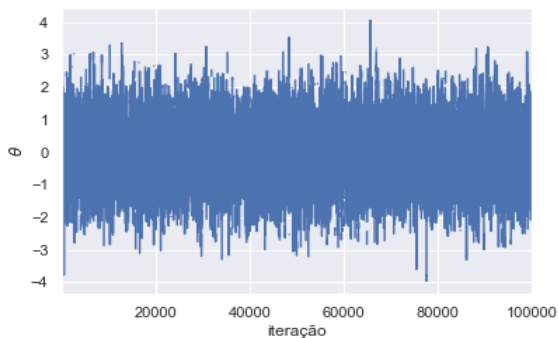


Figura 1: Cadeia do modelo 1

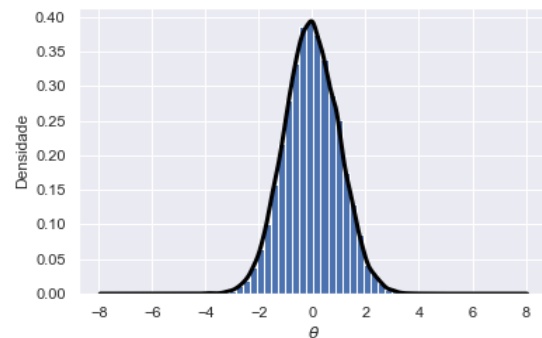
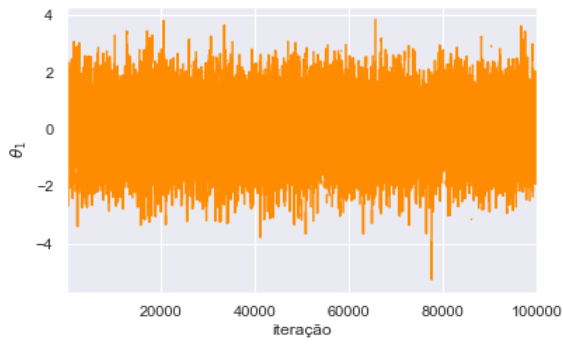
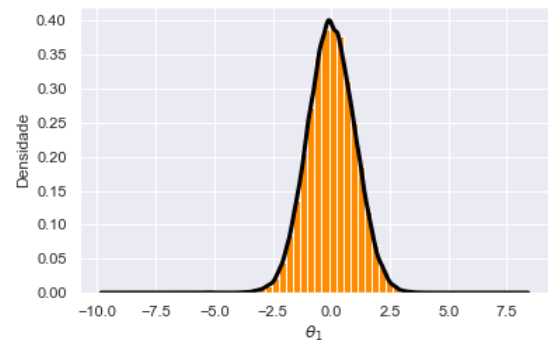
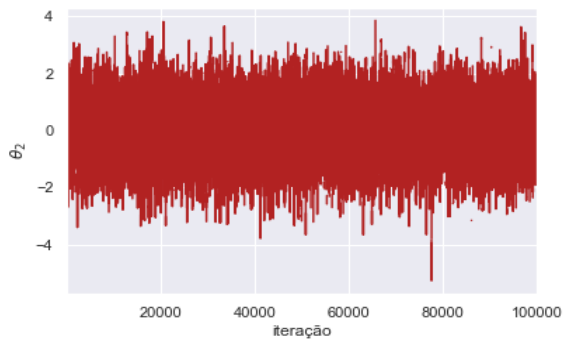
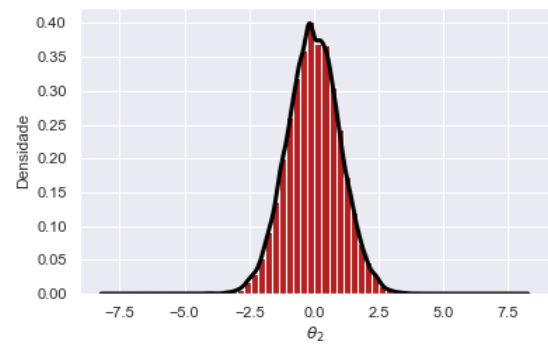


Figura 2: Distribuição (cadeia do modelo 1)

Agora, para os parâmetros do modelo 2:

Figura 3: Cadeia do modelo 2 ( $\theta_1$ )Figura 4: Distribuição de  $\theta_1$ Figura 5: Cadeia do modelo 2 ( $\theta_2$ )Figura 6: Distribuição de  $\theta_2$ 

Para termos uma noção da frequência dos *jumps*, vamos ver os gráficos a seguir.

