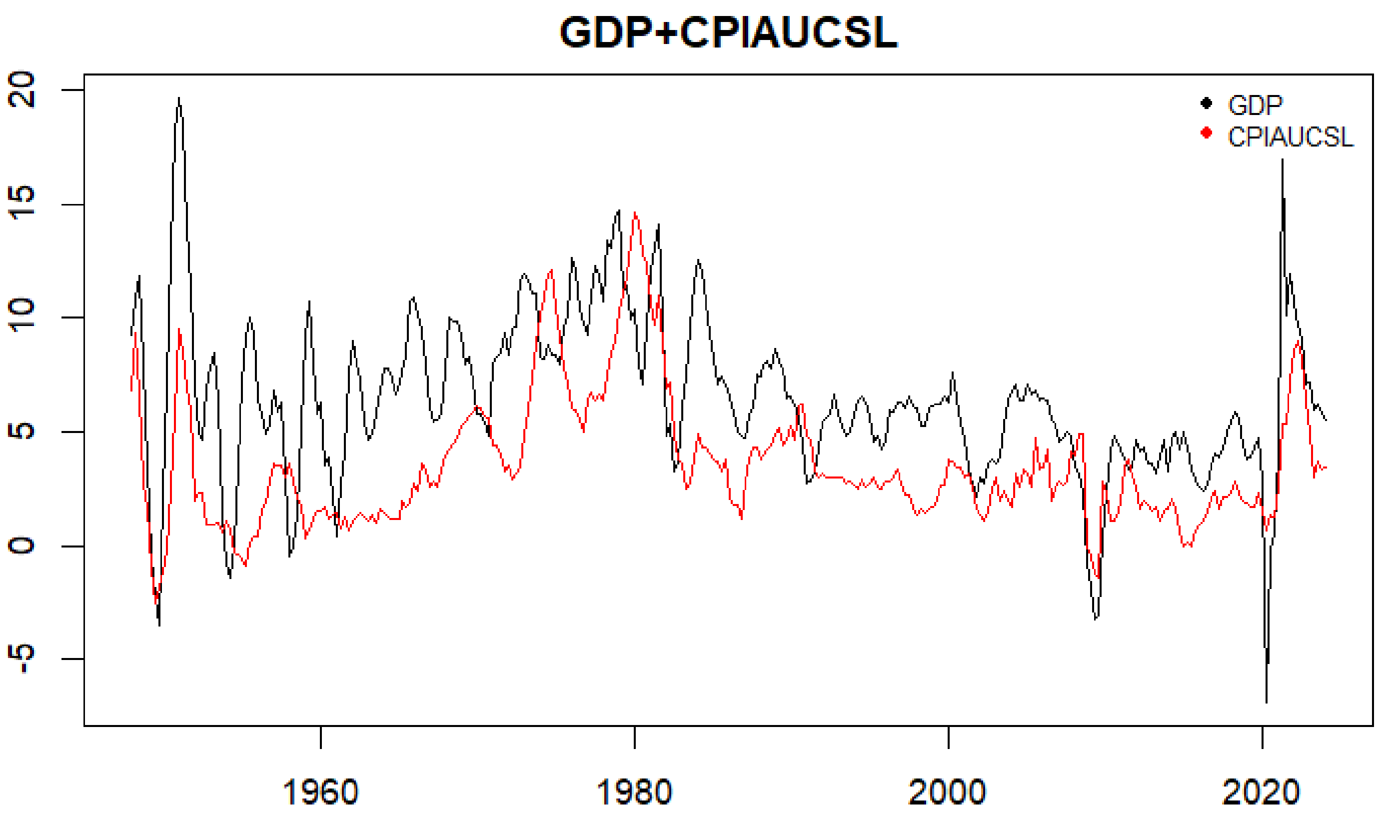
US GDP & Inflation Dataset

Bayesian Learning and Montecarlo Simulation

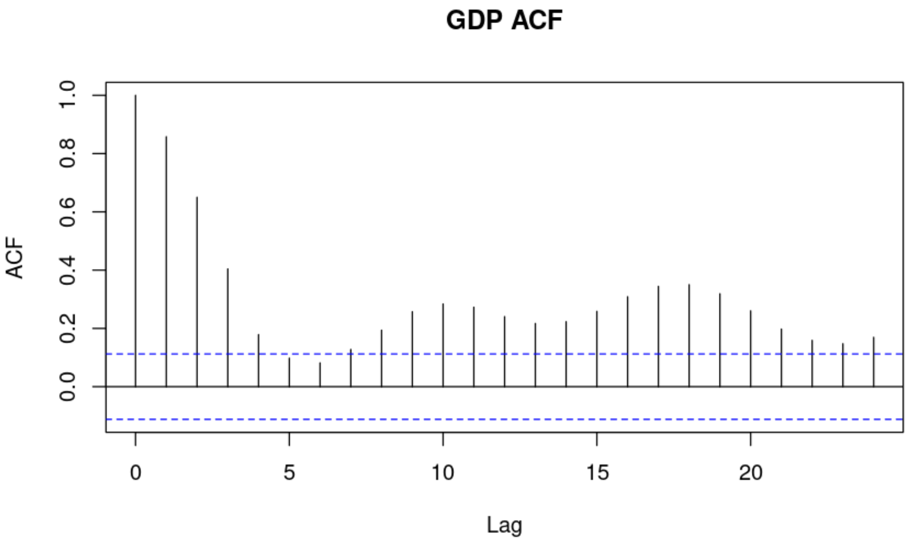
POLITECNICO
MILANO 1863

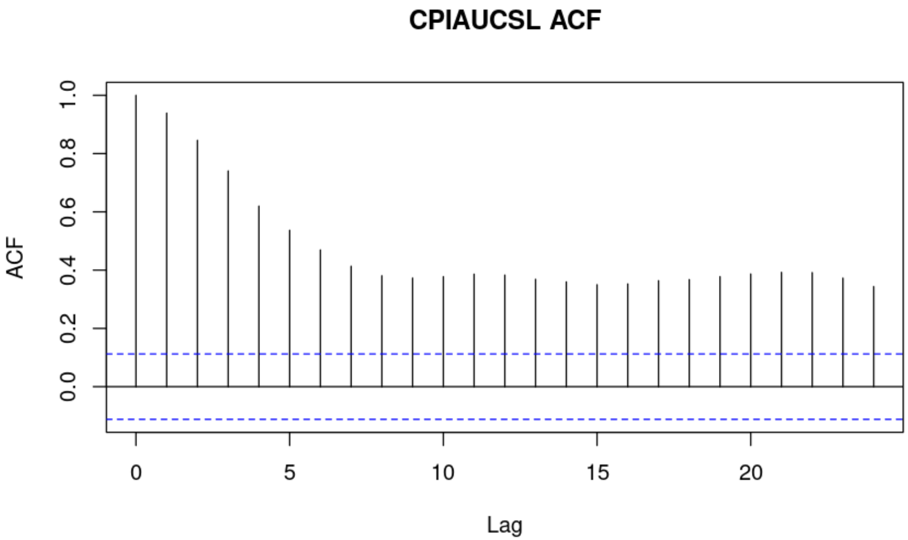
Problem Description

US GDP & CPIAUCSL
Data



GDP & CPIAUCSL
Autocorrelation plots





Project objectives

- Fit each time series independently using AR, MA, ARMA, and GARCH models.
- Fit the two time series jointly using a VAR model.
- Use these models for in-sample and out-of-sample predictions.
- Compare different models using DIC and WAIC criteria.

Jags settings

- 3 Chains.
- Total of 10,000 Iterations.
- 1,000 Burn-in Iterations.
- 10% of the data for comparison with out-of-sample predictions

POLITECNICO
MILANO 1863

Autoregressive Model (AR)

Autoregressive Model (AR)
General formulation

$$y_t = \mu_0 + \sum_{i=1}^{p} \alpha_i y_{t-i} + \epsilon_t$$

- p: number of past values considered
- μ_0: constant value
- α_i: model parameters
- ϵ_t: white noise
- $y_{t-1}, \dots, y_{t-p}$: past values

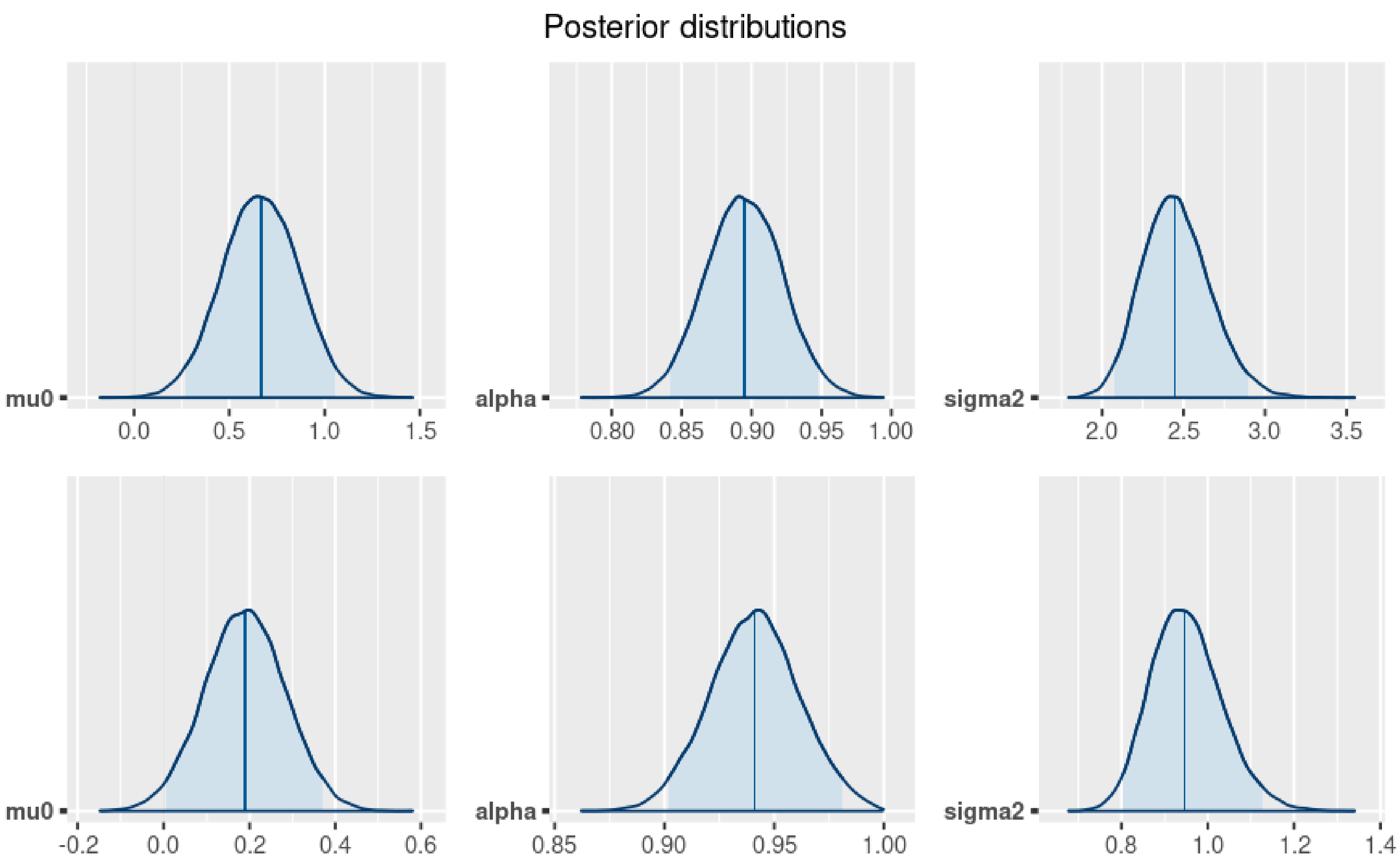
Autoregressive Model (AR)
AR(1)

