Universidade Federal do Paraná

Vinicius Ricardo Riffel

Equações de estimação regularizadas

Curitiba
2022

Vinicius Ricardo Riffel

Equações de estimação regularizadas

Trabalho de Conclusão de Curso apresentado à disciplina Laboratório B do Curso de Graduação em Estatística da Universidade Federal do Paraná, como exigência parcial para obtenção do grau de Bacharel em Estatística.

Orientador: Prof. Dr. Wagner Hugo Bonat

Curitiba
2022

Resumo

Os modelos multivariados de covariância linear generalizados compõem um *framework* que permite analisar diversos tipos de conjuntos de dados através de regressão, como dados longitudinais e dados multivariados. Outro tópico relevante em regressão são os métodos de regularização. A regularização é uma restrição imposta ao espaço paramétrico de tal forma que as estimativas tenham valor absoluto menor do que as estimativas sem nenhuma regularização. Esses métodos permitem, por exemplo, a análise de dados em alta dimensão, contudo, geralmente fornecem estimativas enviesadas. No presente trabalho, foram propostos e implementados métodos de regularização nos modelos multivariados de covariância linear generalizados. A proposta se baseia na adição de uma quantia (função de regularização) nas equações de estimação dos modelos. Estudos de simulação foram conduzidos a fim de verificar a adequação da metodologia. Foi visto que o método proposto teve sucesso em regularizar as estimativas e que se comporta de maneira similar às propostas já consolidadas, como os métodos implementados por Friedman, Hastie e Tibshirani (2010). Em nossa pesquisa bibliográfica, não encontramos um trabalho anterior envolvendo regularização dos parâmetros de perturbação na análise de dados correlacionados, o que é possível com o *framework* desenvolvido.

Palavras-chave: Regularização; Equações de estimação; Dados correlacionados; Estatística multivariada; Regressão

Sumário

1 Introdução

Os modelos de regressão são métodos estatísticos que permitem explicar a relação entre um fenômeno assumido como aleatório e outras variáveis. Em outras palavras, um modelo estatístico é uma forma de descrever problemas do mundo real de forma probabilística. É comum que cada uma das técnicas de regressão carreguem suposições inerentes às particularidades dos dados que os modelos buscam aprender. A depender da situação, essas suposições podem ser desnecessárias, ou ainda desvantajosas. Por essa e outras razões, devemos fazer uma escolha minuciosa das técnicas que aplicamos para cada conjunto de dados.

Nos dias atuais, encontramos diversas propostas de modelos de regressão na literatura. Contudo, até o *framework* proposto por Nelder e Wedderburn (1972), chamado de modelos lineares generalizados, ou *generalized linear models* (GLM), havia uma grande restrição que envolvia a distribuição de probabilidades assumida para o fenômeno aleatório e, consequentemente, a relação funcional especificada entre as variáveis também era afetada (PAULA, 2013). Uma limitação dos GLM é que a distribuição de probabilidades da variável resposta é restrita à família exponencial uniparamétrica. Outra restrição é a suposição de independência entre as observações da variável resposta.

Embora os GLM representem um avanço na análise de dados, eles não são viáveis para análise de dados correlacionados, como os dados longitudinais. O método dos mínimos quadrados generalizados é uma alternativa que permite a análise de dados dependentes. Contudo, esse método exige que a matriz de covariância populacional seja conhecida e há limitações na distribuição de probabilidades assumida para a variável resposta. Nesse sentido, as equações de estimação generalizadas, ou *generalized estimating equations* (GEE), podem ser uma alternativa em que ambas as limitações são superadas. Esse método é baseado em quase-verossimilhança, portanto não há restrição na distribuição de probabilidades assumida. Também, o método de estimação é robusto a má especificação da matriz de covariâncias (LIANG; ZEGER, 1995).

Os modelos multivariados de covariância linear generalizados, ou *multivariate covariance generalized linear models* (McGLM) foram desenvolvidos recentemente por Bonat e Jørgensen (2016). O *framework* é baseado nas GEE, porém também é possível ajustar modelos para dados multivariados correlacionados. Além disso, é possível obter diversos outros modelos como casos especiais, como os GLM e as GEE. O *framework* está implementado no pacote `mcglm` (BONAT, 2018) no software R (R Core Team, 2022) e exemplos de aplicação desse podem ser encontrados em Bonat et al. (2018) e Bonat et al. (2021). A proposta dos McGLM representa um avanço importante na literatura de modelos de regressão tendo em vista sua ampla flexibilidade.

