Relatorio de mudanças no código

Silvaneo Viera dos Santos Junior

5/19/2022

Repositórios dos códigos

Repositório das funções do pacote e da aplicação do Shiny:

https://github.com/silvaneojunior/GDLM

Repositório dos gráficos e análises produzidas com o código:

https://github.com/silvaneojunior/Projeto_graduacao

O primeiro repositório possui a versão estável mais recente dos códigos, ademais, ele é restrito aos códigos gerais, sem análises de dados. Este repositória é atualizado regularmente conforme o desenvolvimento do pacote GDLM avança.

O segundo repositório possui versões experimentais do código (o caso normal, por exemplo) e versões antigas, ademais, os ajustes feitos para dados especificos se encontram nele. Eventualmente esse repositório deixará de ser atualizado, permanecendo apenas como um registro do que foi feito.

Mudanças gerais (exceto caso Normal)

- Limpeza do código: remoção de variáveis e comentários não utilizados.
- Simplificação do loop de filtragem: todo o processo definido em um loop (originalmente a primeira iteração era executada fora do loop).
- Otimização do loop de filtragem: minimização de produtos matriciais e de chamadas de funções *gamma*, *digamma* e *trigamma*.
- Separação das funções de filtragem, predição, suavização e ajuste.
- Otimização do loop de suavização: minimização da quantidade de inversões de matrizes.
- Inclusão da possibilidade de variáveis com variância nula: durante a suavização, estas variáveis são removidas das operações.
- Mudança na matriz R_t, inicialmente definida como $\frac{1}{\delta}P_t$, onde $P_t = GC_tG'$ e δ é o fator de desconto (não precisa ser escalar no código, mas por simplicidade, escreverei como um escalar). Na versão atual, temos $R_t = \frac{1}{\delta}P_t + W_t$, onde W_t é uma matriz simétrica definida não-negativa (falta incluir um erro para quando esta restrição não é respeitada pelo usuário). A ideia dessa mudança veio da observação do fato de que, na especificação original, se $C_0 = 0$ (variância à priori), então $C_t = 0$ para todo t. Com a inclusão da matriz W_t, é possível que uma variável tenha variância 0 até um certo tempo e depois receba um choque aleatória.
- Adição da possibilidade de fatores de desconto distintos a cada tempo.
- Padronizações entre os códigos para o caso Poisson (com Linear Bayes) e Multinomial.

Caso multinomial

- Generialização do código para k categorias (conforme o artigo no arxiv).
- Extensão da F_t para regressão dinâmica. No original, $F_t = F$ para todo t (série temporal).
- F_t se tornou um array de dimensão $n \times r \times T$ (n é o número de variáveis latentes, r é a quantidade de series na saída do modelo e T é o tempo final). Para cada tempo, temos uma matriz $n \times r$, sendo que o preditor linear está definido como $F_t'\theta_t$.
- F_t não é mais bloco diagonal, é possível definir variáveis latentes que tem efeito em mais de uma série. Desvantagem: um pouco mais complicado para o usuário fazer a especificação, porém algumas funções auxiliares foram criadas para facilitar o processo, além disso, a aplicação do *Shiny* simplifica o processo.
- Generização da condição inicial para o Newton-Raphson (garantindo inicialização válida): $\vec{x}_{inicial} = (0.01, ..., 0.01, 0.01 \times r)$. Essa inicialização garante que $x_{r+1} - \sum_{i=1}^{r} x_i > 0$ (x_{r+1} está associado ao total de ocorrências).

Caso Normal

Esse script recebeu o mínimo de alteranções necessárias para que o ajuste funcionasse, de modo a evitar introduzir erros antes que resolver os que já estavam presentes

- Correções gerais para igualar o código e as equações do artigo (as mudanças já tinham sido quase integralmente observadas pela Mariane anteriormente).
- Mudança de variáveis do sistema de compatibilização da priori: o sistema passa a ser escrito em termos de $\mu_0 = -\frac{\tau_2}{\tau_1}$, $c_0 = -2\tau_1$, $d_0 = \frac{\tau_2^2}{2\tau_1} - 2\tau_3$ e $n_0 = 2\tau_0 + 1$.
- Simplificação do sistema (mais detalhes no final).
- Inclusão do cálculo analítico do Jacobiano no Newton-Raphson: redução do tempo computacional.

Simplificação do sistema no caso Normal

Sistema original:

$$\begin{aligned}
\frac{(2\tau_0+1)\tau_2^2}{2\tau_1\tau_2^2 - 8\tau_1^2\tau_3} - \frac{1}{2\tau_1} &= (q_1 + f1^2)\exp(f_2 + q_2/2) \\
-\frac{(2\tau_0+1)\tau_2}{\tau_2^2 - 4\tau_1\tau_3} &= f1\exp(f_2 + q_2/2) \\
\frac{4\tau_1(\tau_0+1/2)}{\tau_2^2 - 4\tau_1\tau_3} &= \exp(f_2 + q_2/2) \\
\gamma(\tau_0+1/2) - \log(\frac{\tau_2^2}{4\tau_1 - \tau_3}) &= f_2
\end{aligned}$$