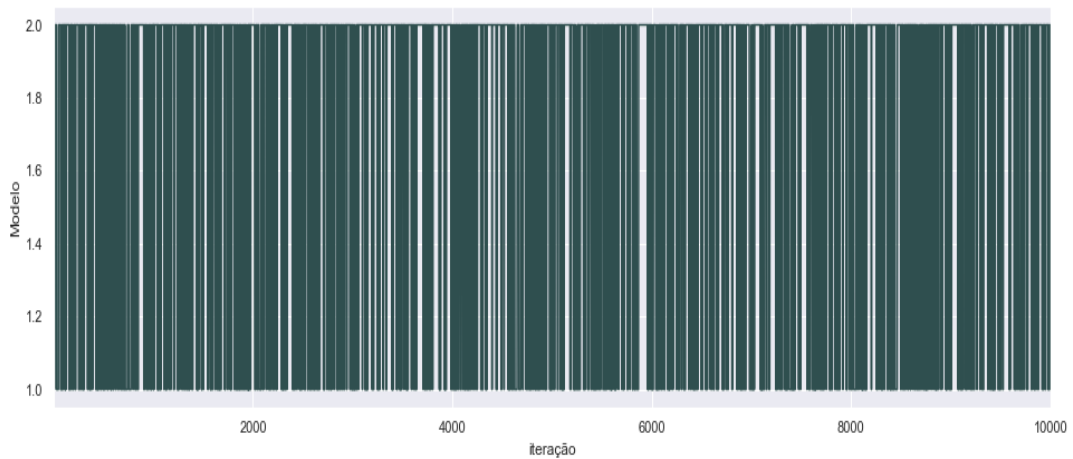


Figura 7: Modelo amostrado em cada iteração

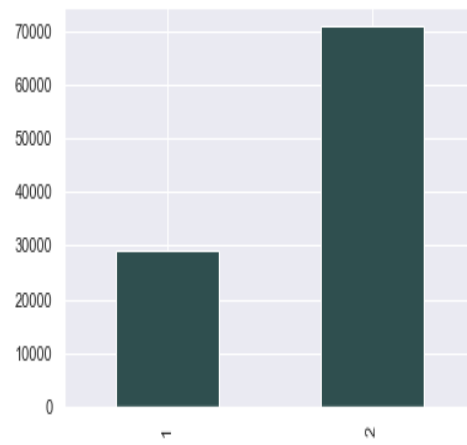
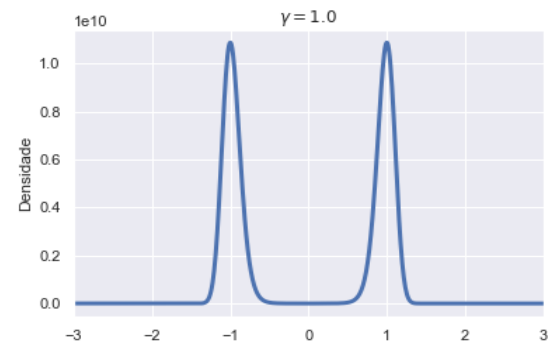
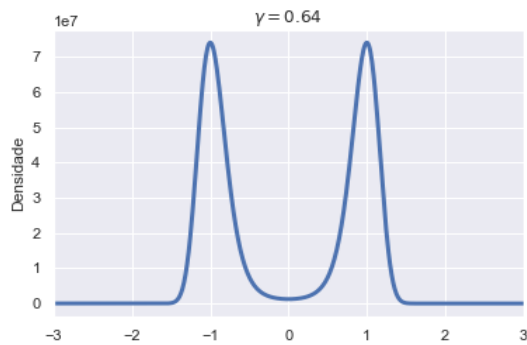
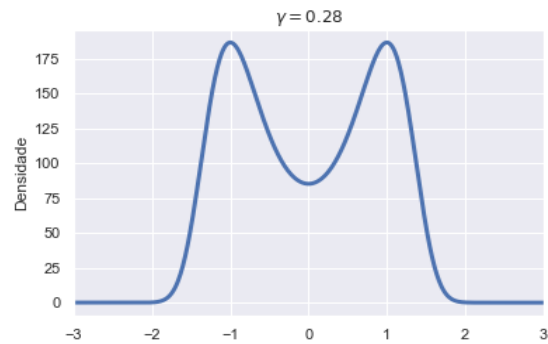
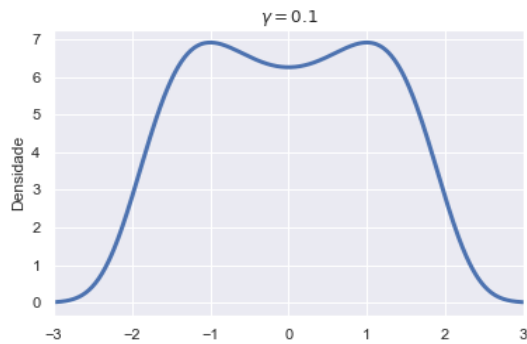