Além de dados correlacionados há outros desafios na análise de dados, como

dados em alta dimensão (caso geral em que há mais covariáveis do que observações). Além disso, é primordial que os modelos adotados sejam parcimoniosos, isto é, balanceiem complexidade e explicação do fenômeno de interesse. A busca por um modelo parcimonioso pode ser feita através de seleção de covariáveis. Ambas dificuldades podem ser superadas através de métodos de regularização, também chamado de métodos de penalização. Esses métodos podem ser úteis em outras situações, como presença de multicolinearidade ou até mesmo na predição de novos valores. Tais procedimentos buscam diminuir o valor absoluto das estimativas adicionando um custo em cada variável adicionada, fazendo com que as variáveis menos relevantes tendem a ser menos significativas na análise estatística (BICKEL et al., 2006). A regularização é frequentemente aplicada aos GLM e suas variações, como pode ser visto em Friedman, Hastie e Tibshirani (2010), Fu (2003) fez um desenvolvimento desses métodos para a estimação via equações de estimação, como são as GEE. Inan, Zhou e Wang (2017) fez uma implementação de equações de estimação penalizadas para os GEE, mas há diversas limitações, como o método de regularização utilizado que é restrito aos implementados pelos autores.

Não havia um desenvolvimento nem implementação desses métodos para o *framework* dos McGLM. Desse modo, esse trabalho teve como objetivo desenvolver um método para lidar com os mais diferentes tipos de conjuntos de dados via regularização nos McGLM. Em nossa pesquisa bibliográfica, não encontramos um trabalho anterior envolvendo regularização dos parâmetros de perturbação na análise de dados correlacionados, o que é possível com o *framework* desenvolvido.

2 Metodologia

2.1 Modelos multivariados de covariância linear generalizados

Considere um estudo em que há R variáveis resposta dependentes e que há N medidas repetidas em cada uma delas. Essas medidas repetidas também podem ser dependentes.

Seja $\mathbf{Y}_i = (y_{i1}, y_{i2}, \ldots, y_{iN})^T$, onde $i = 1, 2, \ldots, R$, os vetores de observações das respostas de interesse. A matriz com as respostas será denotada por $\mathbf{Y}_{N\times R} = (\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_R)$ e a esperança desta matriz por $\mathbf{M} = (\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_R)$. Seja $\boldsymbol{\Sigma}_R$ a matriz de covariância do vetor $\mathbf{Y_i}$, enquanto que $\boldsymbol{\Sigma}_b$ denota a matriz covariância entre os vetores de observações.

Seguindo Bonat e Jørgensen (2016), para ajustar um McGLM é necessário especificar uma estrutura de média:

$$E(\mathbf{Y}) = \left(g_1^{-1}(\mathbf{X}_1\boldsymbol{\beta}_1), g_2^{-1}(\mathbf{X}_2\boldsymbol{\beta}_2), \ldots, g_R^{-1}(\mathbf{X}_R\boldsymbol{\beta}_R)\right),$$

em que $g_r(\cdot)$, é uma função de ligação, $\mathbf{X}_r$ é a matriz do modelo de dimensão $N \times R$ e $\boldsymbol{\beta}_r$ é o vetor parâmetros associados à matriz $\mathbf{X}_r$, para $r = 1, \ldots, R$.

Também é necessário a especificação de uma estrutura de covariância:

$$\mathbf{C} = Var(\mathbf{Y}) = \boldsymbol{\Sigma}_R \overset{G}{\otimes} \boldsymbol{\Sigma}_b,$$

em que $\boldsymbol{\Sigma}_R \overset{G}{\otimes} \boldsymbol{\Sigma}_b = \text{Bdiag}\left(\tilde{\boldsymbol{\Sigma}}_1, \tilde{\boldsymbol{\Sigma}}_2, \ldots, \tilde{\boldsymbol{\Sigma}}_R\right)(\boldsymbol{\Sigma}_b \otimes \mathbf{I})\,\text{Bdiag}\left(\tilde{\boldsymbol{\Sigma}}_1^T, \tilde{\boldsymbol{\Sigma}}_2^T, \ldots, \tilde{\boldsymbol{\Sigma}}_R^T\right)$ é o produto generalizado de Kronecker (MARTINEZ-BENEITO, 2013), $\tilde{\boldsymbol{\Sigma}}_r$ denota a matriz triangular inferior obtida na decomposição de Choslesky da matriz $\boldsymbol{\Sigma}_r$, o operador Bdiag$(\cdot)$ denota uma matriz bloco diagonal e $\mathbf{I}$ é uma matriz identidade de dimensão $N \times N$.

A matriz de covariância do i-ésimo vetor $\mathbf{Y}_r$ é definida por

$$\boldsymbol{\Sigma}_r = V(\boldsymbol{\mu}_r; p_r)^{\frac{1}{2}}\boldsymbol{\Omega}(\boldsymbol{\tau}_r)V(\boldsymbol{\mu}_r; p_r)^{\frac{1}{2}},$$

em que $V(\boldsymbol{\mu}_r; p_r) = \text{diag}(v(\boldsymbol{\mu}_r; p_r))$ denota uma matriz diagonal, em que cada entrada é a função de variância aplicada em cada valor de $\boldsymbol{\mu}_r$, $\boldsymbol{\Omega}(\boldsymbol{\tau}_r)$ é chamada de matriz de dispersão, isto é, os componentes de variância que não dependem de $\boldsymbol{\mu}_r$. A especificação da estrutura de dependência entre as medidas repetidas são incluídas na matriz de dispersão da seguinte forma:

$$h(\boldsymbol{\Omega}(\boldsymbol{\tau}_r)) = \tau_{r0}Z_{r0} + \tau_{r1}Z_{r1} + \ldots + \tau_{rD}Z_{rD}, \tag{2.1}$$

em que $h(\cdot)$ é a função de ligação de covariância, τ_{rd} são os parâmetros de dispersão e cada matriz $Z_{r.}$ é especificada de modo a criar a estrutura desejada para a r-ésima variável resposta. Por exemplo, se desejamos definir uma estrutura de simetria composta a Equação (2.1) deve ser especificada da seguinte forma:

$$h\left(\boldsymbol{\Omega}(\boldsymbol{\tau}_r)\right) = \tau_1 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

.