- Model: $$y_t = \mu_0 + \alpha y_{t-1} + \epsilon_t$$
- Likelihood: $$y_t | \mu_0, \alpha, \sigma^2, y_{t-1} \sim \mathcal{N}(\mu_0 + \alpha y_{t-1}, \sigma^2)$$
- Priors: $$\begin{aligned} \mu_0 &\sim \mathcal{N}(0.0, 10000) \\ \tau = 1/\sigma^2 &\sim \mathcal{G}(2, 0.1) \\ \alpha &\sim \mathcal{U}(-1.0, 1.0) \end{aligned}$$

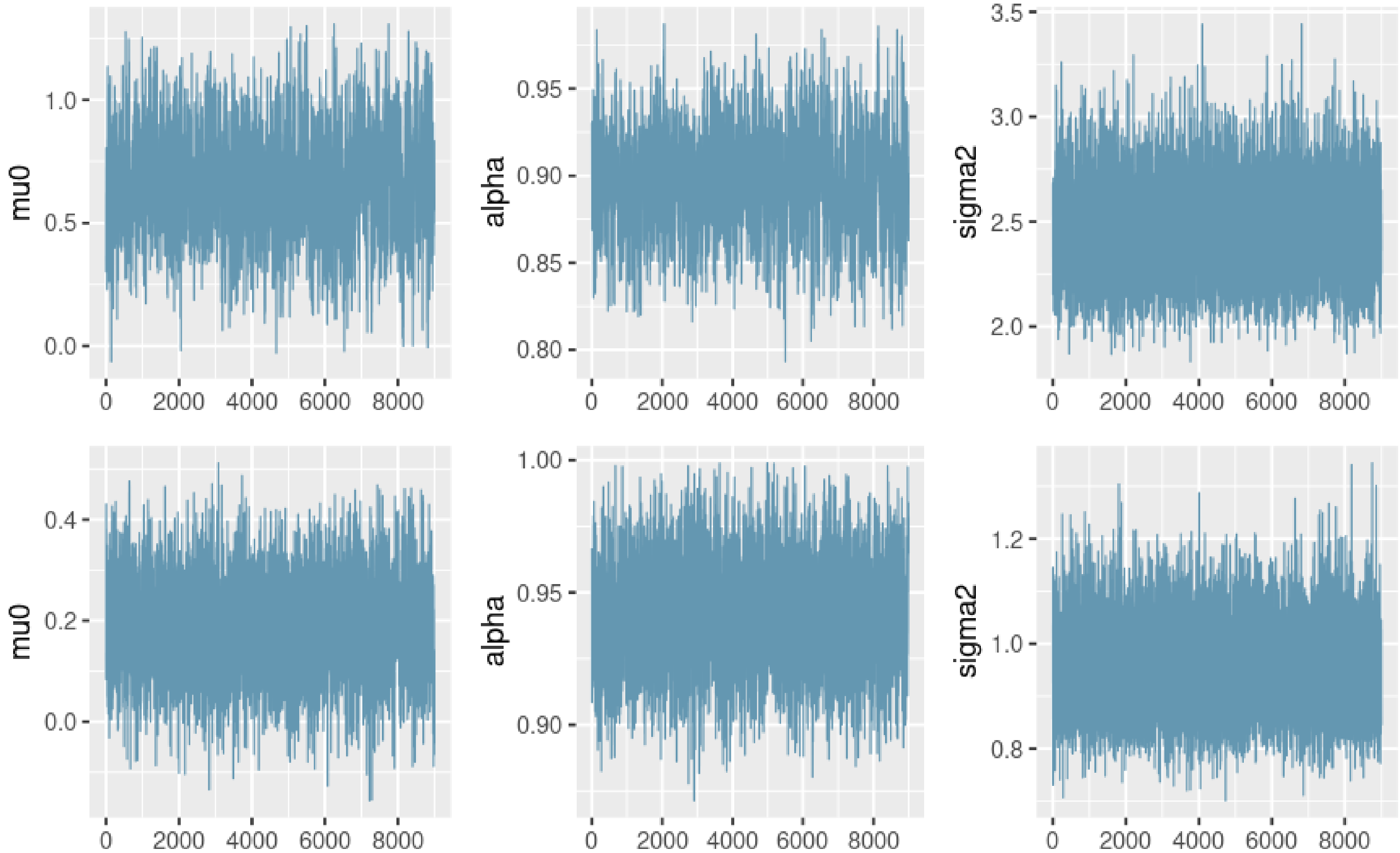
AR(1)
Jags

```
# Define model in JAGS
modelAR.string <-"model {
  ## Parameters: alpha, tau, mu0
  # Likelihood
  mu[1] <- Y[1]
  Yp[1] <- Y[1]
  for (i in 2:N) {
    Y[i]      ~  dnorm(mu[i], tau)
    mu[i]     <- mu0 + alpha * Y[i-1]
    Yp[i]     ~  dnorm(mu[i],tau)      # Prediction in sample
    LogLik[i] <- log(dnorm(Y[i], mu[i], tau))
  }

  # Prediction out of sample
  ypOut[1] ~ dnorm(mu0 + alpha * Y[N], tau)
  for(k in 2:Npred){
    ypOut[k] ~ dnorm(mu0 + alpha * ypOut[k-1], tau)
  }
  sigma2 <- 1/tau

  # Prior
  alpha ~ dunif(-1.0, 1.0)
  tau   ~ dgamma(2, 0.1)
  mu0   ~ dnorm(0.0, 1.0E-4)
}"
```

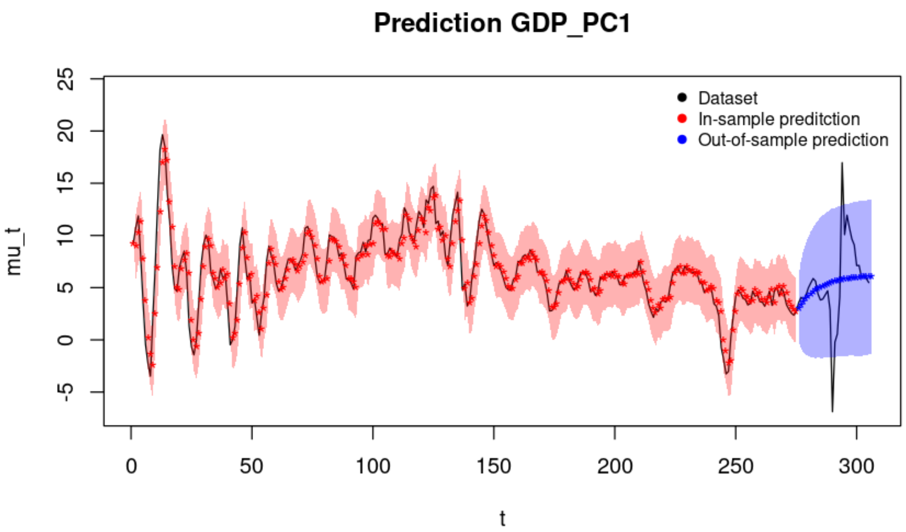
AR(1)
Posterior Distribution

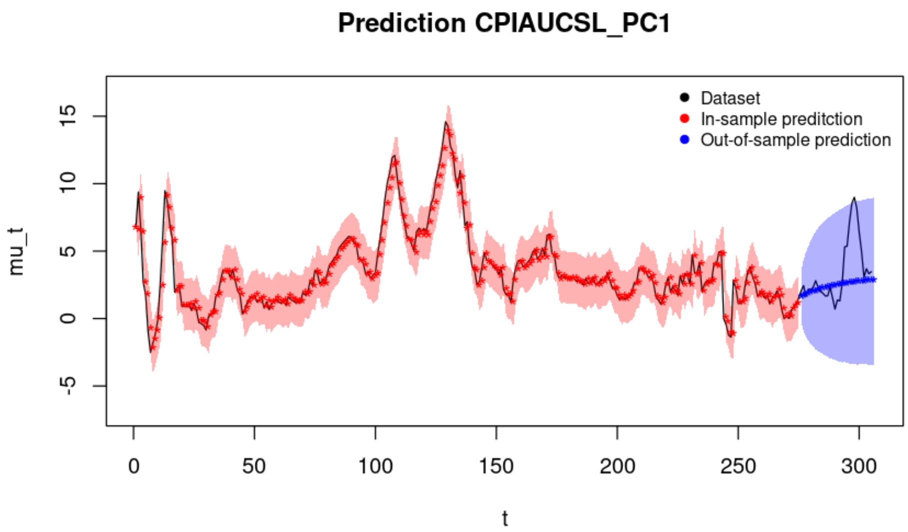


AR(1)
Trace plots



AR(1)
In-sample and Out-of-sample Predictions





POLITECNICO
MILANO 1863

Moving Average Model (MA)

Moving Average Model (MA)
General formulation

$$y_t = \mu_0 + \sum_{i=1}^{q} \theta_i \epsilon_{t-i} + \epsilon_t$$

- q: number of past error terms considered
- μ_0: mean of the series
- θ_i: model parameters
- ϵ_t: white noise
- $\epsilon_{t-1}, \ldots, \epsilon_{t-q}$: past error terms

