MultiGlarmaVarSel package

Marina Gomtsyan

Introduction

The package MultiGlarmaVarSel provides functions for performing variable selection approach in sparse multivariate GLARMA models, which are pervasive for modeling multivariate discrete-valued time series. The method consists in iteratively combining the estimation of the autoregressive moving average (ARMA) coefficients of GLARMA models with regularized methods designed to perform variable selection in regression coefficients of Generalized Linear Models (GLM). For further details on the methodology we refer the reader to [1].

We describe the multivariate GLARMA model. Given the past history $\mathcal{F}_{i,j,t-1} = \sigma(Y_{i,j,s}, s \leq t-1)$, we assume that

$$Y_{i,j,t}|\mathcal{F}_{i,j,t-1} \sim \mathcal{P}(\mu^\star_{i,j,t}), \tag{1}$$

where $\mathcal{P}(\mu)$ denotes the Poisson distribution with mean μ, $1 \leq i \leq I$, $1 \leq j \leq n_i$ and $1 \leq t \leq T$. For instance, $Y_{i,j,t}$ can be seen as a random variable modeling RNA-Seq data of the jth replication of gene t obtained in condition i. In (1)

$$\mu^\star_{i,j,t} = \exp(W^\star_{i,j,t}) \quad \text{with} \quad W^\star_{i,j,t} = \eta^\star_{i,t} + Z^\star_{i,j,t}, \tag{2}$$

where

$$Z^\star_{i,j,t} = \sum_{k=1}^{q} \gamma^\star_k E^\star_{i,j,t}, \qquad \text{with } 1 \leq q \leq \infty, \tag{3}$$

and $\eta^\star_{i,t}$, the non random part of $W^\star_{i,j,t}$, does not depend on j.

Let us denote $\boldsymbol{\eta}^\star = (\eta^\star_{1,1}, \ldots, \eta^\star_{I,1}, \eta^\star_{I,2}, \ldots, \eta^\star_{I,T})'$ the vector of coefficients corresponding to the effect of a qualitative variable on the observations. For instance, in the case of RNA-Seq data, $\eta^\star_{i,t}$ can be seen as the effect of condition i on gene t. Assume moreover that $\boldsymbol{\gamma}^\star = (\gamma^\star_1, \ldots, \gamma^\star_q)'$ is such that $\sum_{k\geq 1} |\gamma^\star_k| < \infty$, where u' denotes the transpose of u. Additionally,

$$E^\star_{i,j,t} = \frac{Y_{i,j,t} - \mu^\star_{i,j,t}}{\mu^\star_{i,j,t}} = Y_{i,j,t} \exp\big(-W^\star_{i,j,t}\big) - 1. \tag{4}$$

with $E^\star_{i,j,t} = 0$ for all $t \leq 0$ and $1 \leq q \leq \infty$. When $q = \infty$, $Z^\star_{i,j,t}$ satisfies an ARMA-like recursion in (4), because causal ARMA can be written as MA process of infinite order.

Data generation

In the following, we shall explain how to analyze the `Y` dataset of observations provided within the package.

We load the dataset `Y` provided within the package:

```
data(Y)
```

The number of conditions I is equal to 3, the number of replications J is equal to 100 and the number of time points T is equal to 15. Data `Y` is generated with $\boldsymbol{\gamma}^\star = (0.5)$ and $\boldsymbol{\eta}^\star$, such that all the $\eta^\star_{i,t} = 0$ except for six of them: $\eta^\star_{1,2} = 0.63$, $\eta^\star_{2,4} = 2.62$, $\eta^\star_{3,3} = 1.69$, $\eta^\star_{3,4} = 2.27$, $\eta^\star_{3,7} = 0.72$ and $\eta^\star_{3,11} = 0.41$. The design matrix X is the design matrix of one-way analysis of variance (ANOVA) with I groups and T observations.

```
I=3
J=100
T=dim(Y)[2]
q=1
```

```
gamma = matrix(c(0.5), nrow = 1, ncol = q)
```

```
active=c(2, 24, 33, 34, 37, 41)
non_active = setdiff(1:(I*T),active)
eta_true=rep(0,(I*T))
eta_true[active]=c(0.63, 2.62, 1.69, 2.27, 0.72, 0.41)
```

```
X=matrix(0,nrow=(I*J),ncol=I)
for (i in 1:I)
{
  X[((i-1)*J+1):(i*J),i]=rep(1,J)
}
```

Initialization

We initialize $\boldsymbol{\gamma^0} = (0)$ and $\boldsymbol{\eta^0}$ to be the coefficients estimated by glm function:

```
gamma_0 = matrix(0, nrow = 1, ncol = q)
eta_glm_mat_0 = matrix(0,ncol=T,nrow=I)
for (t in 1:T)
{
  result_glm_0 = glm(Y[,t]~X-1,family=poisson(link='log'))
  eta_glm_mat_0[,t]=as.numeric(result_glm_0$coefficients)
}
eta_0 = round(as.numeric(t(eta_glm_mat_0)),digits=6)
```

Estimation of $\boldsymbol{\gamma^\star}$

We can estimate $\boldsymbol{\gamma^\star}$ with the Newton-Raphson method. The output is the vector of estimation of $\boldsymbol{\gamma^\star}$. The default number of iterations `n_iter` of the Newton-Raphson algorithm is 100.

```
gamma_est=NR_gamma(Y, X, eta_0, gamma_0, I, J, n_iter = 100)
cat("Estimated gamma: ", gamma_est, "\n")
```

```
## Estimated gamma:  0.5009826
```

This estimation is obtained by taking initial values $\boldsymbol{\gamma^0}$ and $\boldsymbol{\eta^0}$, which can improve once we substitute the initial values by $\hat{\boldsymbol{\gamma}}$ and $\hat{\boldsymbol{\eta}}$ obtained by `variable_selection` function.