A estimação dos parâmetros desse modelo pode ser feita pelo procedimento proposto por Jørgensen e Knudsen (2004), onde os parâmetros são separados em dois vetores: $\boldsymbol{\theta} = (\boldsymbol{\beta}^T, \boldsymbol{\lambda}^T)^T$, em que $\boldsymbol{\beta}^T$ são os parâmetros de regressão (da estrutura de média) e $\boldsymbol{\lambda}^T$ são os parâmetros de perturbação (da estrutura de covariância). Para simplificar a notação, seja $\mathcal{Y}$ o vetor de respostas empilhados e $\mathcal{M}$ a esperança deste vetor, ambos de dimensão $NR \times 1$.

As funções de estimação do vetor $\boldsymbol{\beta}$ e $\boldsymbol{\lambda}$ são, respectivamente, dadas por:

$$\begin{aligned} \psi_{\boldsymbol{\beta}}(\boldsymbol{\beta}, \boldsymbol{\lambda}) &= \boldsymbol{D}^T \mathbf{C}^{-1} \left(\mathcal{Y} - \mathcal{M}\right) \\ \psi_{\boldsymbol{\lambda}_i}(\boldsymbol{\beta}, \boldsymbol{\lambda}) &= \operatorname{tr}\left(W_{\boldsymbol{\lambda}_i}(\mathbf{r}^T\mathbf{r}) - \mathbf{C}\right), \end{aligned} \tag{2.2}$$

em que $\mathbf{D} = \nabla_{\boldsymbol{\beta}}\mathcal{M}$, $W_{\boldsymbol{\lambda}_i} = -\partial\mathbf{C}^{-1}/\partial\boldsymbol{\lambda}_i$ e $\mathbf{r} = (\mathcal{Y} - \mathcal{M})$. A solução de $\psi_\beta = \mathbf{0}$ e $\psi_\lambda = \mathbf{0}$ são os respectivos estimadores dos vetores $\boldsymbol{\beta}$ e $\boldsymbol{\lambda}$. Nesse contexto, o algoritmo proposto por Jørgensen e Knudsen (2004), chamado de *modified chaser*, pode ser escrito como:

$$\begin{aligned} \boldsymbol{\beta}^{(i+1)} &= \boldsymbol{\beta}^{(i)} - S_{\boldsymbol{\beta}}^{-1}\psi_{\boldsymbol{\beta}}(\boldsymbol{\beta}^{(i)}, \boldsymbol{\lambda}^{(i)}) \\ \boldsymbol{\lambda}^{(i+1)} &= \boldsymbol{\lambda}^{(i)} - S_{\boldsymbol{\lambda}}^{-1}\psi_{\boldsymbol{\lambda}_i}(\boldsymbol{\beta}^{(i+1)}, \boldsymbol{\lambda}^{(i)}), \end{aligned} \tag{2.3}$$

em que $S_{\boldsymbol{\beta}}$ e $S_{\boldsymbol{\lambda}}$ são as respectivas matrizes de sensitividade de $\psi_{\boldsymbol{\beta}}$ e $\psi_{\boldsymbol{\lambda}}$, dadas por:

$$\begin{aligned} S_{\boldsymbol{\beta}} &= E\left(\nabla_{\boldsymbol{\beta}}\psi_{\boldsymbol{\beta}}\right) = \boldsymbol{D}^T\mathbf{C}^{-1}\boldsymbol{D} \\ S_{\boldsymbol{\lambda}_{ij}} &= E\left(\frac{\partial}{\partial\boldsymbol{\lambda}_i}\psi_{\boldsymbol{\lambda}_j}\right) = -\operatorname{tr}(W_{\boldsymbol{\lambda}_i}\mathbf{C}W_{\boldsymbol{\lambda}_j}\mathbf{C}). \end{aligned}$$

Informações detalhadas sobre a obtenção da matriz $W_{\boldsymbol{\lambda}}$ e construção de intervalos de confiança para o vetor $\boldsymbol{\theta}$ são encontradas em Bonat e Jørgensen (2016).

2.1.1 Casos especiais

Na presente subseção é explicitado a forma de obter outros modelos como caso particular de um McGLM.