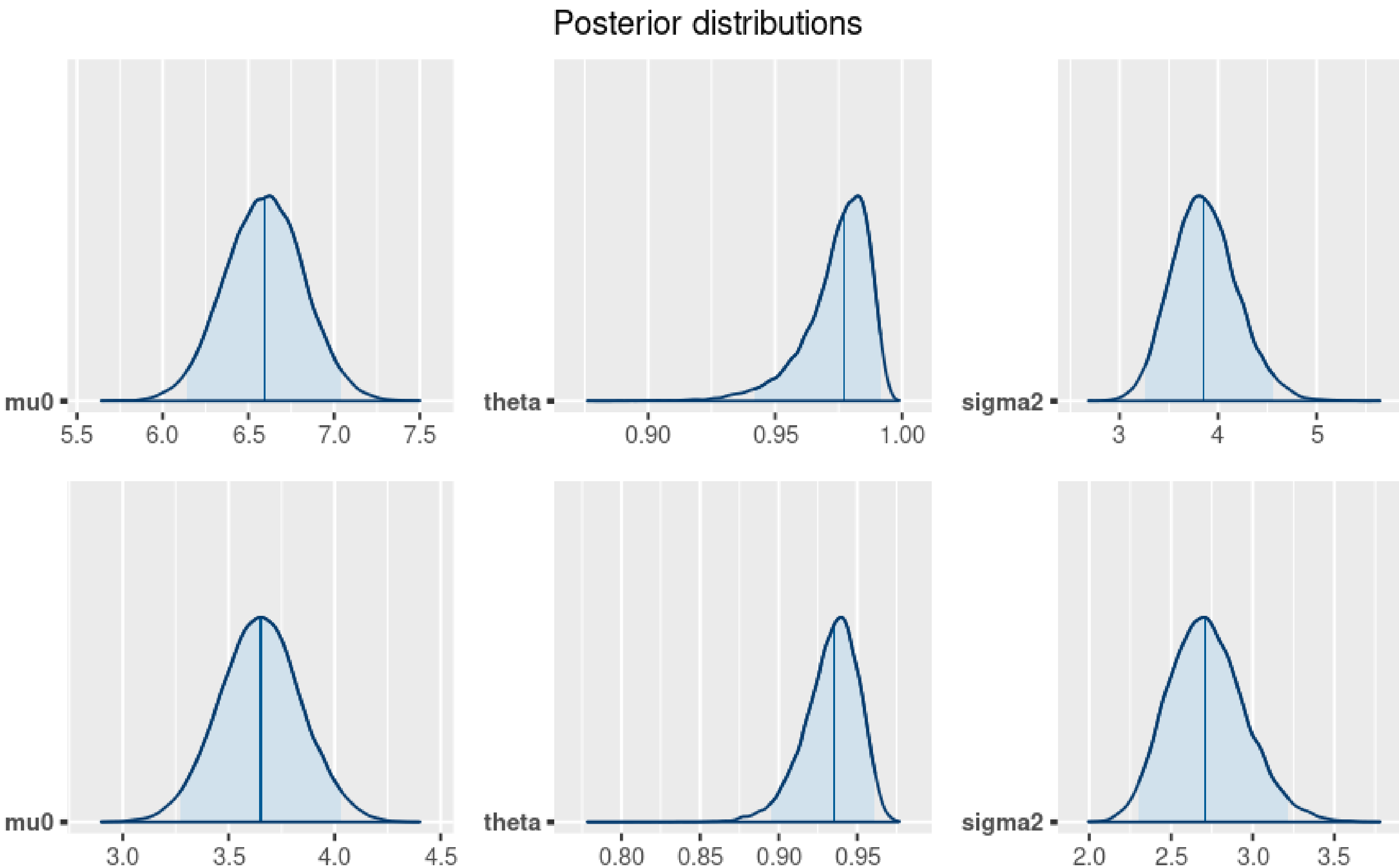
Moving Average Model (MA)
MA(1)

- Model: $$y_t = \mu_0 + \theta \epsilon_{t-1} + \epsilon_t$$
- Likelihood: $$y_t | \mu_0, \theta, \sigma^2, \epsilon_{t-1} \sim \mathcal{N}(\mu_0 + \theta \epsilon_{t-1}, \sigma^2)$$
- Priors:
$$\begin{aligned} \mu_0 &\sim \mathcal{N}(0.0, 10000) \\ \tau = 1/\sigma^2 &\sim \mathcal{G}(2, 0.1) \\ \theta &\sim \mathcal{U}(-1.0, 1.0) \end{aligned}$$

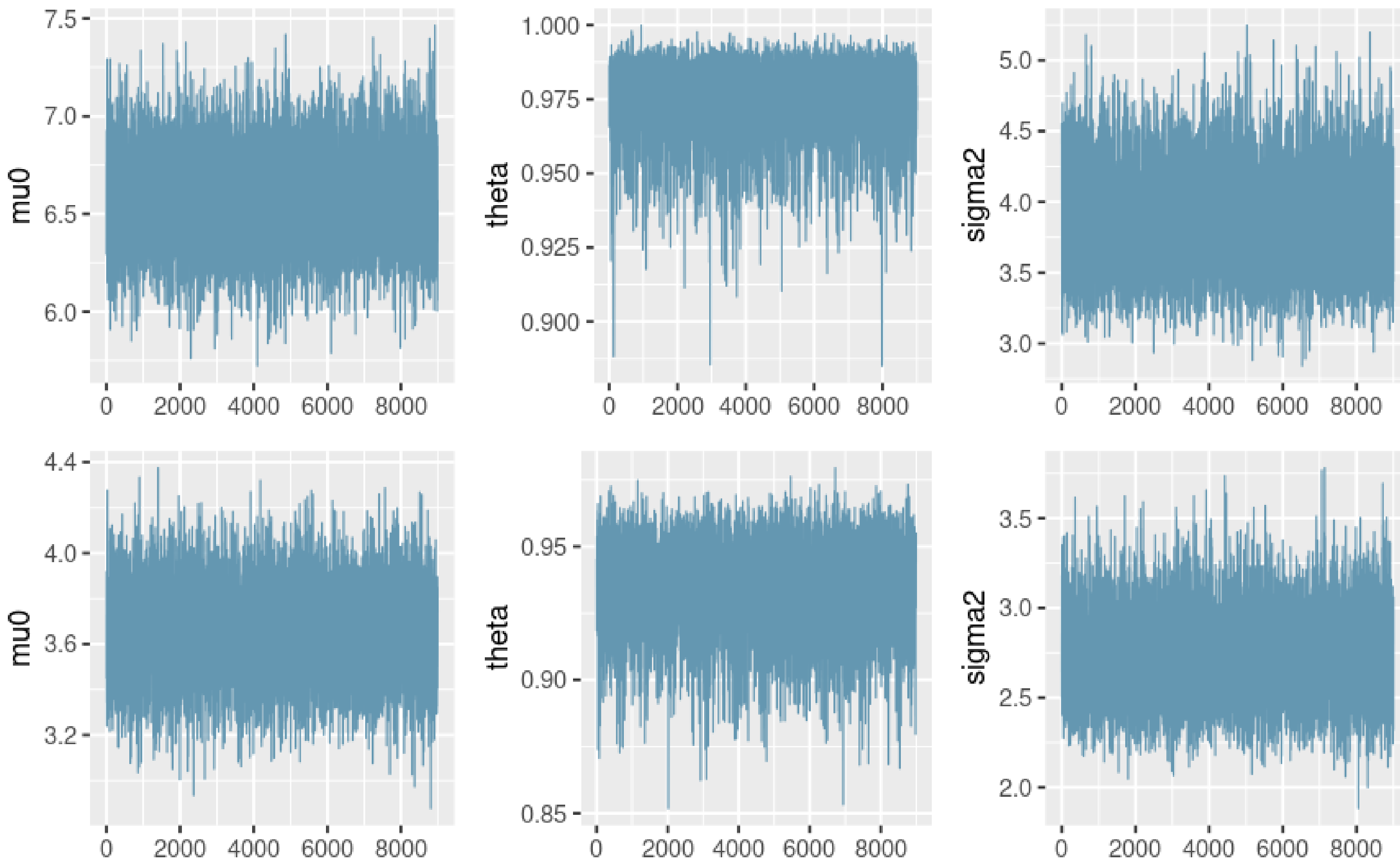
MA(1)
Jags

```
# Define model in JAGS
modelMA.string <-"model {
## Parameters: alpha, tau, mu0
  # Likelihood
  Yp[1]  <- mu[1]
  mu[1]  <- Y[1]
  eps[1] <- Y[1] - mu[1]
  for (i in 2:N) {
    Y[i]   ~  dnorm(mu[i], tau)
    mu[i]  <- mu0 + theta * eps[i-1]
    eps[i] <- Y[i] - mu[i]
    Yp[i]  ~  dnorm(mu[i], tau)
    LogLik[i] <- log(dnorm(Y[i], mu[i], tau))
  }

  # prediction out of sample
  ypOut[1]  ~  dnorm(mu0+theta*eps[N],tau)
  muOut[1]  <- mu0+theta*eps[N]
  epsOut[1] <- ypOut[1] - muOut[1]
  for(k in 2:Npred){
    ypOut[k]  ~  dnorm(mu0+theta*epsOut[k-1],tau)
    muOut[k]  <- mu0+theta*epsOut[k-1]
    epsOut[k] <- ypOut[k] - muOut[k]
  }

  sigma2<-1/tau
  #prior
  theta ~ dunif(-1.0, 1.0)
  tau   ~ dgamma(2, 0.1)
  mu0    ~ dnorm(0.0, 1.0E-4)
}"
```

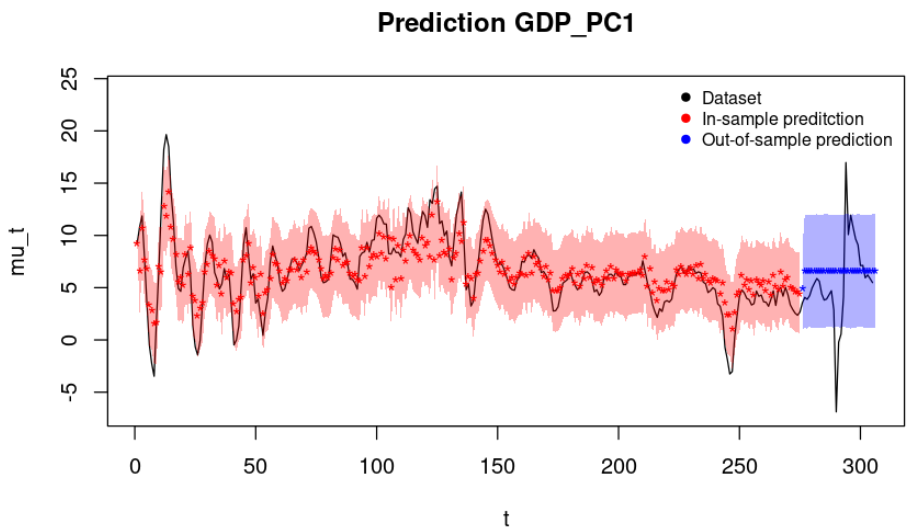
MA(1)
Posterior Distribution

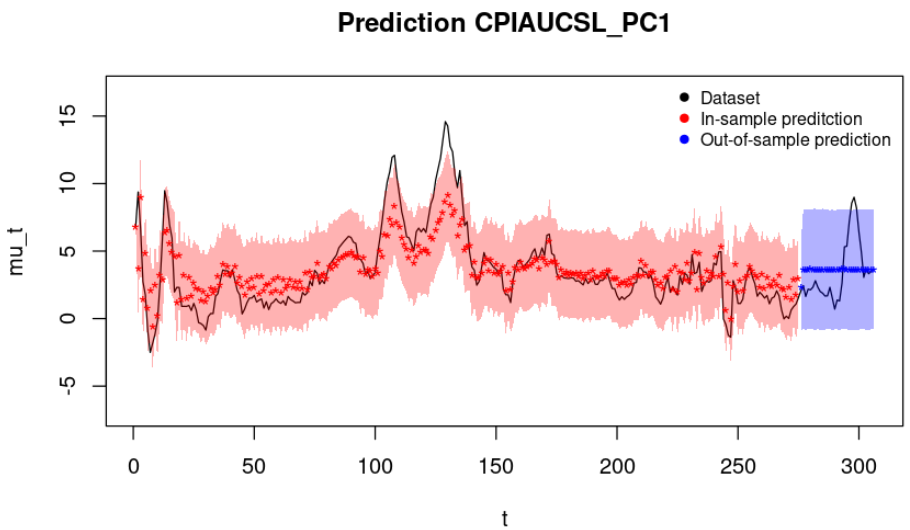


MA(1)
Trace plots



MA(1)
In-sample and Out-of-sample Predictions





Moving Average Model (MA)
MA(2)

- Model: $$y_t = \mu_0 + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \epsilon_t$$

- Likelihood: $$y_t | \mu_0, \theta_1, \theta_2, \sigma^2, \epsilon_{t-1}, \epsilon_{t-2} \sim \mathcal{N}(\mu_0 + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2}, \sigma^2)$$