Figura 8: N° de amostragens de cada modelo

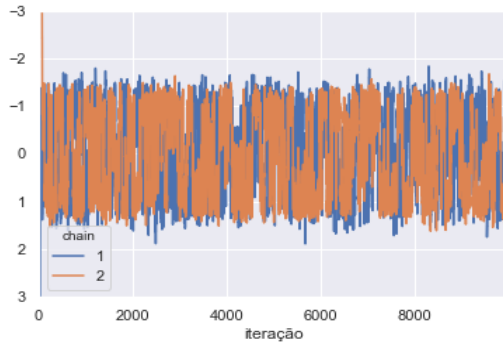


## Simulation question (Parallel Tempering)

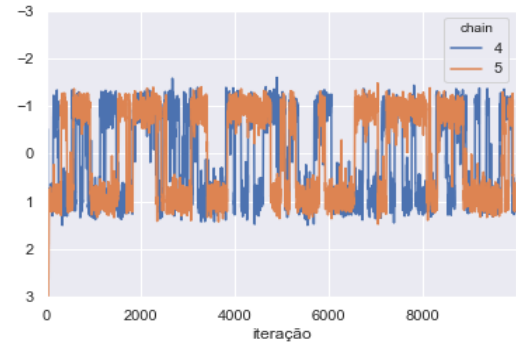
Primeiramente, vamos visualizar a distribuição target para os valores de  $\gamma_k = (0.1, 0.28, 0.64, 1.0)$ .



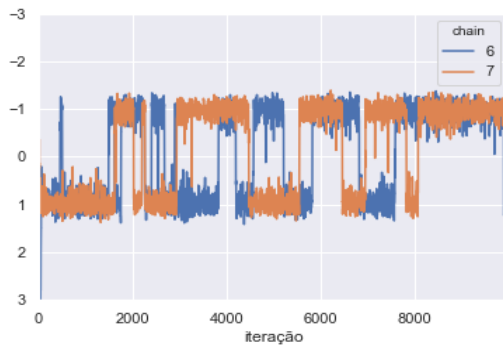
Como é fácil de ver nos gráficos, a distribuição *target* tem modas em -1 e 1. Agora, vamos definir  $N$  como sendo o número de cadeias, que coincide com o número de elementos na sequência  $(\gamma_k)$ , onde  $0 < \dots < \gamma_k < \dots < 1, \forall k$ . Para visualizar como as cadeias se misturam para diferentes valores de  $\gamma$ , tomamos 10 elementos igualmente espaçados da sequência  $\gamma_k$ . Os resultados podem ser analisados nas figuras a seguir:



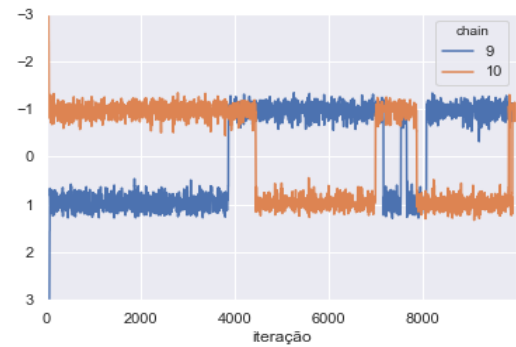
Cadeias 1 e 2



Cadeias 4 e 5



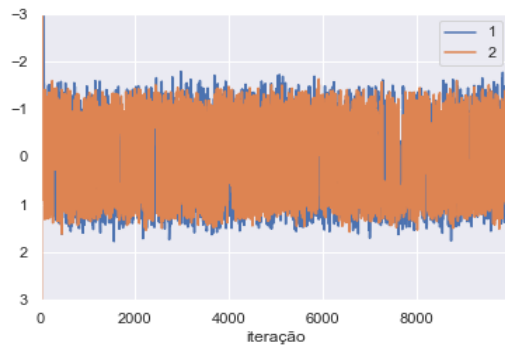
Cadeias 6 e 7



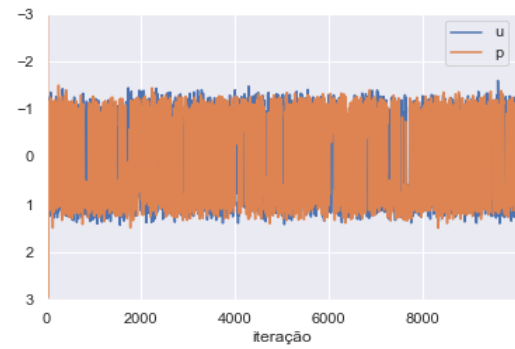
Cadeias 9 e 10

O que percebemos facilmente é que as cadeias são mais misturadas para valores pequenos de  $\gamma$ . Para altos valores de  $\gamma$ , as cadeias tendem a não se misturar muito. A ideia do *parallel tempering* é, então, permitir que as cadeias para baixo valor de  $\gamma$  proponham pontos para as cadeias com alto valor de  $\gamma$ . Assim, seguindo a sugestão do exercício, amostramos  $k, l \in \{1, \dots, N\}$  e propomos mudar de  $X_k$  para  $X_l$  com probabilidade  $\min\left(1, \frac{\pi^{\gamma_k}(x_l)\pi^{\gamma_l}(x_k)}{\pi^{\gamma_k}(x_k)\pi^{\gamma_l}(x_l)}\right)$ .