2.1.1.1 GLM

Na Seção 2.1 foi apresentada a especificação dos McGLM. Nota-se que nenhuma suposição sobre a distribuição das variáveis resposta foi feita. Por ser estimado via equações de estimação pode-se fazer o uso de quase-verossimilhança (WEDDERBURN, 1974) para obter a estrutura de média de um GLM. Para obter a estrutura de covariância de um GLM, deve-se definir $\boldsymbol{\Omega}(\tau) = \tau I$. Ou seja:

$$\begin{aligned}\boldsymbol{\Sigma} =& V(\mu; p)^{\frac{1}{2}}(\tau I)V(\mu; p)^{\frac{1}{2}} \\ =& \tau V(\mu; p).\end{aligned}$$

A suposição distribucional exigida pelos GLM é feita por meio da função de variância $V(\mu; p)$.

2.1.1.2 GEE

A equação de estimação para a estrutura de média é a mesma que a utilizada pelos GEE. Sendo assim, para se obter um GEE como caso particular de um McGLM deve-se especificar a matriz $\boldsymbol{\Omega}(\tau)$ de maneira a obter a matriz de correlação de trabalho desejada. Exemplos dessas matrizes podem ser encontradas em Diggle et al. (2002).

2.1.1.3 Modelo Misto

Uma outra abordagem para a modelagem de dados longitudinais são os modelos mistos. O modelo se dá devido à mistura entre efeitos fixos e aleatórios (DIGGLE et al., 2002), induzindo uma estrutura de correlação nas observações. Embora a especificação de um modelo misto seja diferente dos modelos marginais, como é o caso dos McGLM, é possível definir uma estrutura de covariância de um modelo misto através do preditor matricial. A título de ilustração, no caso de um modelo com apenas o intercepto aleatório, a matriz $\boldsymbol{\Omega}(\tau)$ deve ser uma matriz bloco diagonal, em que cada bloco é uma matriz em cada entrada é igual a 1.

2.2 Regularização

Os métodos de regularização são restrições adicionadas no espaço paramétrico. Essas restrições são adicionadas somando-se uma função dos parâmetros (função *penalty*) na estimação, fazendo com que as estimativas tenham valor absoluto menor do que as estimativas obtidas sem nenhum tipo de regularização. Por exemplo, a estimação pelo método de mínimos quadrados consiste em encontrar o valor de $\boldsymbol{\beta}$ que minimiza a função $(y - X\boldsymbol{\beta})(y - X\boldsymbol{\beta})^T$. Na estimação por mínimos quadrados regularizados, a função a ser otimizada é (TIBSHIRANI et al., 2012):

$$RSS^* = (y - X\boldsymbol{\beta})(y - X\boldsymbol{\beta})^T + \gamma p(\boldsymbol{\beta}),$$

em que γ é uma constante positiva que controla o grau de restrição que é imposta ao espaço paramétrico, isto é, um γ próximo a zero representa um cenário com menor restrição, e $p(\boldsymbol{\beta})$ é a função *penalty*. A escolha do valor de γ é, em geral, feita por validação cruzada.

Diferentes funções podem ser assumidas para $p(\boldsymbol{\beta})$. Hoerl e Kennard (1970) introduziram a chamada penalização *Ridge*, em que $p(\boldsymbol{\beta}) = |\beta_i|^2$, note que o intercepto não é penalizado. Os estimadores obtidos sob essa penalização têm variância menor que os estimadores de mínimos quadrados, porém são enviesados. Tibshirani (1996) apresentam a restrição LASSO, em que $p(\boldsymbol{\beta}) = |\beta_i|$. Na restrição LASSO, as estimativas assumem valor 0 mais rapidamente do que a restrição *Ridge*. Por essa razão, a regularização LASSO pode ser usada para selecionar variáveis. Uma proposta que generaliza as propostas anteriores foi feita por Frank e Friedman (1993), em que $p(\boldsymbol{\beta}) = |\beta_i|^\alpha$ para $\alpha > 0$. A penalização *elastic net* (ZOU; HASTIE, 2005) utiliza a restrição *Ridge* e LASSO, neste caso, a função de penalização fica dada por $p(\boldsymbol{\beta}) = \gamma(l_1 \sum_{i=1}^{R} |\beta_i| + l_2 \sum_{i=1}^{R} |\beta_i|^2)$.

No contexto dos McGLM, a regularização foi imposta nas funções de estimação. Através da regularização na estrutura de média, podemos estimar modelos com dados em alta dimensão, realizar seleção de variáveis *etc*. Quando a regularização é imposta à estrutura de covariância, pode-se reduzir o número de parâmetros estimados, e consequentemente diminuir a complexidade do modelo estimado.

Seguindo Fu (2003), foi somado às funções de estimação um vetor $\Gamma(\cdot)$ em que cada entrada é a derivada do módulo da função *penalty*. A motivação da utilização da derivada da função *penalty* se da pois as funções de estimação dos McGLM são funções quase-escore, uma generalização da função escore. Por exemplo, no caso da penalização *Ridge* a quantidade somada será $\Gamma(p'(|\boldsymbol{\beta}|)) = 2\gamma \left(\frac{\beta}{|\beta|}\right)^T \boldsymbol{\beta}$. A fração $\frac{\beta}{|\beta|}$ será uma indefinição a conforme $\boldsymbol{\beta} \to 0$, por essa razão na implementação da regularização foi adicionada uma constante de 1×10^{-6} no denominador para evitar problemas numéricos.