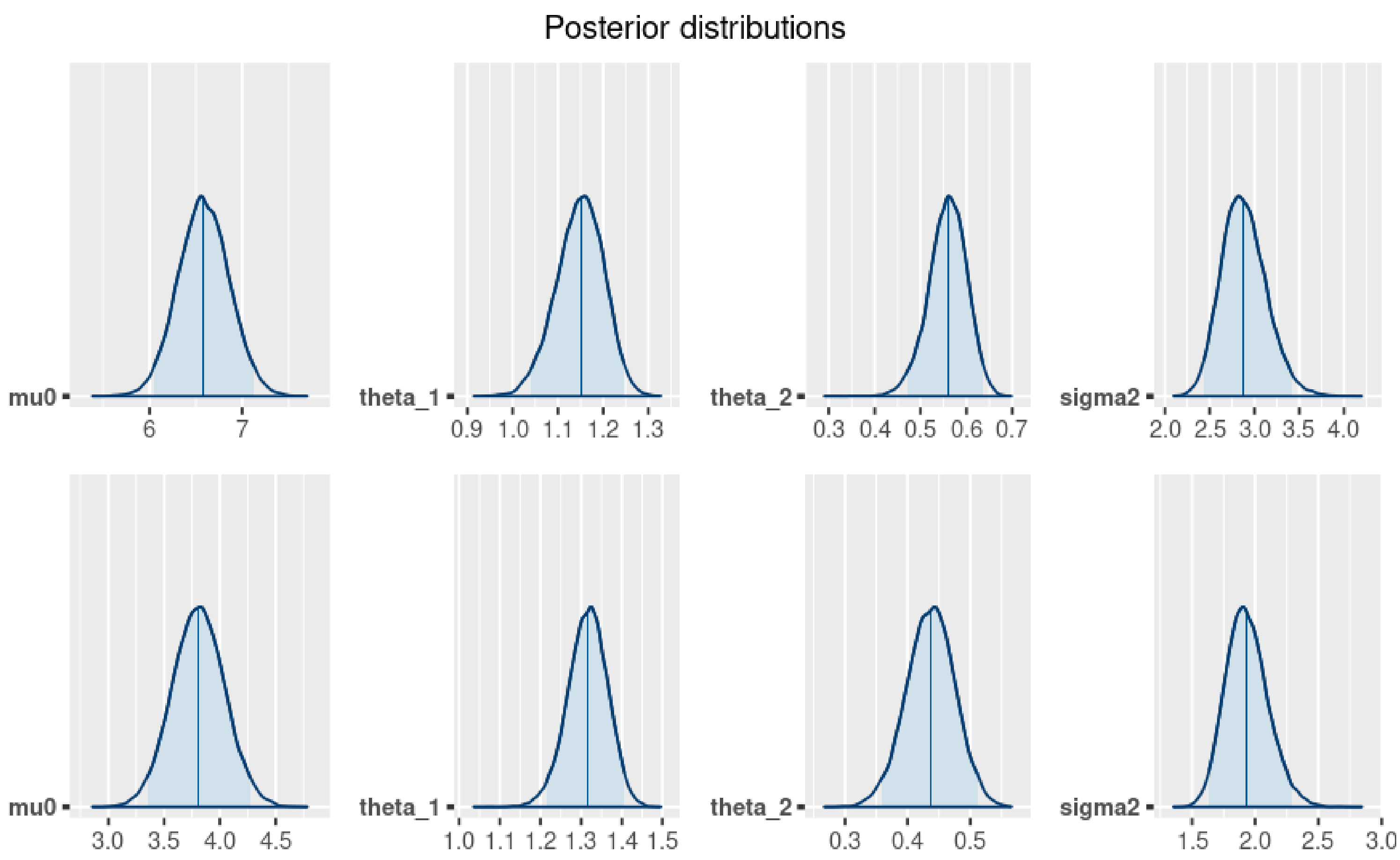
- Priors: $$\begin{aligned} \mu_0 &\sim \mathcal{N}(0.0, 10000) \\ \tau = 1/\sigma^2 &\sim \mathcal{G}(2, 0.1) \\ \theta_1 &\sim \mathcal{U}(-1.5, 1.5) \\ \theta_2 &\sim \mathcal{U}(-1.0, 1.0) \end{aligned}$$

MA(2)

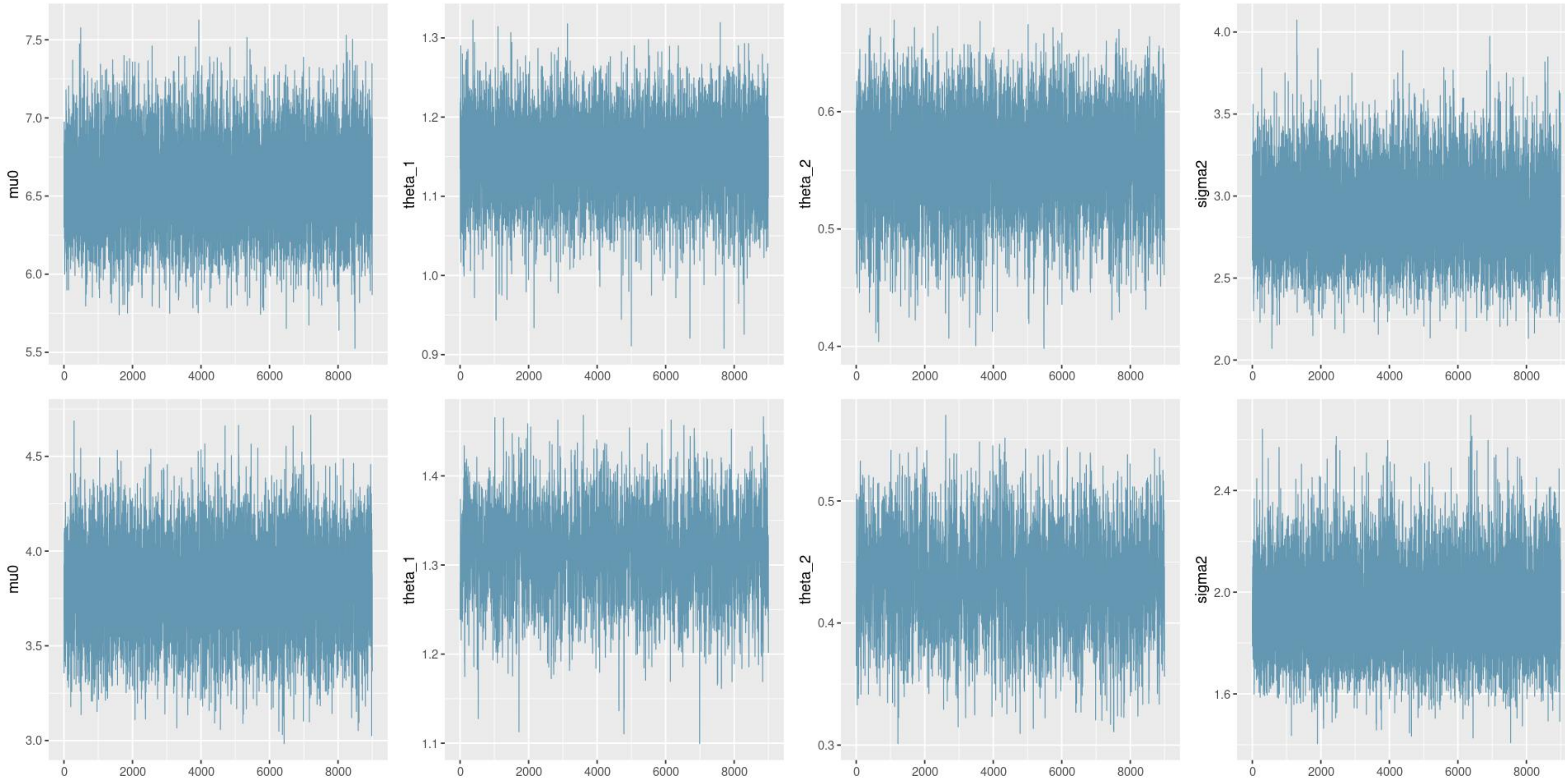
Jags

```
## Parameters: alpha, tau, mu0
  # Likelihood
  Yp[1] <- mu[1]
  mu[1] <- Y[1]
  eps[1] <- Y[1] - mu[1]
  Y[2] ~ dnorm(mu[2], tau)
  Yp[2] ~ dnorm(mu[2], tau)
  mu[2] <- mu0 + theta_1 * eps[1]
  eps[2] <- Y[2] - mu[2]
  for (i in 3:N) {
    Y[i] ~ dnorm(mu[i], tau)
    mu[i] <- mu0 + theta_1 * eps[i-1] + theta_2 * eps[i-2]
    eps[i] <- Y[i] - mu[i]
    Yp[i] ~ dnorm(mu[i], tau)
    LogLik[i] <- log(dnorm(Y[i], mu[i], tau))
  }

  # prediction out of sample
  ypOut[1] ~dnorm(muOut[1],tau)
  muOut[1] <- mu0+theta_1*eps[N]+theta_2*eps[N-1]
  epsOut[1] <- ypOut[1] - muOut[1]
  ypOut[2] ~dnorm(muOut[2],tau)
  muOut[2] <- mu0+theta_1*epsOut[1]+theta_2*eps[N]
  epsOut[2] <- ypOut[2] - muOut[2]
  for(k in 3:Npred){
    ypOut[k] ~ dnorm(muOut[k],tau)
    muOut[k] <- mu0+theta_1*epsOut[k-1]+theta_2*epsOut[k-2]
    epsOut[k] <- ypOut[k] - muOut[k]
  }

  sigma2<-1/tau
  #prior
  theta_1 ~ dunif(-1.5, 1.5)
  theta_2 ~ dunif(-1, 1)
  tau ~ dgamma(2, 0.1)
  mu0 ~ dnorm(0.0, 1.0E-4)
}"
```

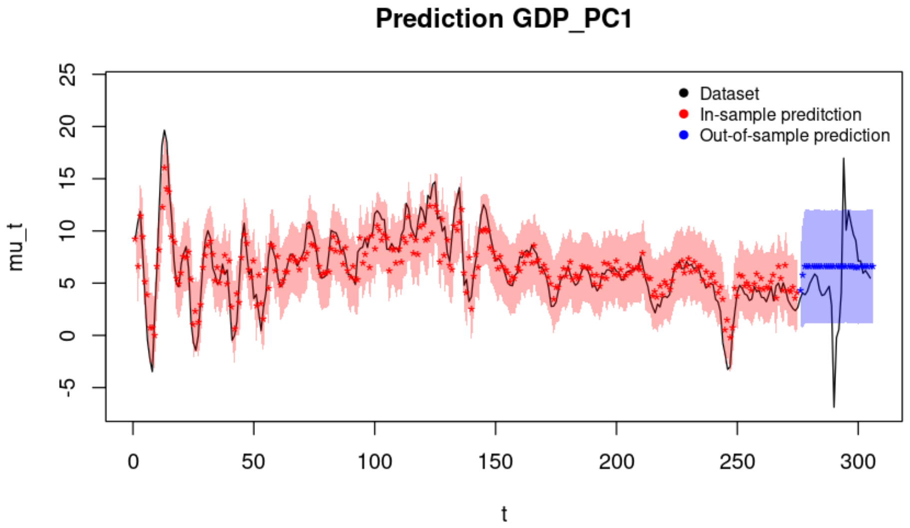
MA(2)
Posterior Distribution

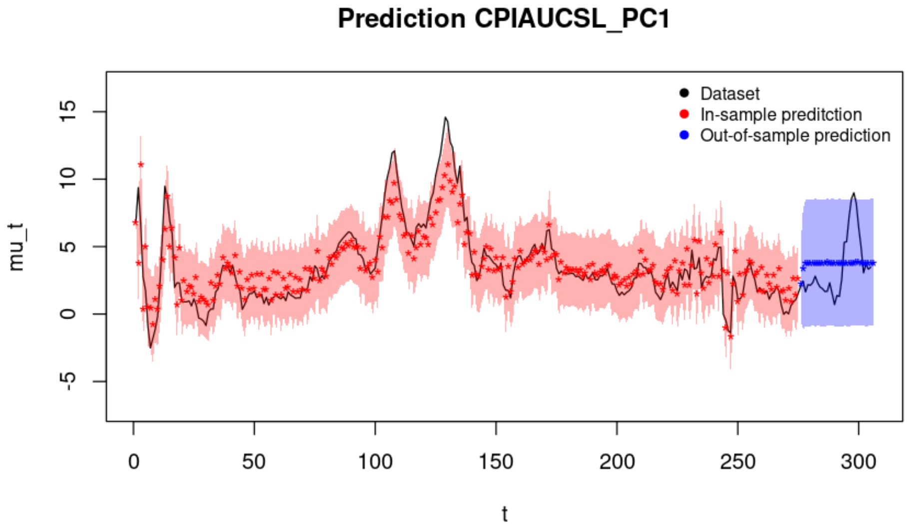


MA(2)
Trace plots



MA(2)
In-sample and Out-of-sample Predictions





Autoregressive Moving Average Model (ARMA)