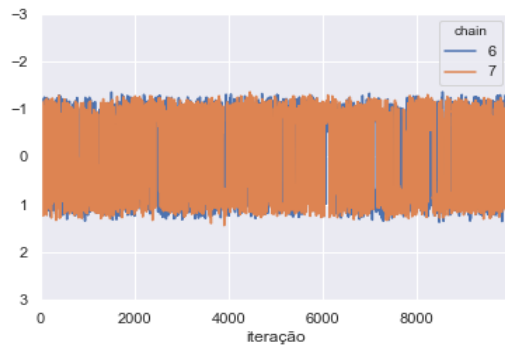
Os resultados desse novo algoritmo são expostos nas figuras a seguir:



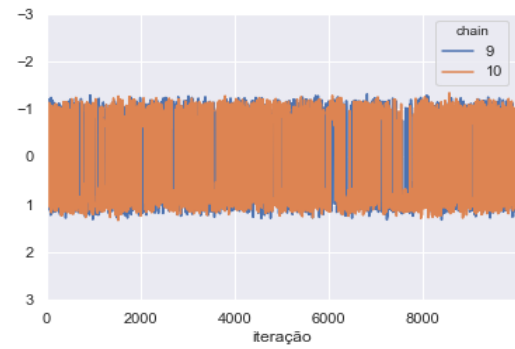
Cadeias 1 e 2



Cadeias 4 e 5

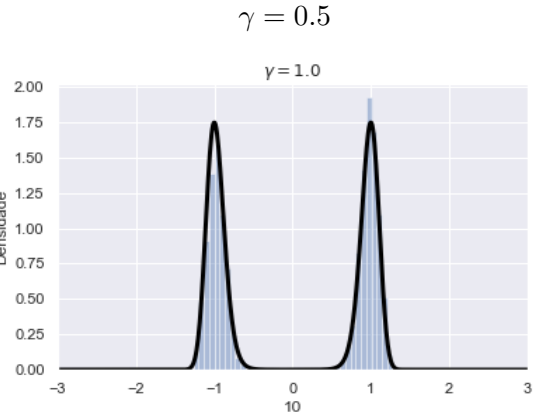
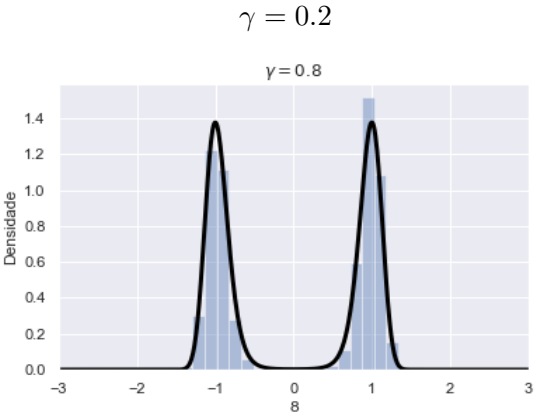
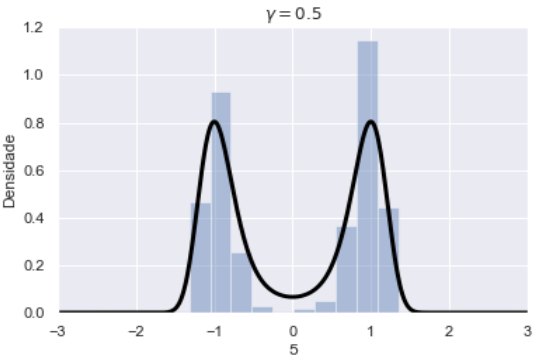
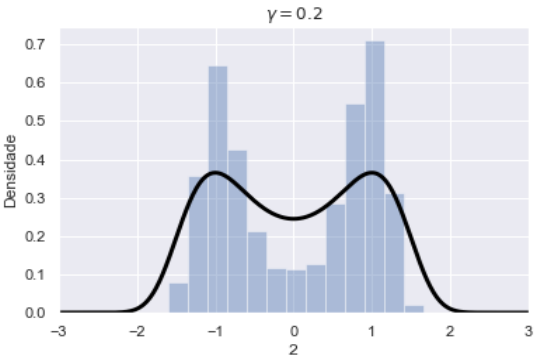


Cadeias 6 e 7



Cadeias 9 e 10

Por fim, vamos sobrepor os gráficos das distribuições *target* exibidos no começo deste exercício com as distribuições das cadeias para, visualmente, termos uma intuição do quão bem esse algoritmo conseguiu amostrar da distribuição *target* para diferentes valores de  $\gamma$ .