As funções de estimação regularizadas ficam dadas por:

$$\psi^*_{\boldsymbol{\beta}}(\boldsymbol{\beta}, \boldsymbol{\lambda}) = \boldsymbol{D}^T \mathbf{C}^{-1} (\mathcal{Y} - \mathcal{M}) - \boldsymbol{\gamma} \odot \Gamma(p'_1(|\boldsymbol{\beta}|))$$
$$\psi^*_{\boldsymbol{\lambda}_i}(\boldsymbol{\beta}, \boldsymbol{\lambda}) = \operatorname{tr}\left(W_{\boldsymbol{\lambda}_i} \mathbf{r}^T \mathbf{r}) - \mathbf{C}\right) - \boldsymbol{\gamma}_i \odot \Gamma(p'_2(|\boldsymbol{\lambda}_i|)))$$

em que $\boldsymbol{\gamma}$ é um vetor que controla o grau de regularização de cada resposta, $p(\cdot)$ é uma função de penalização e $\odot$ denota o produto de Hadamard.

As matrizes de sensitividade serão afetadas pela inclusão da penalização, sendo dadas por:

$$S^*_{\boldsymbol{\beta}} = E\left(\nabla_{\boldsymbol{\beta}}\psi_{\boldsymbol{\beta}}\right) = \boldsymbol{D}^T\mathbf{C}^{-1}\boldsymbol{D} - \boldsymbol{\gamma} \odot \frac{\partial}{\partial \boldsymbol{\beta}}\Gamma(p_1'(|\boldsymbol{\beta}|))$$
$$S^*_{\boldsymbol{\lambda}_{ij}} = E\left(\frac{\partial}{\partial \boldsymbol{\lambda}_i}\psi_{\boldsymbol{\lambda}_j}\right) = -\text{tr}(W_{\boldsymbol{\lambda}_i}\mathbf{C}W_{\boldsymbol{\lambda}_j}\mathbf{C}) - \boldsymbol{\gamma}_i \odot \frac{\partial}{\partial \boldsymbol{\lambda}_i}\Gamma(p_2'(|\boldsymbol{\lambda}_i|))$$

Ainda pode-se utilizar o método apresentado na Equação 2.3 para a estimação, mas agora com as equações de estimação e matrizes de sensitividade modificadas:

$$\boldsymbol{\beta}^{(i+1)} = \boldsymbol{\beta}^{(i)} - S^{*-1}_{\boldsymbol{\beta}}\psi^*_{\boldsymbol{\beta}}(\boldsymbol{\beta}^{(i)}, \boldsymbol{\lambda}^{(i)})$$
$$\boldsymbol{\lambda}^{(i+1)} = \boldsymbol{\lambda}^{(i)} - S^{*-1}_{\boldsymbol{\lambda}}\psi^*_{\boldsymbol{\lambda}_i}(\boldsymbol{\beta}^{(i+1)}, \boldsymbol{\lambda}^{(i)})$$

2.3 Implementação

Os McGLM foram implementados no software `R` (R Core Team, 2022) por Bonat (2018) em um pacote nomeado `mcglm`. A implementação computacional do método apresentado na Seção 2.2 foi feita no pacote `mcglm`.

Foram adicionada duas novas funções ao pacote, chamadas de `mc_penalization` e `mc_penalization_cov`, responsáveis por aplicar a penalização aos vetores das estimativas de regressão e perturbação, respectivamente. As funções encontram-se no Apêndice A. Também foram adicionados 4 argumentos na função `mcglm`:

- `penalization`: utilizado na penalização da estrutura de média. Deve-se passar uma *string* de mesmo nome da função de penalização. A função de penalização é criada pelo usuário e ela deve retornar uma lista de tamanho 2 nomeada como `first` e `second`. O *slot* `first` deve conter a primeira derivada da função de penalização e o *slot* `second` deve conter a segunda derivada da função de penalização.

- `penalization_cov`: a estrutura é idêntica ao comentado no item acima, mas penaliza-se a estrutura de covariância.

- `gamma`: lista contendo o grau de penalização para cada resposta da estrutura de média.

- `gamma_cov`: lista contendo o grau de penalização para cada resposta da estrutura de covariância.

Por padrão da implementação, os interceptos ($\hat{\tau}_{r0}$ inclusos) não são penalizados, assim como as correlações da matriz $\widehat{\boldsymbol{\Sigma}}_b$.

Outro fator importante da implementação é que o usuário deve padronizar as variáveis das matrizes $\mathbf{X}_r$, uma vez que isso pode interferir na escolha do vetor $\boldsymbol{\gamma}$ (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

Uma outra implementação disponível no software `R` que utiliza equações de estimação generalizadas penalizadas foi feita por Inan, Zhou e Wang (2017), no pacote nomeado `PGEE`. Embora esse trabalho represente um avanço na análise de dados, ele sofre de algumas restrições. Por exemplo, o usuário está limitado às funções de variâncias e estruturas de correlação implementadas pelos autores. Também, não é possível modificar as penalizações implementadas.