Autoregressive Moving Average Model (ARMA)
General formulation

$$y_t = \mu_0 + \sum_{i=1}^{p} \alpha_i y_{t-i} + \sum_{j=1}^{q} \theta_j \epsilon_{t-j} + \epsilon_t$$

- p: number of past values considered
- q: number of past error terms
- μ_0: constant value
- α_i: model parameters
- θ_i: model parameters
- ϵ_t: white noise
- $y_{t-1}, \dots, y_{t-p}$: past values
- $\epsilon_{t-1}, \dots, \epsilon_{t-q}$: past error terms

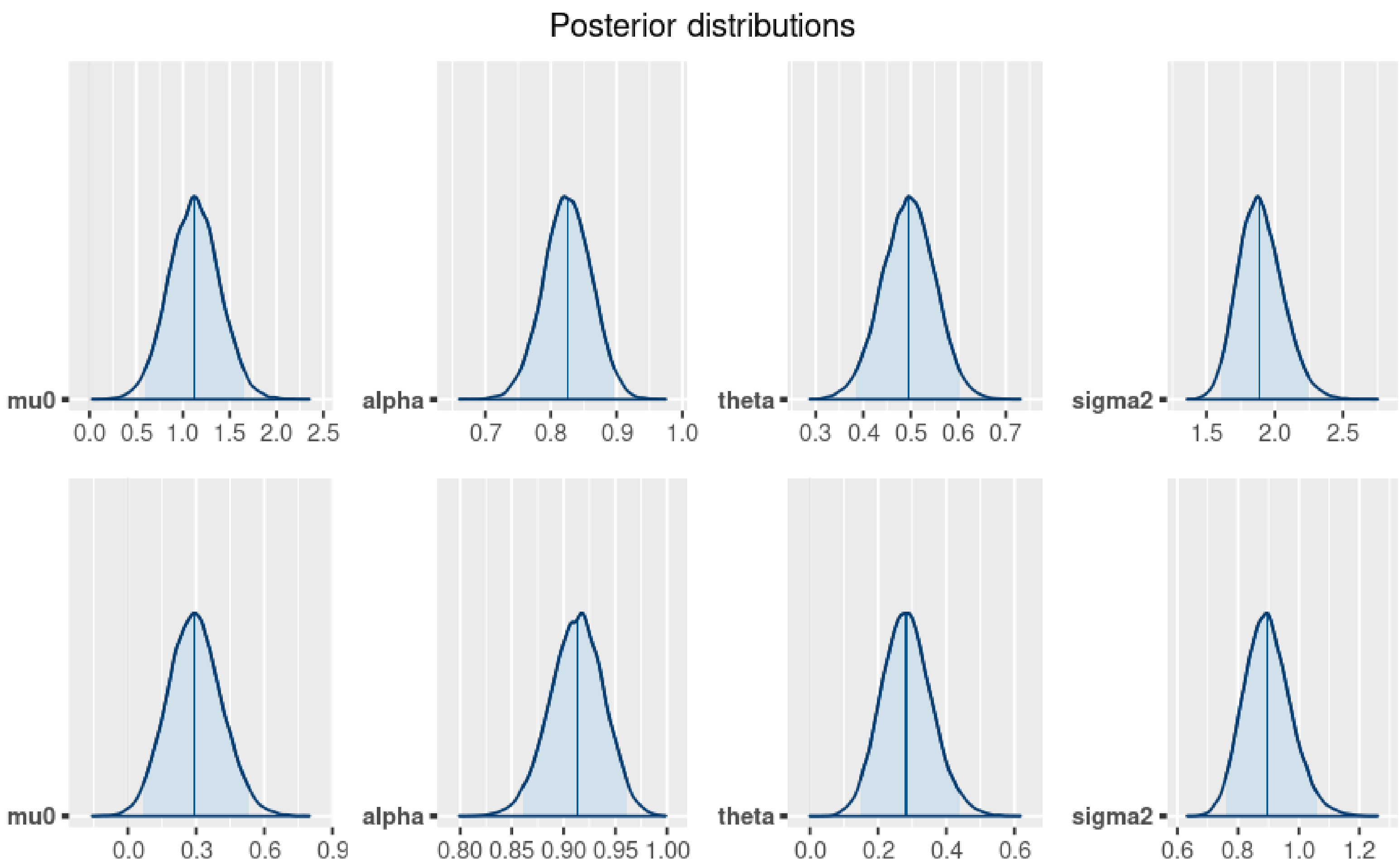
Autoregressive Moving Average Model (ARMA)
ARMA(1, 1)

- Model: $$y_t = \mu_0 + \alpha y_{t-1} + \theta \epsilon_{t-1} + \epsilon_t$$
- Likelihood: $$y_t | \mu_0, \alpha, \theta, \sigma^2, y_{t-1}, \epsilon_{t-1} \sim \mathcal{N}(\mu_0 + \alpha y_{t-1} + \theta \epsilon_{t-1}, \sigma^2)$$
- Priors: $$\begin{aligned} \mu_0 &\sim \mathcal{N}(0.0, 10000) \\ \tau = 1/\sigma^2 &\sim \mathcal{G}(2, 0.1) \\ \alpha &\sim \mathcal{U}(-1.0, 1.0) \\ \theta &\sim \mathcal{U}(-1.0, 1.0) \end{aligned}$$

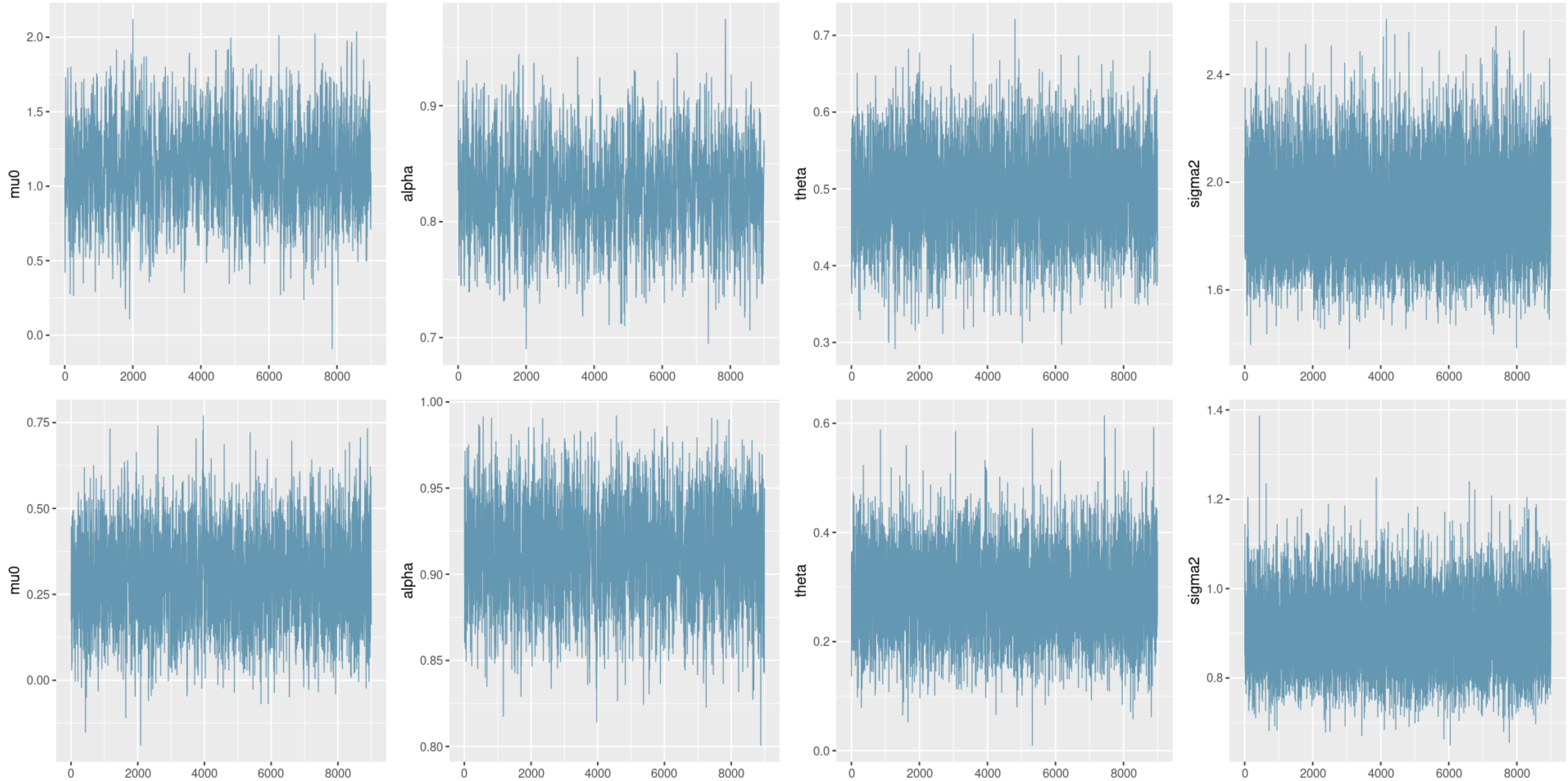
ARMA(1,1)
Jags

```
# Define model in JAGS
modelARMA.string <-"model {
  ## Parameters: alpha, tau, mu0
  # Likelihood
  Yp[1]  <- mu[1]
  mu[1]  <- Y[1]
  eps[1] <- Y[1] - mu[1]
  for (i in 2:N) {
    Y[i]   ~ dnorm(mu[i], tau)
    Yp[i]  ~ dnorm(mu[i], tau)      # Prediction in sample
    mu[i]  <- mu0 + alpha * Y[i-1] + theta * eps[i-1]
    eps[i] <- Y[i] - mu[i]
    LogLik[i] <- log(dnorm(Y[i], mu[i], tau))
  }

  # Prediction out of sample
  ypOut[1]  ~ dnorm(muOut[1], tau)
  muOut[1]  <- mu0 + alpha * Y[N] + theta * eps[N]
  epsOut[1] <- ypOut[1] - muOut[1]
  for (k in 2:Npred) {
    ypOut[k] ~ dnorm(muOut[k], tau)
    muOut[k] <- mu0 + alpha * ypOut[k-1] + theta * epsOut[k-1]
    epsOut[k] <- ypOut[k] - muOut[k]
  }

  sigma2 <- 1/tau

  # Prior
  alpha ~ dunif(-1, 1)
  theta ~ dunif(-1, 1)
  tau ~ dgamma(2, 0.1)
  mu0    ~ dnorm(0.0, 1.0E-4)
}"
```