**Simulation question (linear Gaussian model - SIS and SIR)****1.**

Assim como solitido, foram simuladas  $T = 100$  observações do modelo Gaussiano proposto, considerando:

- $X_0 \sim \mathcal{N}(0, 1)$
- $V_t, W_t \stackrel{iid}{\sim} \mathcal{N}(0, 1)$
- $\phi = 0.95$
- $\sigma_V = \sigma_W = 1$

O gráfico das variáveis de interesse  $X$  e  $Y$  simuladas segue abaixo.

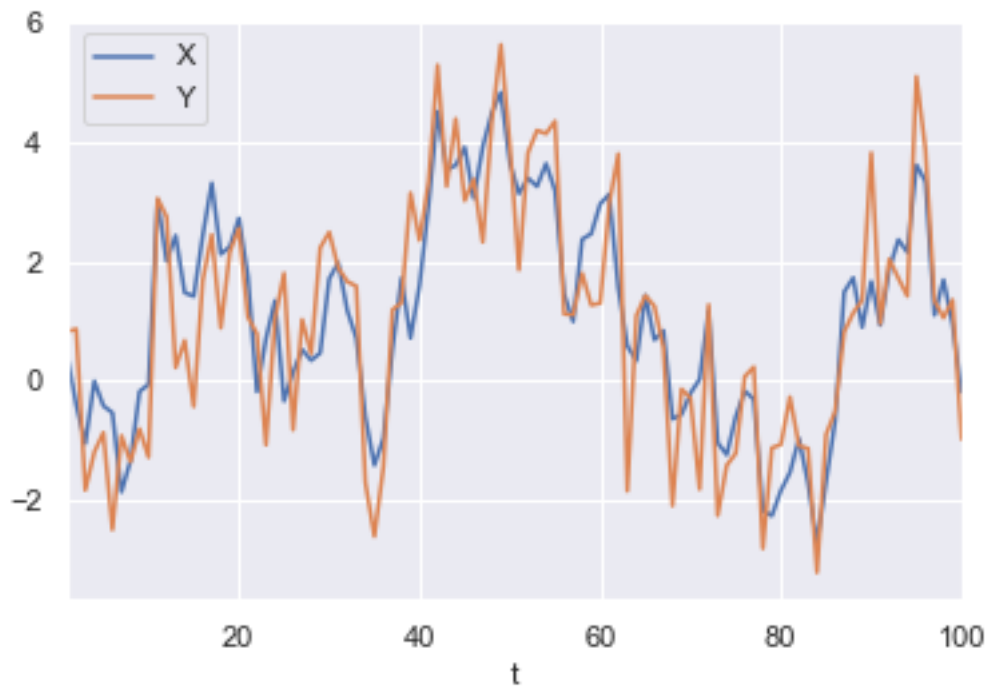


Figura 9:  $X_t = \phi X_{t-1} + \sigma_V V_t$  e  $Y_t = X_t + \sigma_W W_t$

2.

Como estamos considerando um modelo canônico Gaussiano, podemos aplicar o filtro de Kalman para calcular as médias  $\mathbb{E}[X_t | Y_{0:t}]$ . O resultado é exibido abaixo.

