

# Sorry arima, I'm going Bayesian

Pierre Gauthier

École des Mines de Nancy

May 2019

Tuteur : Denis Villemonais

# Introduction

- Projet à partir du billet *Sorry ARIMA, but I'm Going Bayesian* par Kim Larsen. S'appuie sur plusieurs publications :
  - ▶ Predicting the Present with Bayesian Structural Time Series, S.L. Scott and H. Varian
  - ▶ Variable Selection Via Gibbs sampling , E.I. George, R.E. McCulloch
  - ▶ A simple and efficient simulation smoother, J. Durbin, S.J. Koopman
- Étudier une autre approche que celle classique pour les séries temporelles
- Utilisation des méthodes de Monte-carlo par chaîne de Markov.

# Sommaire

- 1 Méthode de Monte-Carlo par chaînes de Markov
- 2 Approche bayésienne pour la modélisation des series temporelles
- 3 Modélisation d'un jeu de données

# Approche bayésienne pour les séries temporelles

# Approche bayésienne pour la modélisation des séries temporelles

## ► Les modèle espace-états

		bruit blanc gaussien
equation d'observation	$y_t = Z_t^T \alpha_t + \epsilon_t$	$\epsilon_t \sim N(0, H_t)$
equation de transition	$\alpha_{t+1} = T_t \alpha_t + R_t \eta_t$	$\eta_t \sim N(0, Q_t)$

- $y_t$  observations
- $\alpha_t$  variables d'états / latentes / cachées
- $Z_t$  matrice de mesure
- $T_t$  matrice de transition

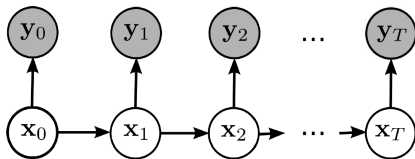


Figure – Hidden Markov Chain[researchgate.net]

# Approche bayésienne pour la modélisation des séries temporelles

→ Bayesian structural time series (BSTS)

bruit blanc gaussien

observation	$y_t = \mu_t + \beta^T x_t + \tau_t + \varepsilon_t$	$\varepsilon_t \sim N(0, \sigma_\varepsilon^2)$
regression	$\beta^T x_t$	
tendance + marche aléatoire	$\mu_t = \mu_{t-1} + \delta_{t-1} + u_t$	$u_t \sim N(0, \sigma_u^2)$
marche aléatoire	$\delta_t = \delta_{t-1} + v_t$	$v_t \sim N(0, \sigma_v^2)$
saisonnalité	$\tau_t = -\sum_{s=1}^{s-1} \tau_{t-s} + w_t$	$w_t \sim N(0, \sigma_w^2)$

→ Bayesian structural time series (BSTS)

observation	$y_t = Z_t^T \alpha_t + \epsilon_t$	$\epsilon_t \sim N(0, H_t)$
	$Z_t^T$ $(1 \ 0 \ \beta^T \mathbf{x}_t)$	$\alpha_t^T$ $(\mu_t \ \delta_t \ 1)^T$
equation de transition	$\alpha_{t+1} = T_t \alpha_t + R_t \eta_t$	$\eta_t \sim N(0, Q_t)$
$\alpha_t$ $(\mu_t \ \delta_t \ 1)$	$T_t$ $(1 \ 1 \ 0)$ $(0 \ 1 \ 0)$ $(0 \ 0 \ 1)$	$N_t \eta_t$ $(u_t \ v_t \ w_t)$

→ estimation des paramètres

## Loi à postériori états cachés $\alpha_t$ : Le filtre de Kalman

Itérations sur l'estimation  $p(\alpha_t|y_{1:t}) \sim \mathcal{N}(\hat{\alpha}_t, P_t)$

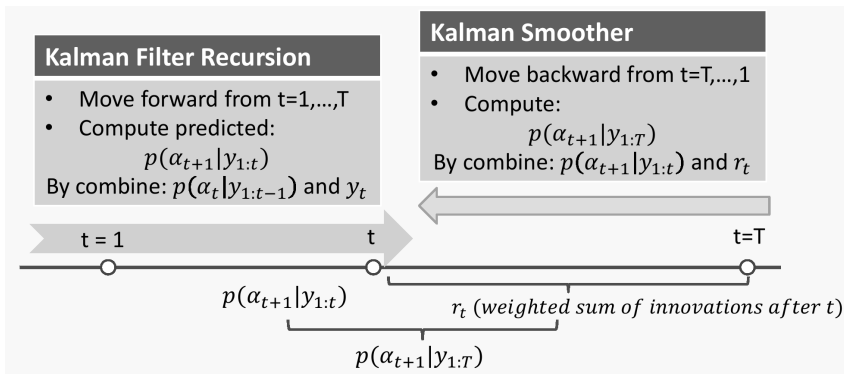


Figure – [github : anhdanggit/nowcasting-google-queries/]