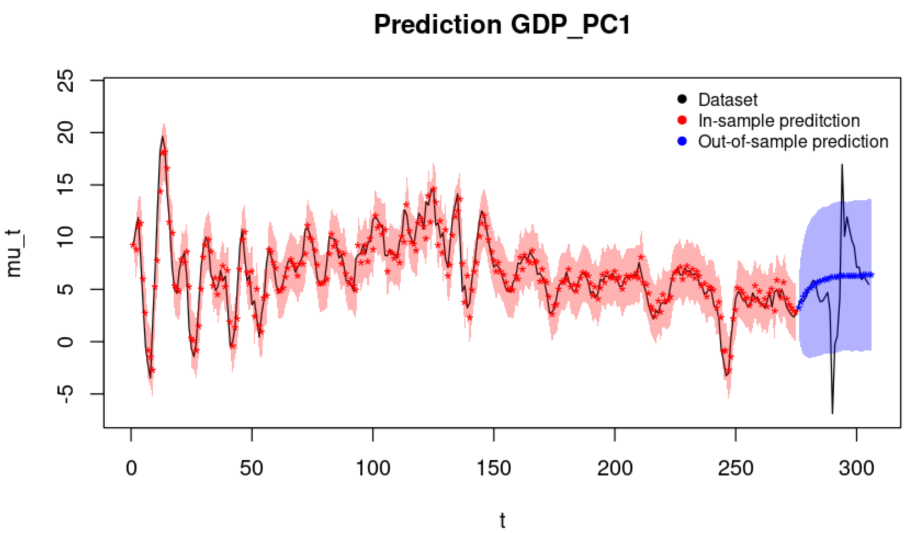
ARMA(1, 1)
Posterior Distribution

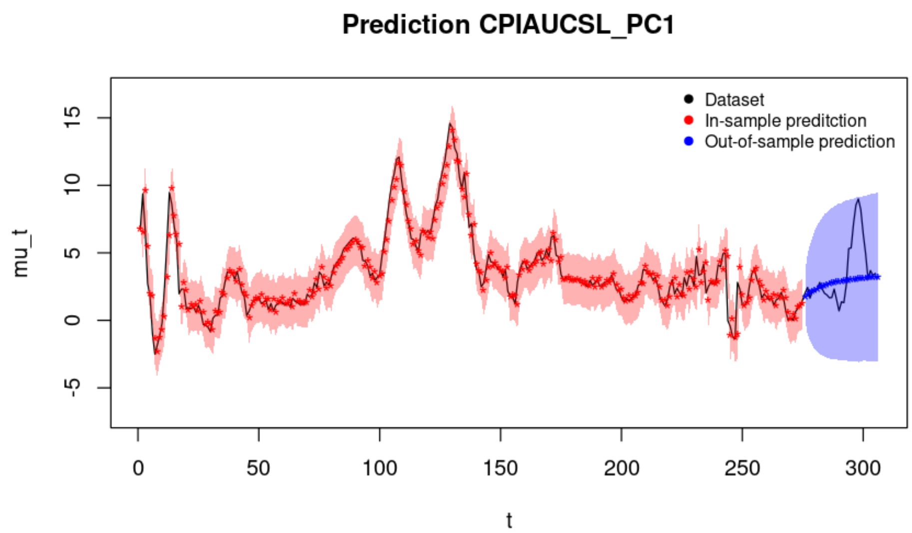


ARMA(1, 1)
Trace plots



ARMA(1, 1)
In-sample and Out-of-sample Predictions





POLITECNICO
MILANO 1863

Generalized Autoregressive Conditional Heteroskedasticity Model (GARCH)

Generalized Autoregressive Conditional Heteroskedasticity Model (GARCH)
General formulation

$$\sigma_t^2 = a_0 + \sum_{i=1}^{p} a_i \epsilon_{t-i}^2 + \sum_{j=1}^{q} a_{j+p} \sigma_{t-j}^2$$

- p: number of past squared error terms
- q: number of past variances considered
- α_i: model parameters
- ϵ_t: white noise
- $\epsilon_{t-1}^2, \dots, \epsilon_{t-p}^2$: past squared error terms
- $\sigma_{t-1}^2, \dots, \sigma_{t-q}^2$: past variances

Generalized Autoregressive Conditional Heteroskedasticity Model (GARCH)

AR(1) + GARCH(1,1)

- Model: $y_t = \mu_0 + \alpha y_{t-1} + \epsilon_t$ $\qquad \sigma_t^2 = a_0 + a_1 \epsilon_{t-1}^2 + a_2 \sigma_{t-1}^2$

- Likelihood: $y_t | \mu_0, \alpha, y_{t-1}, a_0, a_1, a_2, \sigma_{t-1}^2 \sim \mathcal{N}(\mu_0 + \alpha y_{t-1}, a_0 + a_1 \epsilon_{t-1}^2 + a_2 \sigma_{t-1}^2)$

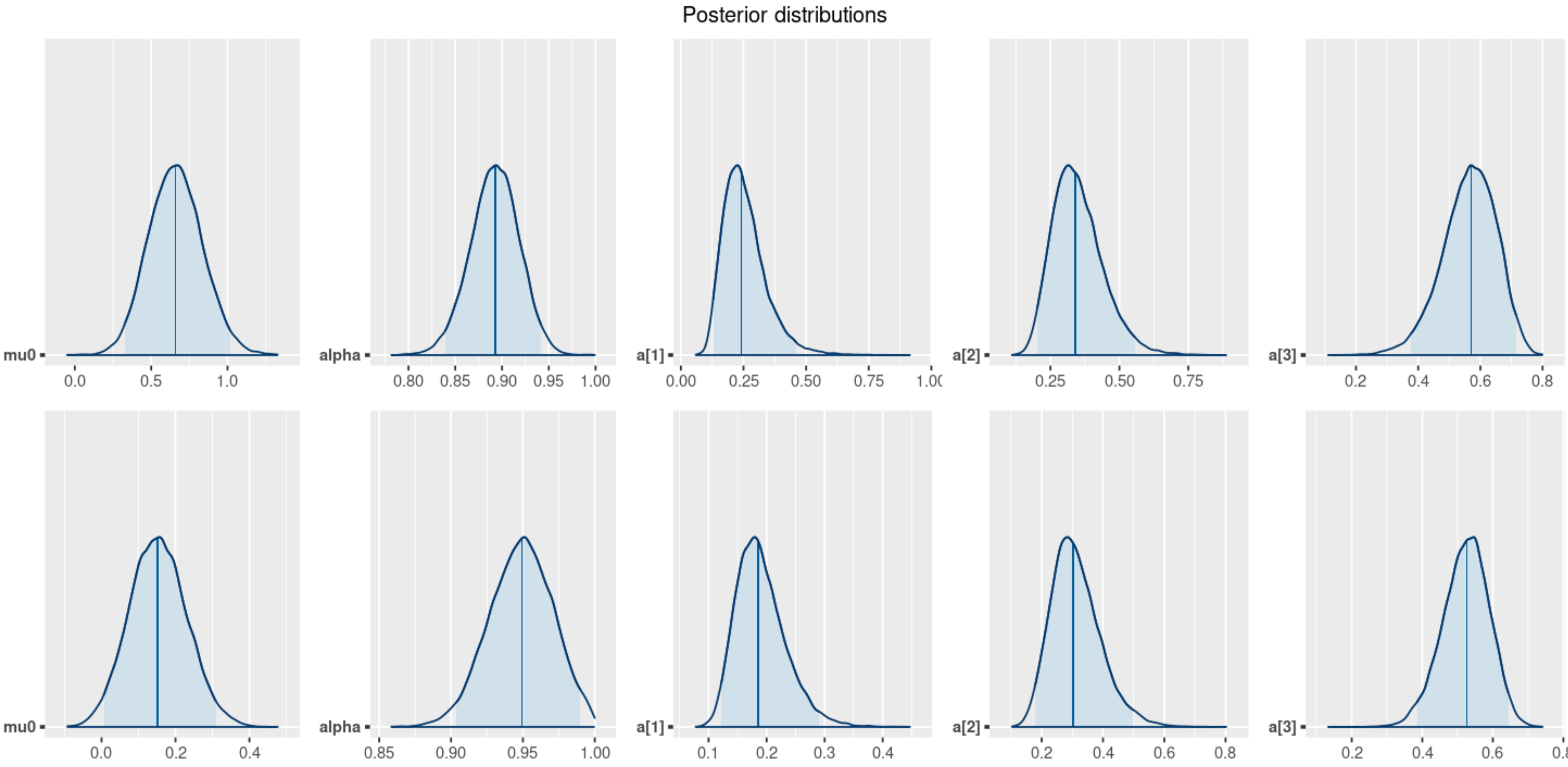
- Priors:

$$
\begin{aligned}
\mu_0 &\sim \mathcal{N}(0.0, 10000) \\
\alpha &\sim \mathcal{U}(-1.0, 1.0) \\
a_0 &\sim \mathcal{G}(0.01, 0.01) \\
a_1 &\sim \mathcal{G}(0.01, 0.01) \\
a_2 &\sim \mathcal{G}(0.01, 0.01)
\end{aligned}
$$