Variable selection

We perform variable selection and obtain the coefficients which are estimated to be active and the estimates of $\boldsymbol{\gamma^\star}$ and $\boldsymbol{\eta^\star}$. We take the number of iterations of the algorithm `k_max` equal to 1. We take `min` method (corresponding to the stability selection method with minimal λ), where `threshold` is equal to 0.67 and the number of replications `nb_rep_ss` = 1000. For more details about stability selection and the choice of parameters we refer the reader to [1].

```
result = variable_selection(Y, X, gamma_est, k_max = 1, n_iter = 100,
    method = "min", nb_rep_ss = 1000, threshold = 0.67)
```

```
estim_active = result$estim_active
eta_est = result$eta_est
gamma_est = result$gamma_est
```

```
## True active coefficient pairs:  (1,2) (2,9) (3,3) (3,4) (3,7) (3,11)
```

```
## Estimated active coefficient pairs:  (1,2) (2,9) (3,3) (3,11)
```

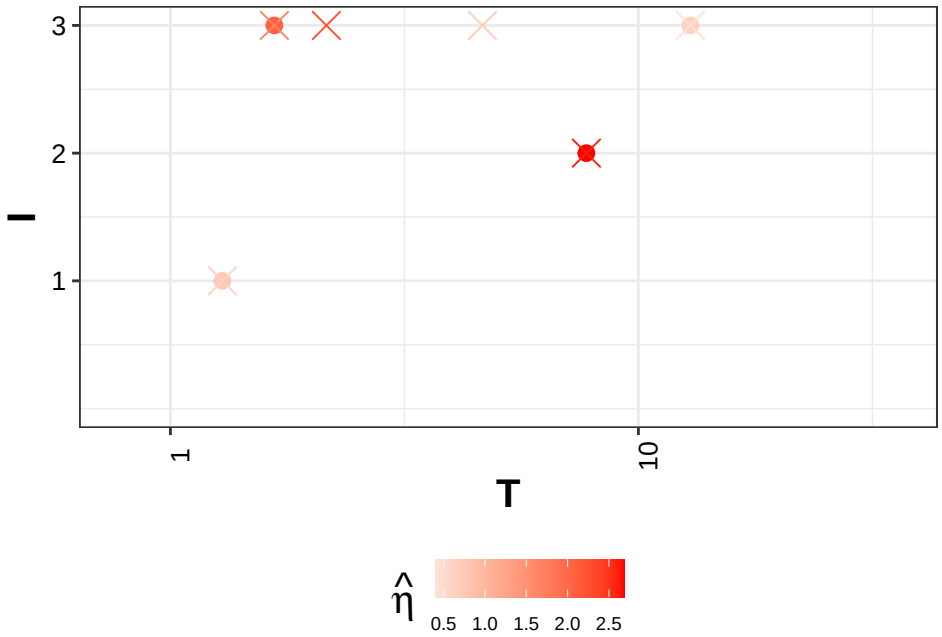
```
## True values of elements :  0.63 2.62 1.69 2.27 0.72 0.41
```

```
## Estimated values of elements:  0.77 2.69 2.1 0 0 0.64
```

Ilustration of the estimation of $\eta^\star$

We display a plot that illustrates which elements of $\boldsymbol{\eta}^\star$ are selected to be active and how close the estimated value $\hat{\eta}_{i,t}$ is to the actual values $\eta^\star_{i,t}$. True values of $\boldsymbol{\eta}^\star$ are plotted in crosses and estimated values are plotted in dots.

```
#First, we make a dataset of estimated etas
eta_df = data.frame(eta_est)
eta_df$t = c(rep(seq(1,T,1),I))
eta_df$I = c(rep(1,T), rep(2,T), rep(3,T))
colnames(eta_df)[1] <- "eta"
eta_df = eta_df[eta_df$eta!=0,]
#Next, we make a dataset of true etas
eta_t_df = data.frame(eta_true)
colnames(eta_t_df)[1] <- "eta"
eta_t_df$I = c(rep(1,T), rep(2,T), rep(3,T))
eta_t_df$t = c(rep(seq(1,T,1),I))
eta_t_df = eta_t_df[eta_t_df$eta !=0,]
#Finally, we plot the result
plot = ggplot()+
  geom_point(data = eta_df, aes(x=t, y=I, color=eta), pch=20, size=3, stroke = 1)+
  geom_point(data= eta_t_df, aes(x=t, y=I, color=eta), pch=4, size=4.5)+
  scale_color_gradient2(name=expression(hat(eta)), midpoint=0,
                        low="steelblue", mid = "white", high ="red")+
  theme_bw()+ylab('I')+xlab('T')+
  theme(legend.position = "bottom")+
  theme(legend.key.size = unit(0.5, 'cm'))+
  theme(legend.title = element_text(size = 15, face="bold"))+
  theme(legend.text = element_text(size = 7, color="black"))+
  scale_y_continuous(breaks=seq(1, I, 1), limits=c(0, I))+
  scale_x_continuous(breaks=c(1, seq(10, T, 10)), limits=c(0, T))+
  theme(axis.text.x = element_text(angle = 90))+
  theme(axis.text=element_text(size=10, color="black"))+
  theme(axis.title=element_text(size=15,face="bold"))
plot
```



References

[1] M. Gomtsyan, C. Lévy-Leduc, S. Ouadah, L. Sansonnet, C. Bailly and L. Rajjou. "Variable selection in multivariate sparse GLARMA models: application to germination control by environment", arXiv:2208.14721