

shrinkage_erro_gamma

Aplicando em modelo não agregado

```
set.seed(123)

# definindo parâmetros
n <- 16
beta <- 10; lambda <- 5 # parâmetros do erro Gamma
alpha <- 0.8

# gerando amostra
f <- f_test(n)$bumps
e <- rgamma(n, shape=beta, rate=lambda)
y <- f + e

# DWT
d <- wd(y, filter.number=5, family='DaubExPhase')
theta_true <- wd(f, filter.number=5, family='DaubExPhase') |> (\"(x) c(accessC(x, lev=0), x$D)\")()

# RAM
theta_1 <- gera_ponto(n, y) # chute inicial
post_gamma(theta_1, d, beta, lambda, tau=5)

## [1] 8.082226e-167

sample <- ram(theta_1, S_1=NULL, y, tau=5, alpha, beta, lambda, n_ite=10000)

## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## Fora do suporte.
## 10000
```

```
kable(data.frame('theta_i'=1:n, 'theta_true'=theta_true,
                 'RAM'=colMeans(sample$theta)))
```

theta_i	theta_true	RAM
1	8.3245060	8.0935815
2	-0.0475269	-0.1164587
3	-4.5040724	-5.7485277
4	-20.3329499	-19.5250787
5	9.0420742	8.6498816
6	-1.6569478	-0.5329409
7	0.2314904	0.2742032
8	0.8669488	0.9131615
9	0.2164796	-0.1936698
10	1.0862039	0.4494280
11	-9.8373110	-10.7859312
12	3.2973765	3.9028046
13	-1.8987408	-1.3939282
14	3.9815792	4.0389498
15	0.8085880	1.0064239
16	-9.3267555	-9.1999988

```
# set.seed(282829)
#
# # definindo parâmetros
# n <- 16
# beta <- 10; lambda <- 5 # parâmetros do erro Gamma
#
# fun_comp <- matrix(c(f_test(n)$bumps, f_test(n)$doppler), 2, byrow=T)
#
# #plotando funções
# par(mfrow=c(1,2))
# plot(fun_comp[1,], type='l', ylab='', main='Bumps')
# plot(fun_comp[2,], type='l', ylab='', main='Doppler')
# par(mfrow=c(1,1))
#
# # gerando amostra
# sample <- sample_gen(fun_comp, n=n, beta=beta, lambda=lambda, erro='gamma')
#
# # plot da amostra
# plot(x=(1:n)/n, y=NULL, type='n', xlim=c(0,1), ylim=c(-10, 30))
```

```
# for (i in 1:n) lines(x=(1:n)/n, sample$fun[i,], type='b', col=i)
#
# # aplicando RAM
# gera_ponto()
```