## Loi à postérieure de $\beta$ : *spike-and-slab prior*

- On prend la partie regression  $y_t^* = y_t - \mu_t$
- On utilise pour  $\beta$  une distribution à priori *spike-and-slab* :

$$\blacktriangleright p(\gamma) = \prod_{k=1}^N \pi^{\gamma_k} (1 - \pi)^{1-\gamma_k}, \quad \gamma_k \in \{0, 1\} \quad N = \text{Card}(\mathbf{x})$$

$$\blacktriangleright \text{À priori : } p(\beta, \gamma, \sigma_\epsilon^2) = p(\beta_\gamma | \gamma, \sigma_\epsilon^2) p(\sigma_\epsilon^2 | \gamma) p(\gamma)$$

$$\blacktriangleright \beta_\gamma | \sigma_\epsilon^2, \gamma \sim \mathcal{N}(b_\gamma, \sigma_\epsilon^2 (\Omega_\gamma^{-1})^{-1}) \quad \sigma_\epsilon^2 | \gamma \sim \text{IG}\left(\frac{\nu}{2}, \frac{ss}{2}\right)$$

paramètres à priori :  $\nu$  nombre de paramètres,  $\frac{ss}{\nu} = (1 - R^2) s_y^2$ ,  $\Omega^{-1} \propto X^T X$

- On utilise les propriétés des lois conjuguées pour obtenir les lois à postérieures

$$\beta_\gamma | \sigma_\epsilon, \gamma, \mathbf{y}^* \quad \gamma_\epsilon^2 | \gamma, \mathbf{y}^* \quad \gamma | \mathbf{y}^*$$

- Intérêt de la *spike-and-slab*

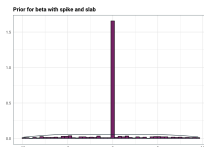


Figure –  
[batisengul.co.uk]

$$\beta_\gamma = \beta[\gamma_k \neq 0]$$

## Échantillonneur de Gibbs pour BSTS : SSVS algorithm

$$\Theta = (\gamma, \beta, \sigma_\varepsilon^2, \sigma_v^2, \sigma_u^2)$$

► Choisir paramètres à priori  $v, R^2, s_y^2, \pi$

► Tirer  $\gamma, \beta, \sigma_\varepsilon^2, \sigma_v^2, \sigma_u^2$   $\sigma_u^2, \sigma_v^2, \sigma_w^2$  sont tiré selon la loi  $\gamma \sim IG\left(\frac{\nu}{2}, \frac{ss}{2}\right)$

Sur  $1, \dots, M$  :

**1** Après application du filtre de Kalman, on tire les états latents  $\alpha$  depuis  $p(\alpha|y, \gamma, \beta, \sigma_\varepsilon^2, \sigma_v^2, \sigma_u^2)$

**2** On tire  $\sigma_u^2$  et  $\sigma_v^2$  selon  $p\left(\frac{1}{\sigma_u^2}, \frac{1}{\sigma_v^2} | y, \alpha, \beta, \sigma_\varepsilon^2\right)$

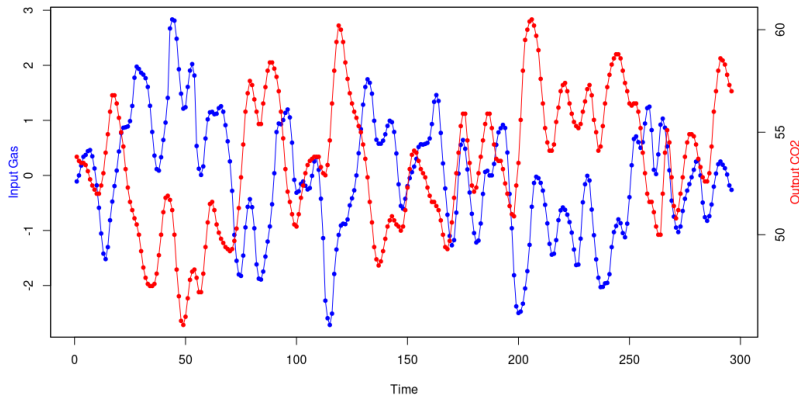
**3** On tire  $\beta$  et  $\sigma_\varepsilon^2$  selon  $p(\beta, \sigma_\varepsilon^2 | y, \alpha, \sigma_u^2, \sigma_v^2)$