3 Resultados

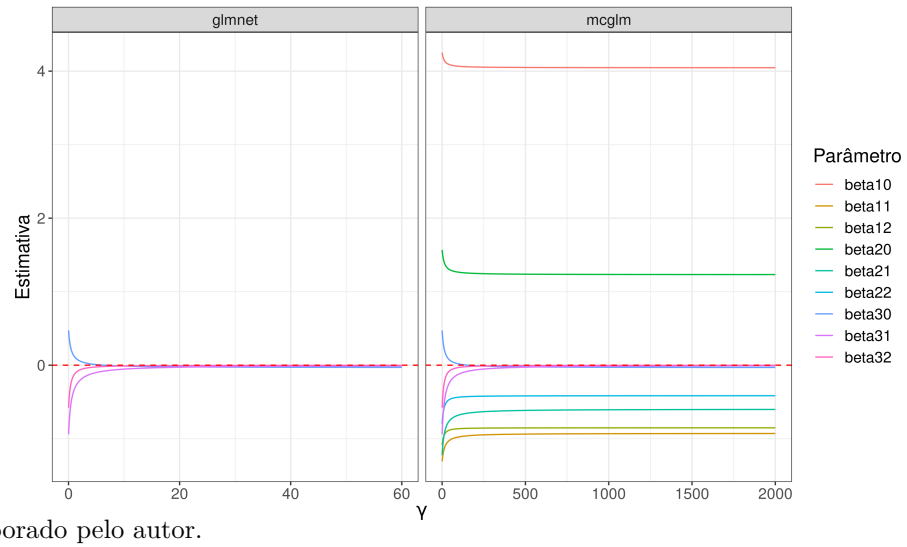
3.1 Aplicação em dados reais

3.1.1 Aplicação 1

Com o objetivo de verificar o impacto das regularizações nas estruturas de média e covariância, ajustamos os modelos a um conjunto de dados reais e variamos o valor do parâmetro de *tunning* (γ). O código `R` encontra-se no Apêndice B.

Os dados utilizados contém o valor de variáveis químicas coletadas em três níveis diferentes do solo em 50 locais cultivados com teca (*Tectona grandis*). O estudo foi conduzido em 2015 em duas fazendas com as plantações no Mato Grosso. A seleção dos locais do estudo foi realizada através de caminhadas pelo local, cobrindo toda a área cultivada de 1869 ha. Durante esse processo, foram realizadas observações de campo e a delimitação de parcelas de acordo com as característica do solo, posição e o nível de desenvolvimento da cultura. Foram alocadas 50 parcelas com 600 m^2 (20×30 m). Como critério de seleção, somente campos com área maior que 7 ha foram selecionados, utilizando-se somente áreas com a mesma densidade de plantação e práticas de plantio com idade de 13 a 14 anos. O objetivo do estudo foi analisar o padrão da concentração de cations (K^+, Ca^{2+} e Mg^{2+}) em três profundidades do solo: [0, 9), [9, 40) e [40, 80]. O conjunto de dados está disponível no pacote `EACS` (ZEVIANI, 2019) do software R, sob o nome de `teca_qui`.

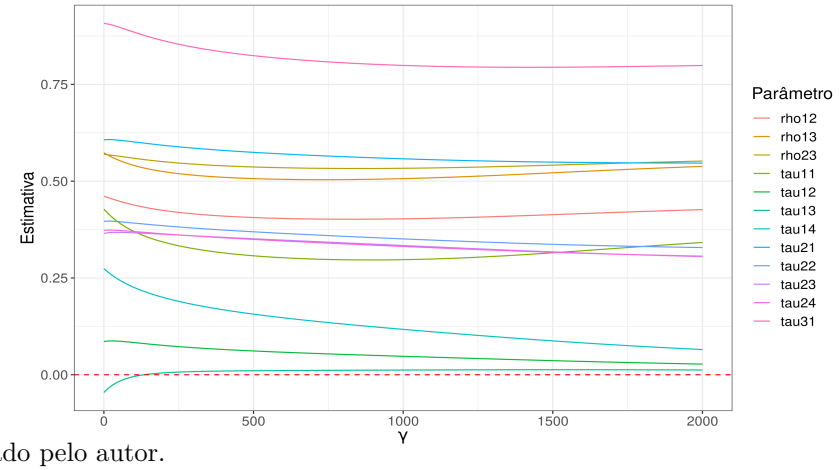
Figura 1 – Valor das estimativas dos parâmetros de regressão versus a restrição imposta na estimação dos parâmetros utilizando McGLM (dir.) e utilizando glmnet (esq.). Em ambos os cenários utilizou-se restrição *Ridge*



Fonte: Elaborado pelo autor.