Observe que:

$$-\frac{(2\tau_0+1)\tau_2}{\tau_2^2 - 4\tau_1\tau_3} = -\frac{2\tau_2}{4\tau_1}\frac{4\tau_1(\tau_0+1/2)}{\tau_2^2 - 4\tau_1\tau_3}$$

Usando a equação 3, temos:

$$-\frac{(2\tau_0+1)\tau_2}{\tau_2^2 - 4\tau_1\tau_3} = -\frac{2\tau_2}{4\tau_1}\frac{4\tau_1(\tau_0+1/2)}{\tau_2^2 - 4\tau_1\tau_3} = -\frac{2\tau_2}{4\tau_1}\exp(f_2 + q_2/2) = f1\exp(f_2 + q_2/2)$$

Daí:

$$-\frac{2\tau_2}{4\tau_1} = \mu_0 = f1$$

Veja agora que:

$$\frac{(2\tau_0+1)\tau_2^2}{2\tau_1\tau_2^2 - 8\tau_1^2\tau_3} = \frac{-\tau_2}{2\tau_1}\frac{(2\tau_0+1)\tau_2}{\tau_2^2 - 4\tau_1\tau_3}$$

Usando a equação 2 e que $-\frac{\tau_2}{2\tau_1} = \mu_0 = f_1$, temos:

$$\frac{(2\tau_0+1)\tau_2^2}{2\tau_1\tau_2^2 - 8\tau_1^2\tau_3} = \frac{-\tau_2}{2\tau_1}\frac{(2\tau_0+1)\tau_2}{\tau_2^2 - 4\tau_1\tau_3} = f_1^2\exp(f_2 + q_2/2)$$

Voltando à equação 1:

$$\frac{(2\tau_0+1)\tau_2^2}{2\tau_1\tau_2^2 - 8\tau_1^2\tau_3} - \frac{1}{2\tau_1} = f_1^2\exp(f_2 + q_2/2) - \frac{1}{2\tau_1} = (q_1 + f1^2)\exp(f_2 + q_2/2)$$

Logo:

$$-\frac{1}{2\tau_1} = (q_1 + f1^2)\exp(f_2 + q_2/2) - f_1^2\exp(f_2 + q_2/2) = q_1\exp(f_2 + q_2/2)$$

$$\frac{1}{c_0} = q_1\exp(f_2 + q_2/2)$$

$$c_0 = \frac{1}{q_1\exp(f_2 + q_2/2)}$$

Assim, as duas primeiras equações do sistema possuem solução analítica e que não depende de n_0 e d_0. Vamos agora para a equação 3. Usando que $\frac{n_0}{2} = \tau_0 + 1/2$ e que $\frac{4\tau_1}{\tau_2^2 - 4\tau_1\tau3} = \frac{1}{\frac{\tau_2^2}{4\tau_1} - \tau_3} = \frac{1}{\frac{d_0}{2}}$:

$$\frac{4\tau_1(\tau_0 + 1/2)}{\tau_2^2 - 4\tau_1\tau_3} = \frac{n_0/2}{d_0/2} = \frac{n_0}{d_0} = \exp(f_2 + q_2/2)$$

Ademais, na equação 4, temos:

$$\gamma(\tau_0 + 1/2) - \log\left(\frac{\tau_2^2}{4\tau_1 - \tau_3}\right) = \gamma\left(\frac{n_0}{2}\right) - \log\left(\frac{d_0}{2}\right) = f_2$$

Com a equação 3, podemos escrever d_0 como função linear de n_0, fazendo essa substituição na equação 4, obtemos:

$$\gamma\left(\frac{n_0}{2}\right) - \log\left(\frac{n_0}{2\exp(f_2 + q_2/2)}\right) = f_2$$

Resolvemos o sistema acima usando Newton-Raphson e usando $x = \log(n_0)$ como argumento do sistema para garantir $n_0 > 0$:

$$\gamma\left(\frac{e^x}{2}\right) - x + \log(2) + f_2 + q_2/2 = f_2$$

Ou melhor:

$$\gamma\left(\frac{e^x}{2}\right) - x + \log(2) + q_2/2 = 0$$

É importante ressaltar que a solução deste sistema não depende de f_2, ademais, o processo de resolução é bem estável numericamente, sendo resolvivel mesmo para valores extremos de q_2.

Uma vez obtido o valor de n_0, temos o valor de d_0 através da relação:

$$d_0 = \frac{n_0}{\exp(f_2 + q_2/2)}$$

Problema: se $f_2 + q_2/2$ for muito grande, $\exp(f_2 + q_2/2)$ pode ser computacionalmente intratável, porém, para qualquer valor de q_2, é possível fazer uma mudança de escala nos dados de maneira que $\exp(f_2 + q_2/2)$ se torne tratável, pois ao multiplicarmos os dados por uma constante c, obtemos:

$$f_2^* = f_2 - \log(c),$$

pois f_2 é a log precisão, ademais, q_2 não muda de valor, pois a variância não é afetada pela soma/subtração de constantes.