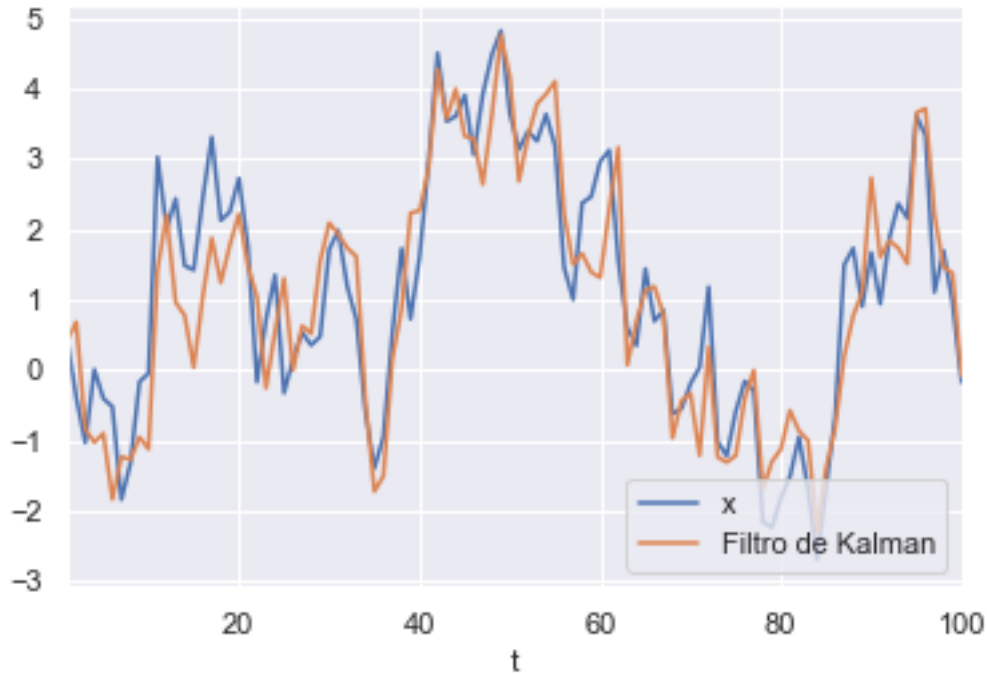
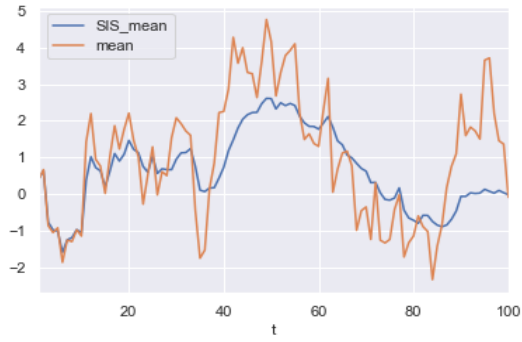
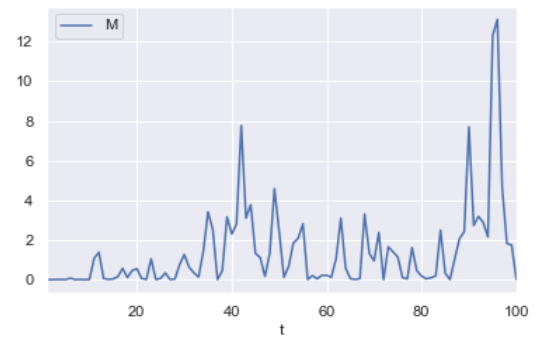
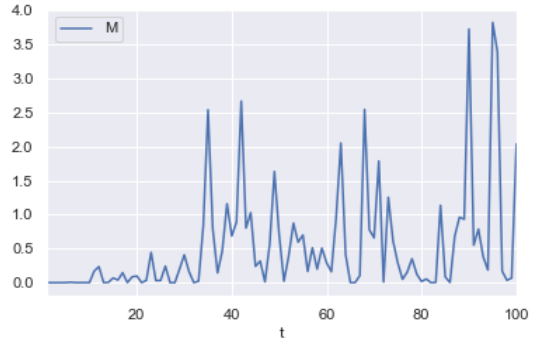
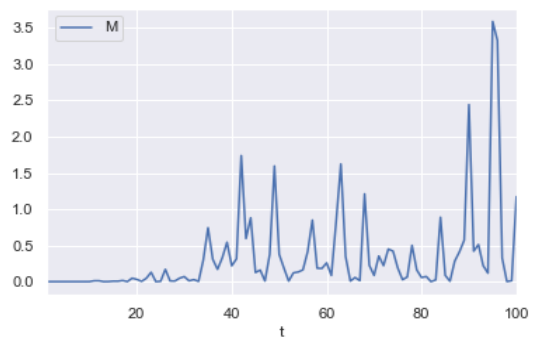
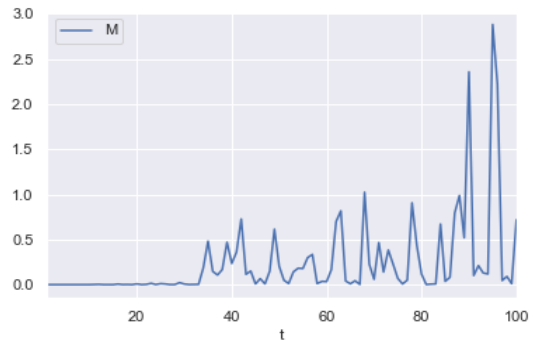


Figura 10:  $X_t$  e  $\mathbb{E}[X_t | Y_{0:t}]$

3.