On prend comme modèle la moyenne des tirages  $(\Theta^1, \dots, \Theta^M)$

## Utilisation du modèle BSTS sur un jeu de données

On prend le jeu de données **CO2** comprenant 295 observations

- ▶ *Input Gas* : arrivée d'essence
- ▶ *Output CO2* : CO2 en sortie



On utilise le modèle précédent pour exprimer *Input Gas* en fonction de *Output CO2*.

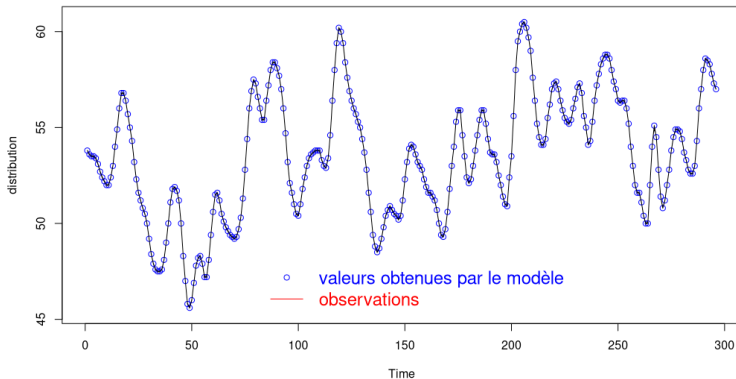
```
library(bsts)
ss <- AddLocalLinearTrend(list(), CO2$output.co2)
bsts.reg <- bsts(output.co2 ~ ., state.specification =
  ss, data =
  CO2, niter = 2000, ping=0, seed=2016)
```

$$y_t = \mu_t + \beta^T \mathbf{x}_t + \epsilon_t \quad \epsilon_t \sim N(0, \sigma_\epsilon^2)$$

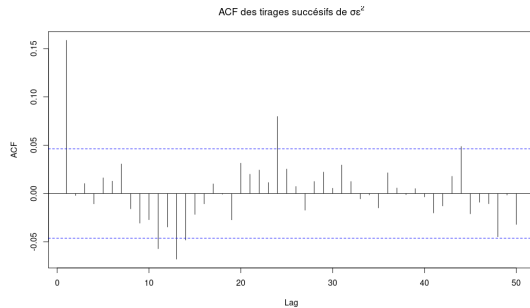
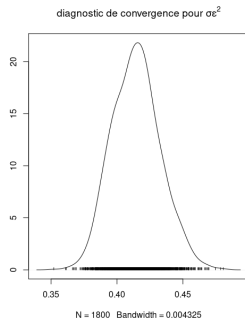
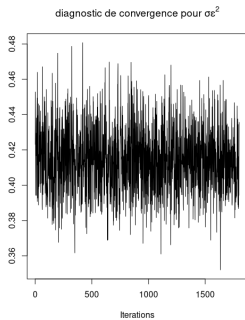
$$\mu_t = \mu_{t-1} + \delta_{t-1} + u_t \quad u_t \sim \mathcal{N}(0, \sigma_u^2)$$

### Listing 1 — package *bsts*

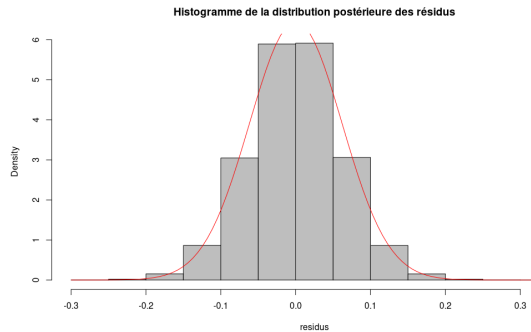
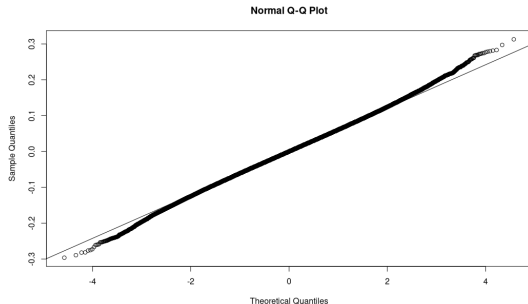
Observations et valeurs obtenues par le modèle *bsts.reg*