A Figura 1 apresenta as das estimativas dos parâmetros de regressão utilizando o método implementado na Seção 2.2 e as estimativas obtidas via máxima verossimilhança penalizada, utilizando o método implementado no pacote `glmnet`. Em ambos os casos, como variável preditora temos apenas o nível do solo em que a amostra foi coletada, um fator de três níveis. O McGLM permite a modelagem de diversas variáveis resposta, portanto, foi modelada o logaritmo das três concentrações de interesse (K^+, Ca^{2+} e Mg^{2+}), enquanto que devido as restrições de implementação do `glmnet` apenas o logaritmo da concentração de magnésio foi modelada. Em ambos os casos foi assumido normalidade. Para fins de comparação, apenas a concentração de magnésio foi penalizada e assumimos uma estrutura independente nas medidas repetidas em cada parcela. A penalização utilizada foi *Ridge*. Como o esperado, o aumento da restrição na equação de estimação ocasionou um decréscimo nos valores estimados, da mesma forma com o método de máxima verossimilhança penalizada. Ao lado disso, é nítido que o comportamento dos métodos é similar a medida que aumentamos o grau de restrição. Também nota-se que a penalização de uma resposta tem um baixo impacto nos valores das estimativas das outras variáveis resposta, o que pode facilitar o processo de validação cruzada para seleção dos valores do vetor $\boldsymbol{\gamma}$ no caso de mais de uma variável resposta resposta penalizada, fazendo-se a seleção dos valores individualmente.

Figura 2 – Valor das estimativas dos parâmetros de perturbação versus a restrição *Ridge* imposta na estimação dos parâmetros



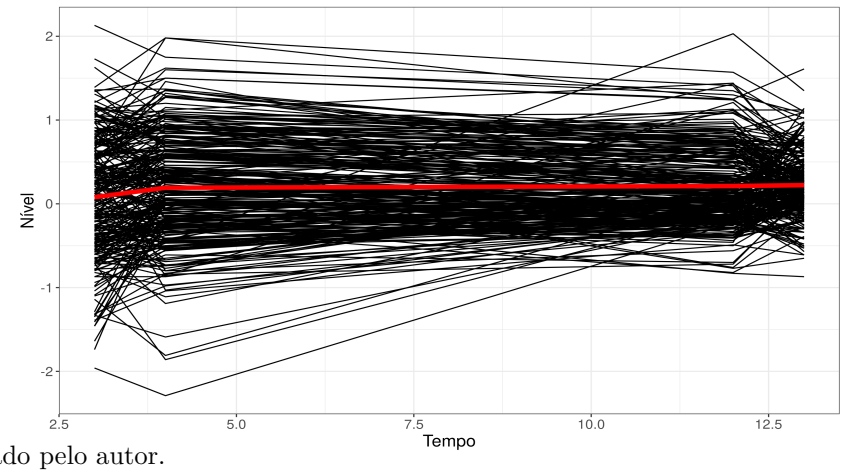
Fonte: Elaborado pelo autor.

A Figura 2 mostra os resultados das estimativas dos parâmetros de perturbação em um cenário em que a estrutura de covariância é regularizada. Foi assumido uma estrutura independente para a concentração de magnésio e não estruturada para as concentrações de cálcio e potássio. Foi assumido normalidade e não penalizamos a estrutura de média. Apenas a concentração de cálcio foi penalizada e utilizamos penalização *Ridge*. Assim como no caso anterior, as estimativas não penalizadas (intercepto, estimativas de outras respostas e correlação) foram pouco afetadas pela regularização. Também, como o esperado, quanto maior foi a restrição, menor foi o valor estimado.

3.1.2 Aplicação 2

Dados de expressão genética do ciclo celular foram coletados por Spellman et al. (1998) em um experimento onde os níveis de mRNA de todo o genoma de 6179 indivíduos nos estágios M/G1-G1-S-G2-M. Um ponto importante deste estudo foi descrever os fatores de transcrição que regulam os níveis da expressão genética da levedura. O conjunto de dados `yeastG1` disponível no pacote `PGEE` contém dados de 283 indivíduos que participaram deste experimento. O conjunto contém 4 medições no estágio G1 para cada indivíduo. A variável resposta é o nível da expressão genética, como covariáveis há o tempo em que o indivíduo foi mensurado (3, 4, 12 ou 13) e 96 fatores de transcrição. O número elevado de covariáveis em relação ao número de indivíduos justifica a utilização de algum tipo de regularização. A penalização *Ridge* parece ser mais adequada, uma vez que há muitas covariáveis correlacionadas. O objetivo desta aplicação foi o ajuste de um McGLM regularizado para fins preditivos do nível de expressão genética dado os fatores de transcrição. As covariáveis e a variável resposta estão padronizadas. O *script* `R` desta aplicação está no Apêndice C.

Figura 3 – Gráfico de perfis para o *dataset* `yeastG1`. A linha em vermelho representa a média do nível para cada tempo



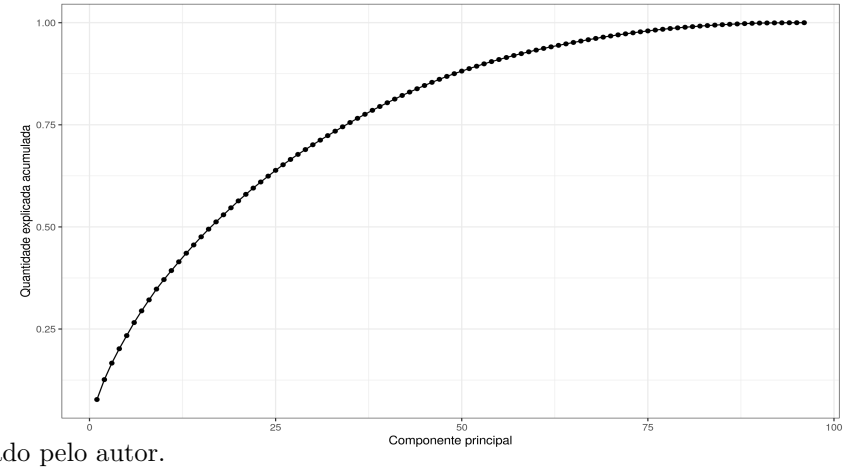
Fonte: Elaborado pelo autor.