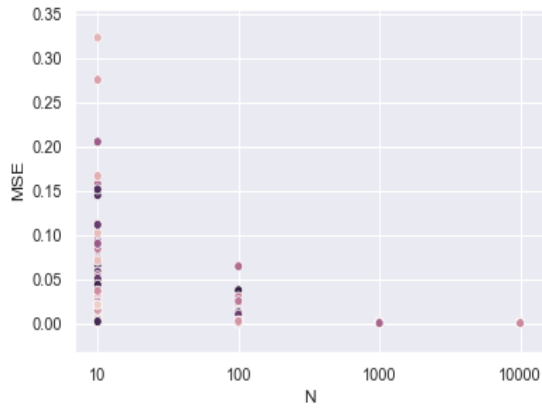
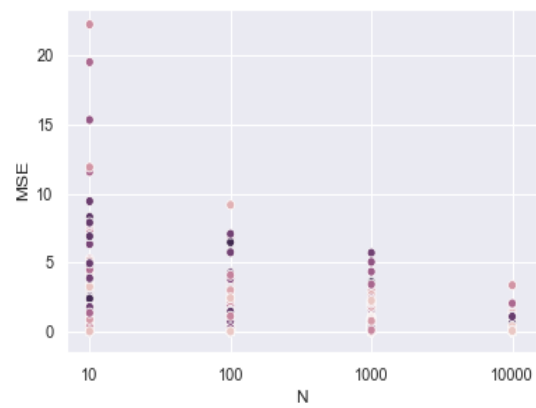
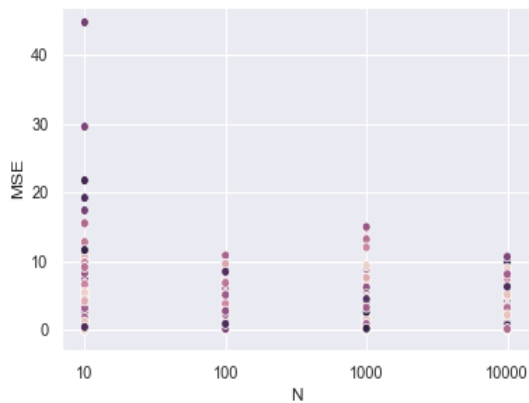
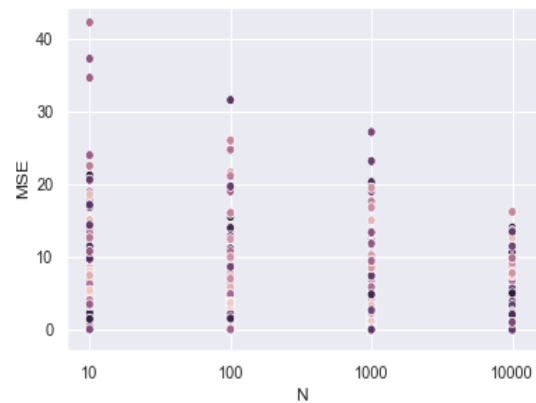
Conforme solicitado no enunciado, vamos aplicar o algoritmo de *Sequential Importance Sampling* analisar a variância dos estimadores para os valores de  $N = (10, 100, 1000, 10000)$  - quantidade de partículas - e para todos os valores de  $t = 1, \dots, 100$ . A intenção do exercício é comparar os resultados obtidos pelo algoritmo *SIS* com o filtro de Kalman.

Nos gráficos a seguir, conseguimos visualizar a média calculada pelo método SIS e pelo filtro de Kalman, assim como o *Mean Squared Error* em função do tempo. É fácil notar que o *MSE* aumenta consideravelmente com o tempo.

SIS mean vs. Kalman Filter ( $N = 10$ )MSE ( $N = 10$ )SIS mean vs. Kalman Filter ( $N = 100$ )MSE ( $N = 100$ )SIS mean vs. Kalman Filter ( $N = 1000$ )MSE ( $N = 1000$ )SIS mean vs. Kalman Filter ( $N = 10000$ )MSE ( $N = 10000$ )

4.

Como indicador de performance do SIS, vamos avaliar o  $MSE$ . Podemos ver que, para  $t$  pequeno, um maior número de partículas ajuda a reduzir o erro. Já para  $t$  grande, o ganho em aumentar o número de partículas não é tão significativo.

MSE (partículas em  $t = 1$ )MSE (partículas em  $t = 33$ )MSE (partículas em  $t = 66$ )MSE (partículas em  $t = 100$ )

5.