# Convergence



# Résidus



```
> MAPE=mean(abs(colMeans(bsts.reg$one.step.prediction.errors)))/C02$output.co2)
> MAPE
[1] 0.005890585
> MAPE
```

## Comparaison avec un modèle à fonction de transfert

Modèle à fonction de transfert :  $Y_t = \mu + \frac{\Omega(B)}{\Delta(B)} X_{t-b} + \frac{\Theta(B)}{\Phi(B)} \varepsilon_t$

Avec :

- $(Y_t)$  chronique à modeliser
- $(X_t)$  chronique explicative
- $\frac{\Omega(B)}{\Delta(B)}$  fonction de transfert
- $(u_t)$  chronique des erreurs

→ On blanchit la chronique  $X_t$  :  $\chi_t = \frac{\Phi_1(B)}{\Theta_1(B)} X_t$

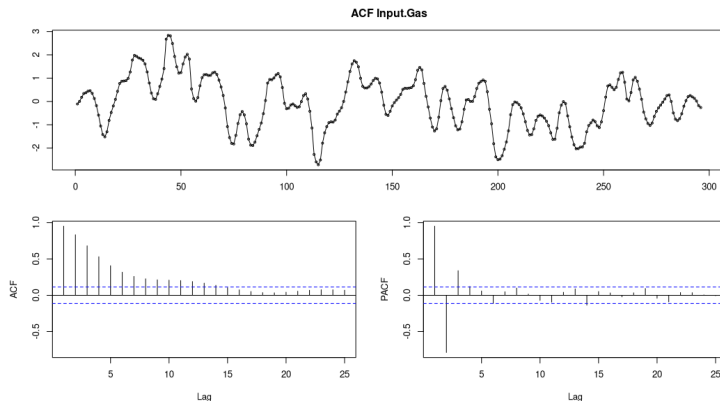
→  $\Upsilon_t = \frac{\Phi_1(B)}{\Theta_1(B)} Y_t = \frac{\Omega(B)}{\Delta(B)} \chi_{t-b} + \tilde{\varepsilon}_t$

On utilise le corrélogramme croisé  $\rho(h) = \frac{\text{Cov}(\chi_t, \Upsilon_{t+h})}{\sqrt{\text{Var}(\chi_t) \cdot \text{Var}(\Upsilon_t)}} = \begin{cases} \nu_h \frac{\sigma_\chi}{\sigma_r} & \text{si } h \geq 0 \\ 0 & \text{si } h < 0 \end{cases}$

où  $\Upsilon_t = \sum_{h \geq 0} \nu_h \chi_{t-h} + \tilde{\varepsilon}_t$

# Application sur le jeu de données CO2

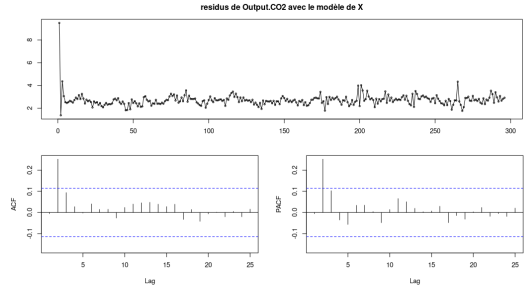
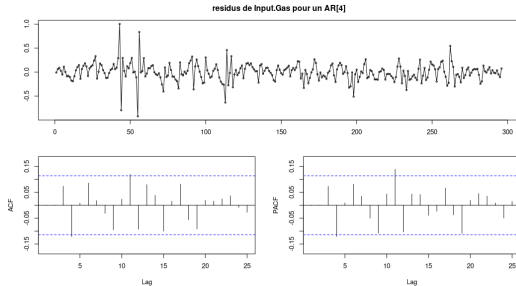
→ blanchissement de *Input Gas*.