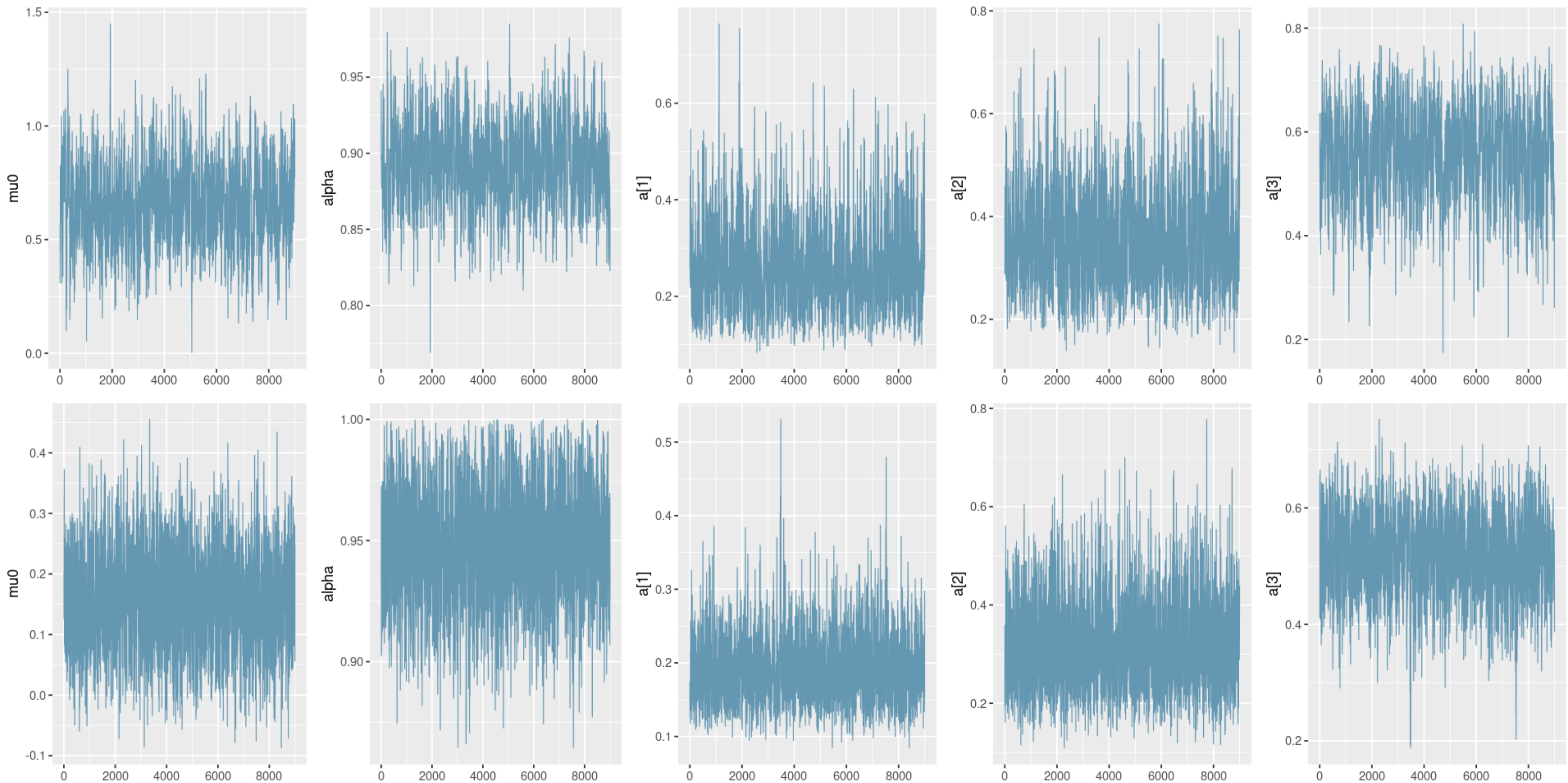
AR(1) + GARCH(1, 1)
Jags

```
# Define model in JAGS
modelGARCH.string <-"model {
  # Likelihood
  Yp[1]     <- Y[1]
  mu[1]     <- Y[1]
  tau[1]    <- 1 / sigma2[1]
  sigma2[1] <- a[1]
  eps2[1]   <- (Y[1] - mu[1]) * (Y[1] - mu[1])
  for(i in 2:N) {
    Y[i]      ~ dnorm(mu[i], tau[i])
    Yp[i]     ~ dnorm(mu[i], tau[i])      # Prediction in sample
    mu[i]     <- mu0 + alpha * Y[i-1]
    tau[i]    <- 1 / sigma2[i]
    sigma2[i] <- a[1] + a[2] * eps2[i-1] + a[3] * sigma2[i-1]
    eps2[i]   <- (Y[i] - mu[i]) * (Y[i] - mu[i])
    LogLik[i] <- log(dnorm(Y[i], mu[i], tau[i]))
  }

  # Prediction out of sample
  ypOut[1]     ~  dnorm(muOut[1], tauOut[1])
  muOut[1]     <- mu0 + alpha * Y[N]
  tauOut[1]    <- 1 / sigma2Out[1]
  sigma2Out[1] <- a[1] + a[2] * eps2[N] + a[3] * sigma2[N]
  eps2Out[1]   <- (ypOut[1] - muOut[1]) * (ypOut[1] - muOut[1])
  for (k in 2:Npred) {
    ypOut[k]     ~  dnorm(muOut[k], tauOut[k])
    muOut[k]     <- mu0 + alpha * ypOut[k-1]
    tauOut[k]    <- 1 / sigma2Out[k]
    sigma2Out[k] <- a[1] + a[2] * eps2Out[k-1] + a[3] * sigma2Out[k-1]
    eps2Out[k]   <- (ypOut[k] - muOut[k]) * (ypOut[k] - muOut[k])
  }

  # Prior
  for(j in 1:3) {
    a[j] ~ dgamma(0.01, 0.01)
  }
  alpha ~ dunif(-1, 1)
  mu0   ~ dnorm(0.0, 1.0E-4)
}"
```

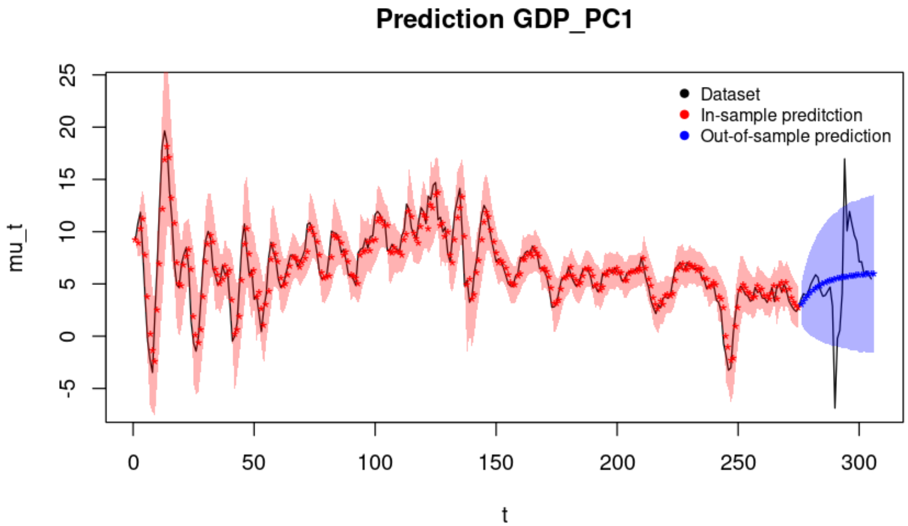
AR(1) + GARCH(1, 1)
Posterior Distribution

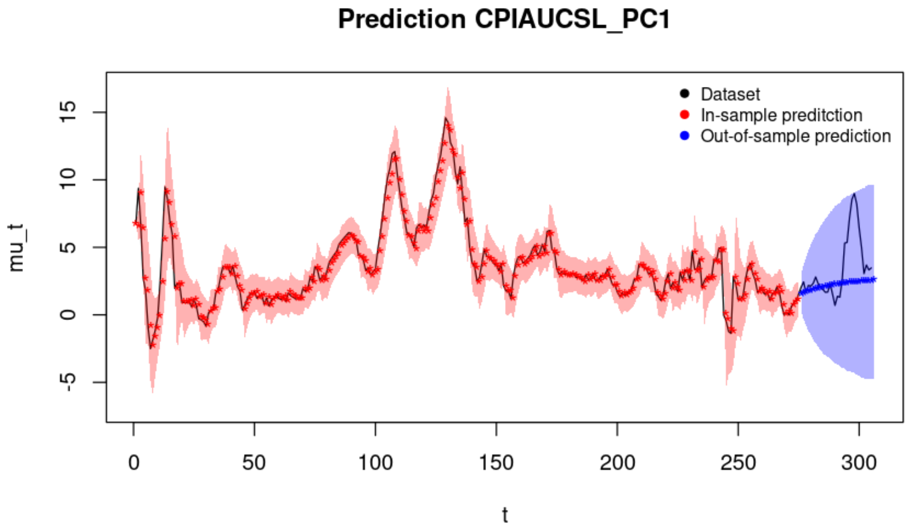


AR(1) + GARCH(1, 1)
Trace plots



AR(1) + GARCH(1, 1)
In-sample and Out-of-sample Predictions





POLITECNICO
MILANO 1863

Vector Autoregressive Model (VAR)

Vector Autoregressive Model (VAR)
General formulation

$$\mathbf{y}_t = \mu_0 + A_1\mathbf{y}_{t-1} + A_2\mathbf{y}_{t-2} + \cdots + A_p\mathbf{y}_{t-p} + \epsilon_t$$

- p: number of past vectors of variables considered
- μ_0: is a K × 1 vector of constants.
- $A_1, \dots, A_p$: are K × K coefficient matrices.
- ϵ_t: is a K × 1 vector of error terms at time t.
- y_t : is a K × 1 vector of variables at time t.

Vector Autoregressive Model (VAR)
VAR(1)

- Model(K=2): $$\begin{pmatrix} y_{1,t} \\ y_{2,t} \end{pmatrix} = \begin{pmatrix} \mu_{0,1} \\ \mu_{0,2} \end{pmatrix} + \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} y_{1,t-1} \\ y_{2,t-1} \end{pmatrix} + \begin{pmatrix} \epsilon_{1,t} \\ \epsilon_{2,t} \end{pmatrix}$$

- Likelihood: $$\mathbf{y}_t | \boldsymbol{\mu}_0, A, \Sigma, \mathbf{y}_{t-1} \sim \mathcal{N}_2(\boldsymbol{\mu}_0 + A\mathbf{y}_{t-1}, \Sigma)$$

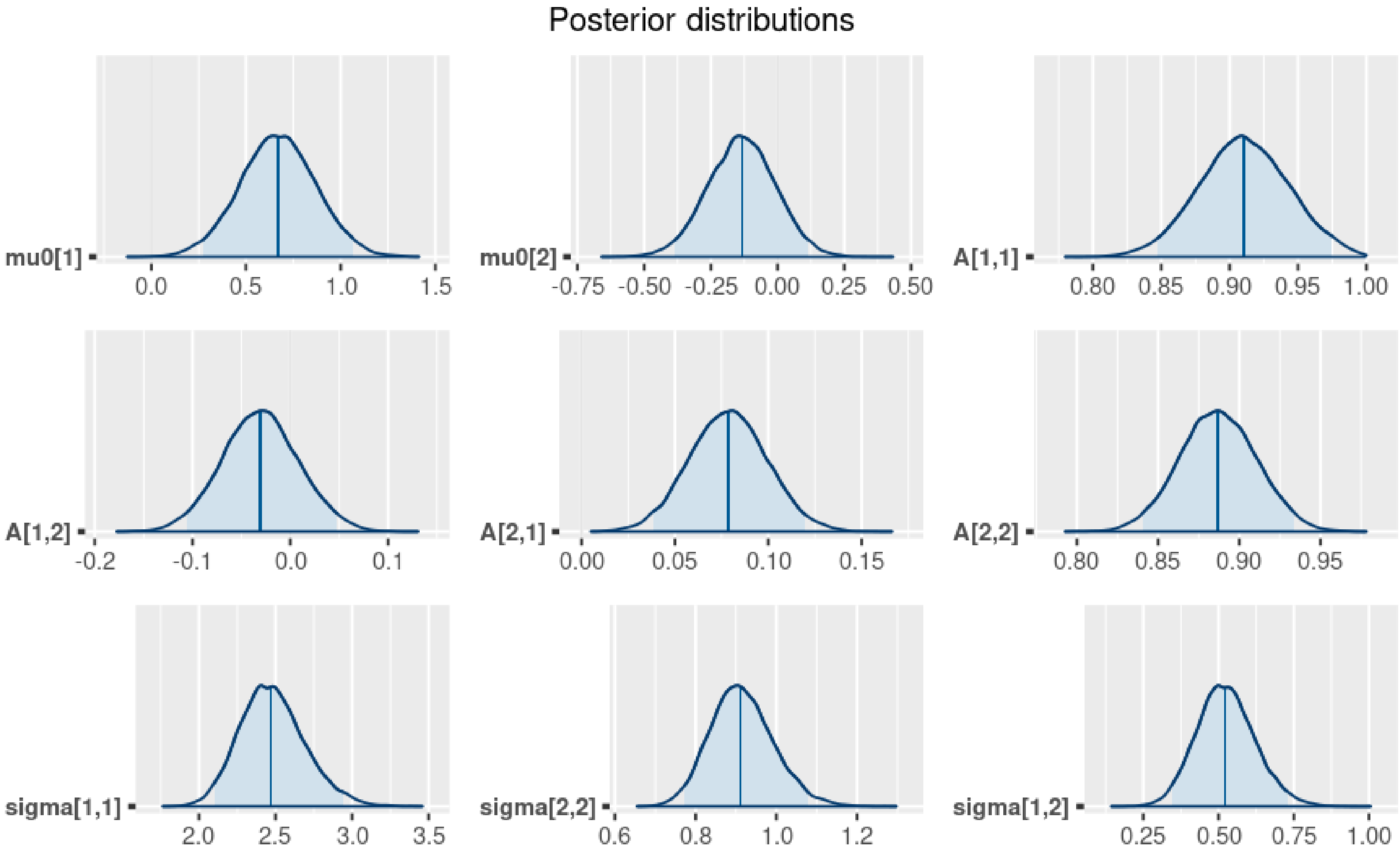
- Priors: $$\begin{aligned} \mu_{0,i} &\sim \mathcal{N}(0.0, 10000) \qquad i = 1,2 \\ a_{ij} &\sim \mathcal{U}(-1,1) \qquad i,j = 1,2 \\ \Omega = \Sigma^{-1} &\sim Wishart(R, 3) \end{aligned}$$ $$R = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$$