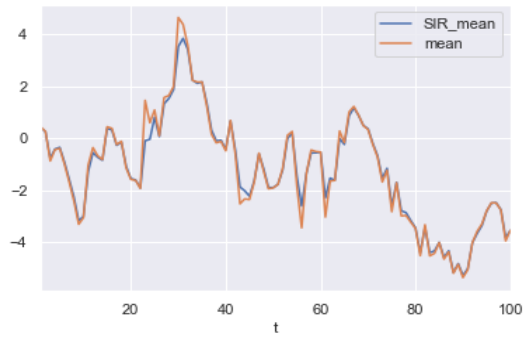
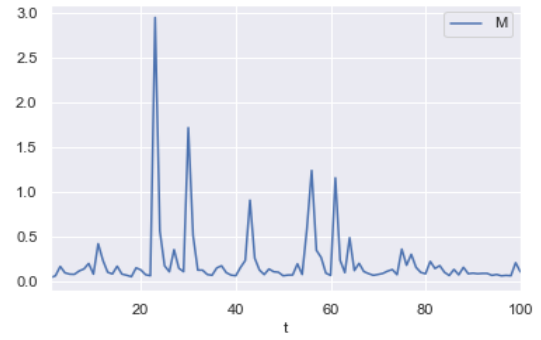
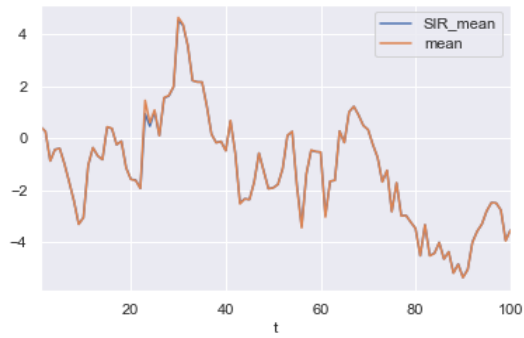
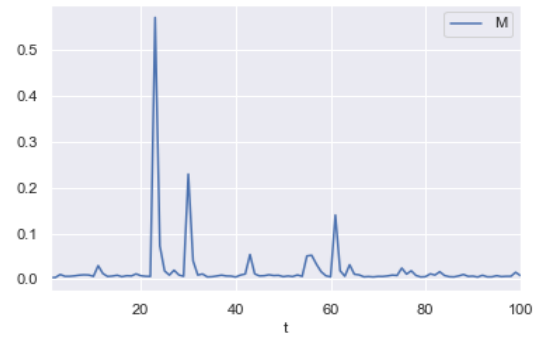
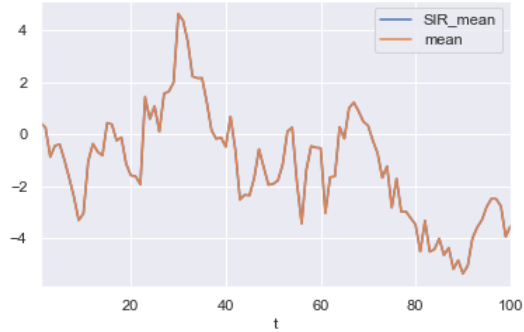
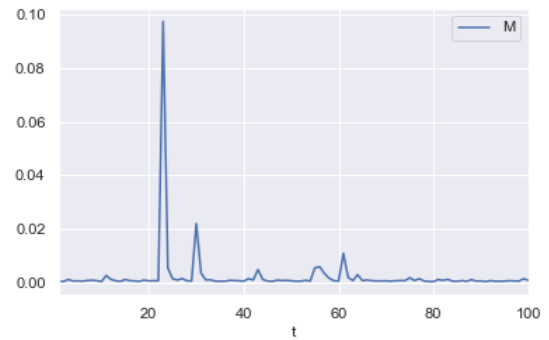
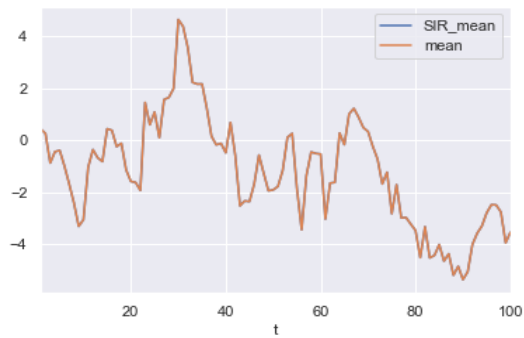
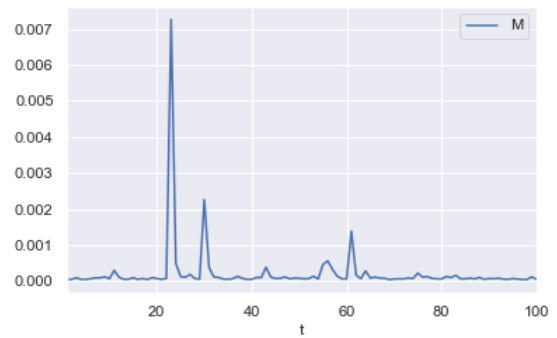
Os gráficos desse exercício estão no item 3 (de título "SIS mean vs. Kalman Filter"), onde foram tomadas as médias das partículas como estimativa de Monte Carlo, e  $N$  é o número de partículas.

6.

Nesse item, vamos apresentar os mesmos gráficos expostos no item 3, agora para o algoritmo proposto no exercício (SIR - *Sequential Importance Resampling*) e comparar com a média, calculada pelo Filtro de Kalman.

Nesses gráficos, vemos que o erro quadrático médio (MSE) reduz com o tempo.



SIR mean vs. Kalman Filter ( $N = 10$ )MSE ( $N = 10$ )SIR mean vs. Kalman Filter ( $N = 100$ )MSE ( $N = 100$ )SIR mean vs. Kalman Filter ( $N = 1000$ )MSE ( $N = 1000$ )SIR mean vs. Kalman Filter ( $N = 10000$ )MSE ( $N = 10000$ )