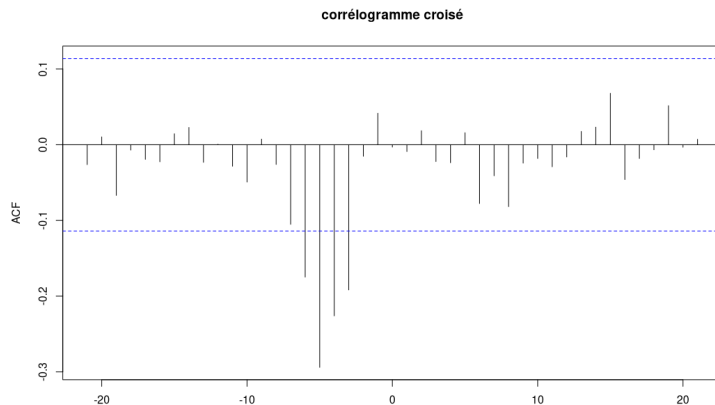
# Application sur le jeu de données CO2

→ blanchissement de *Input Gas*.



# Application sur le jeu de données CO2

→ Déterminer la fonction de transfert



## Application sur le jeu de données CO2

→ Le meilleur modèle est un  $AR[2]$  au numérateur et un polynome de degré 4 au dénominateur

```
> output_XY=arimax(Y,order=c(2,0,0),transfer=list(c(1,5)),fixed=c(NA,NA,NA,NA,0,0,0,NA,NA,NA),xtransf=X)
> summary(output_XY)
Coefficients:
      ar1      ar2  intercept  T1-AR1  T1-MA0  T1-MA1  T1-MA2  T1-MA3  T1-MA4
s.e.  1.5272 -0.6288   53.3618  0.5490      0      0      0  -0.5310  -0.3801
      T1-MA5
s.e.  0.0467  0.0495   0.1375  0.0392      0      0      0   0.0738   0.1017
      T1-MA5
s.e.  -0.5180
      0.1086

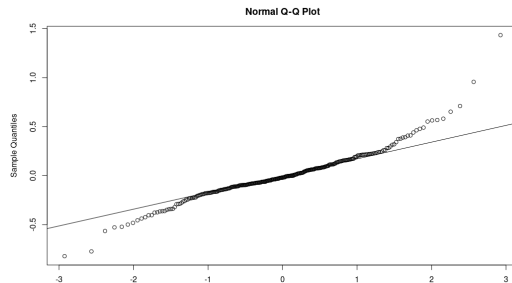
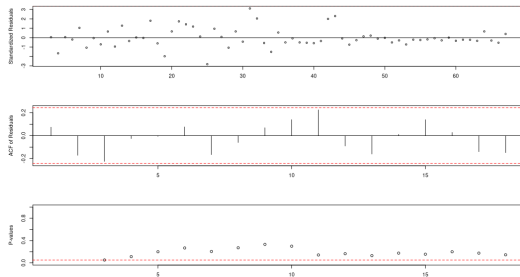
sigma^2 estimated as 0.0571:  log likelihood = 2.08,  aic = 9.83

Training set error measures:
      ME      RMSE      MAE      MPE      MAPE      MASE      ACF1
Training set 0.0001700879 0.2389594 0.1681788 -0.001732428 0.3130213 0.2806151 0.02877323
```

```
> coeftest(output_XY)
z test of coefficients:
      Estimate Std. Error z value Pr(>|z|)
ar1      1.527181   0.046723  32.6859 < 2.2e-16 ***
ar2     -0.628841   0.049471 -12.7114 < 2.2e-16 ***
intercept 53.361773  0.137503 388.0769 < 2.2e-16 ***
T1-AR1    0.549027  0.039191  14.0089 < 2.2e-16 ***
T1-MA3    -0.530964  0.073814  -7.1933 6.325e-13 ***
T1-MA4    -0.380125  0.101704  -3.7376 0.0001858 ***
T1-MA5    -0.518006  0.108562  -4.7715 1.829e-06 ***
```

# Application sur le jeu de données CO2

→ Diagnostic des résidus



# Conclusion

# Conclusion

- Arguments de l'auteur pour une approche bayésienne
- Bayésiens contre fréquentistes
- Les avantages de l'approche bayésienne

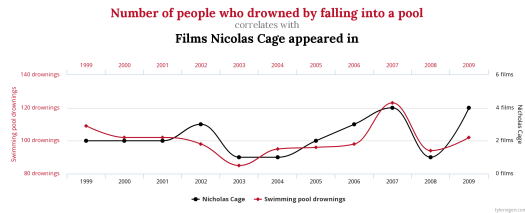


Figure – [<http://www.tylervigen.com/>]

Merci de votre attention