VAR(1)

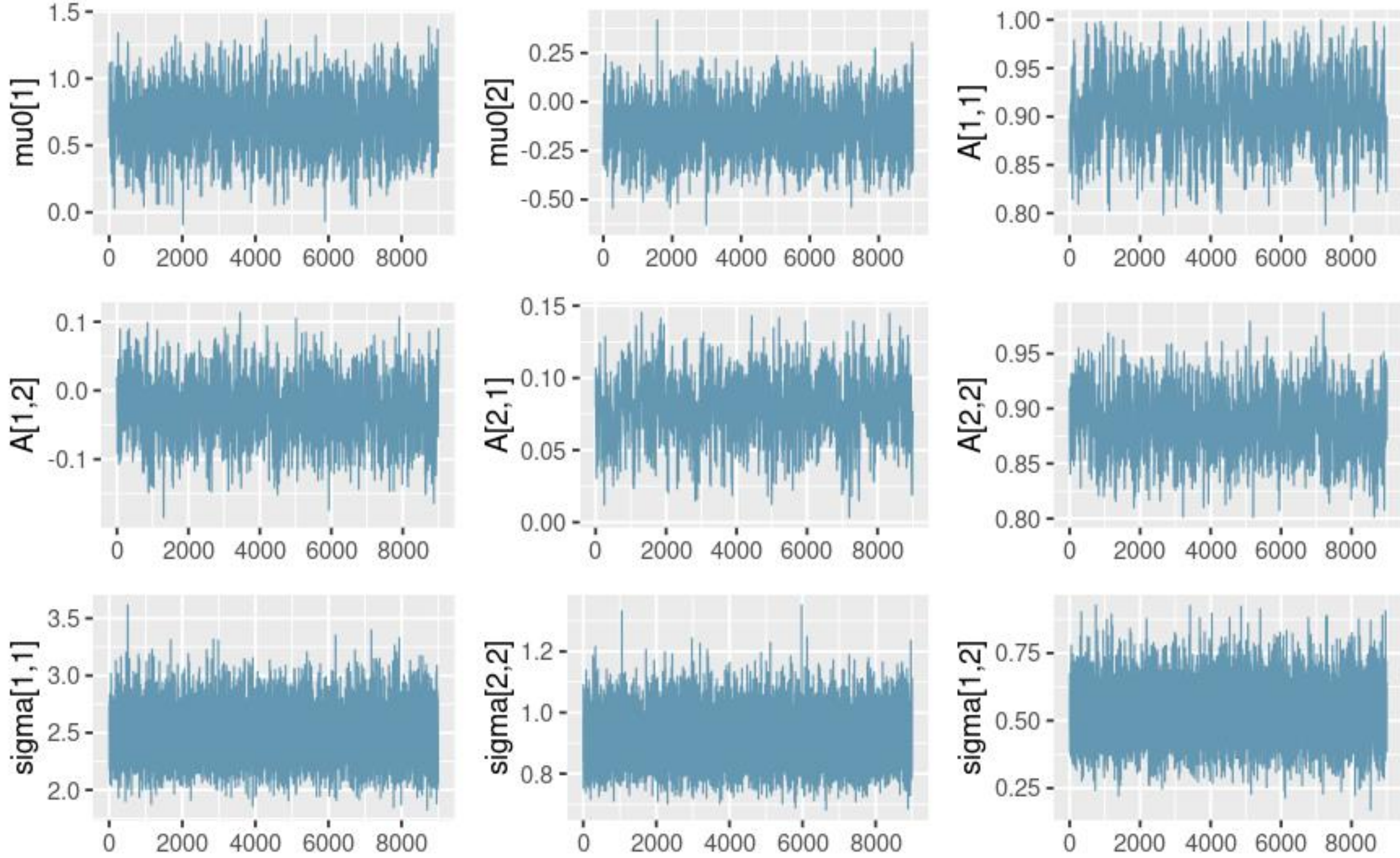
Jags

```
# Define model in JAGS
modelVAR.string <-"model {
  # Likelihood
  Yp[1:2, 1] <- Y[1:2, 1]
  mu[1:2, 1] <- Y[1:2, 1]
  for (i in 2:N) {
    Y[1:2, i]  ~ dmnorm(mu[1:2, i], omega[1:2, 1:2])
    Yp[1:2, i] ~ dmnorm(mu[1:2, i], omega[1:2, 1:2])   # Prediction in sample
    mu[1,i]    <- mu0[1] + A[1,1] * Y[1,i-1] + A[1,2] * Y[2,i-1]
    mu[2,i]    <- mu0[2] + A[2,1] * Y[1,i-1] + A[2,2] * Y[2,i-1]
    LogLik[i]  <- logdensity.mnorm(Y[1:2, i], mu[1:2, i], omega[1:2, 1:2])
  }

  # Prediction out of sample
  ypOut[1:2, 1]  ~ dmnorm(muOut[1:2, 1], omega[1:2, 1:2])
  muOut[1,1]     <- mu0[1] + A[1,1] * Y[1,N] + A[1,2] * Y[2,N]
  muOut[2,1]     <- mu0[2] + A[2,1] * Y[1,N] + A[2,2] * Y[2,N]
  for (k in 2:Npred) {
    ypOut[1:2, k] ~ dmnorm(muOut[1:2, k], omega[1:2, 1:2])
    muOut[1,k]    <- mu0[1] + A[1,1] * ypOut[1,k-1] + A[1,2] * ypOut[2,k-1]
    muOut[2,k]    <- mu0[2] + A[2,1] * ypOut[1,k-1] + A[2,2] * ypOut[2,k-1]
  }

  sigma <- inverse(omega)

  # Prior
  for(j in 1:2) {
    for(h in 1:2) {
      A[j, h] ~ dunif(-1, 1)
    }
    mu0[j]  ~ dnorm(0.0, 1.0E-4)
  }

  omega ~ dwish(R,k)

  k <- 3
  R[1,1] <- 1.0
  R[1,2] <- 0.5
  R[2,1] = R[1,2]
  R[2,2] <- 1.0

}"
```

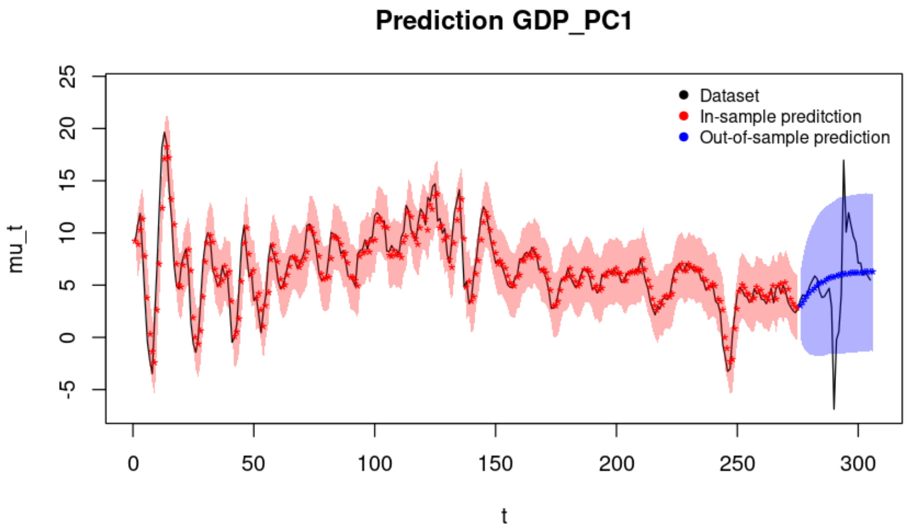
VAR(1)
Posterior Distribution

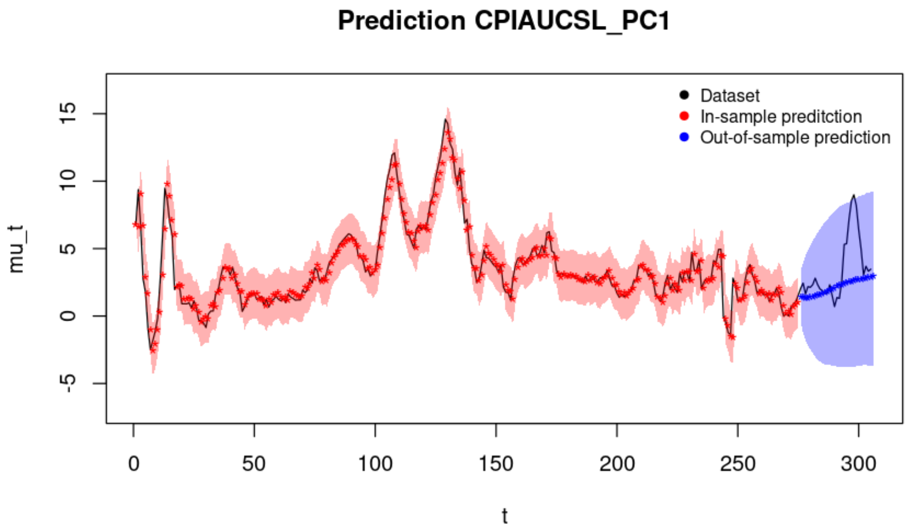


VAR(1)
Trace plots



VAR(1)
In-sample and Out-of-sample Predictions





Model Comparison

Model Comparison

	DIC		WAIC	
Model	GDP	CPIAUCSL	GDP	CPIAUCSL
AR(1)	1029.708	769.6079	1031.7	773.0
MA(1)	1154.302	1057.921	1154.5	1058.6
MA(2)	1074.749	965.4048	1068.2	949.8
ARMA(1,1)	960.1701	754.3378	962.3	759.6
AR(1) + GARCH(1,1)	932.1271	714.7022	934.5	727.5
VAR(1)	1755.134		1762.2	

POLITECNICO
MILANO 1863

Thanks for your attention