A Figura 3 apresenta os gráfico de perfis dos indivíduos e a média em cada tempo (em vermelho) mensurado. Para fins de redução de dimensionalidade foi aplicado uma análise de componentes principais à base de dados. A Figura 4 apresenta a quantidade explicada da variabilidade acumulada. Nota-se que cada componente explica pouco da variação, e portanto não foi possível reduzir a dimensionalidade por meio desta técnica.

Para a estrutura de média, foi considerada uma variância constante e função de ligação identidade. Para a estrutura de covariância, foi assumido com caso similar ao não estruturado, mas com o intercepto matricial τ_0. Assim como anteriormente, a penalização escolhida foi *Ridge*.

A seleção do valor de γ foi feita via validação cruzada com 10 *folds*. Para preservar

Figura 4 – Quantidade explicada acumulada por componente principal



Fonte: Elaborado pelo autor.

a estrutura de dependência, a validação foi feita removendo-se 10% dos indivíduos para a validação, repetindo esse processo 10 vezes (10 *folds*) para cada valor de γ. A medida de erro considerada foi o erro quadrático médio (EQM), ou seja, o melhor valor de γ foi aquele que apresentou o menor EQM do valor predito. A Figura 5 apresenta os valores testados para γ e seu respectivo EQM. Nota-se um ganho expressivo no EQM do valor predito ao incluir alguma regularização na estimação. A fim de buscar um modelo parcimonioso, optou-se por escolher $\gamma = 2.86$.

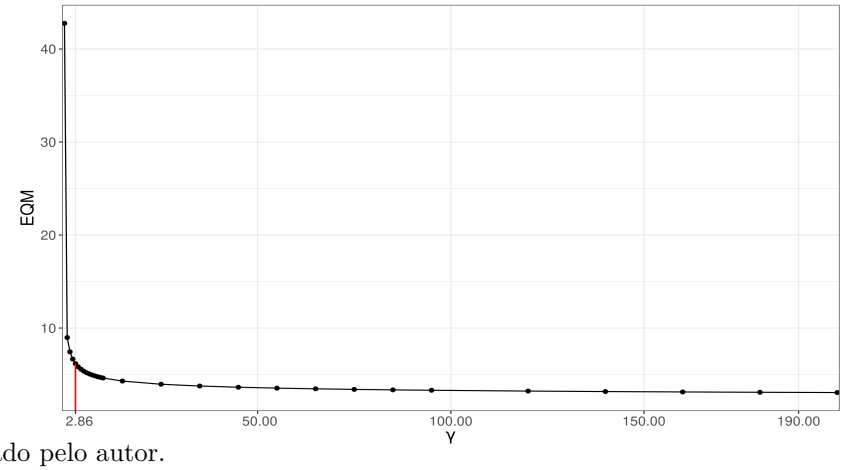
Com esse valor de γ, 8 fatores de transcrição são significativos a um nível de 5%, esses mesmos 8 foram significativos para o modelo sem regularização. A interpretação dos coeficientes não é conveniente devido ao vício das estimativas ao introduzir a regularização, porém, o modelo com a regularização apresenta maior capacidade preditiva.

Para fins de diagnóstico do modelo regularizado, a Figura 6 apresenta os gráficos de resíduos de Pearson versus valores ajustados e um gráfico quantil-quantil dos resíduos e da distribuição normal padrão. Não há indícios de falta de ajuste ou padrão nos resíduos. Isso é um indicador de um bom ajuste ao conjunto de dados pelo modelo proposto.

3.1.3 Aplicação 3

Notas de 88 alunos foram coletadas por Mardia, Kent e Bibby (1979). O conjunto de dados está disponível no pacote `bnlearn` (SCUTARI, 2010) sob o nome de `marks`. As notas (de 0 a 100) registradas foram em cinco disciplinas: álgebra (ALG), análise (ANL), mecânica (MECH), estatística (STAT) e vetores (VECT). Para o ajuste do McGLM, foi considerado que cada uma das disciplinas era uma medida repetida em cada aluno. O interesse nesta aplicação foi definir uma rede de relações entre as disciplinas por meio da matriz de correlação. O *script* `R` desta aplicação está no Apêndice D. A Figura 7 apresenta os *boxplots* das notas dos alunos em cada disciplina. A matriz de correlação entre as variáveis é dada por:

Figura 5 – Valores de γ versus o erro quadrático médio do valor predito. A linha pontilhada representa o valor escolhido como ótimo



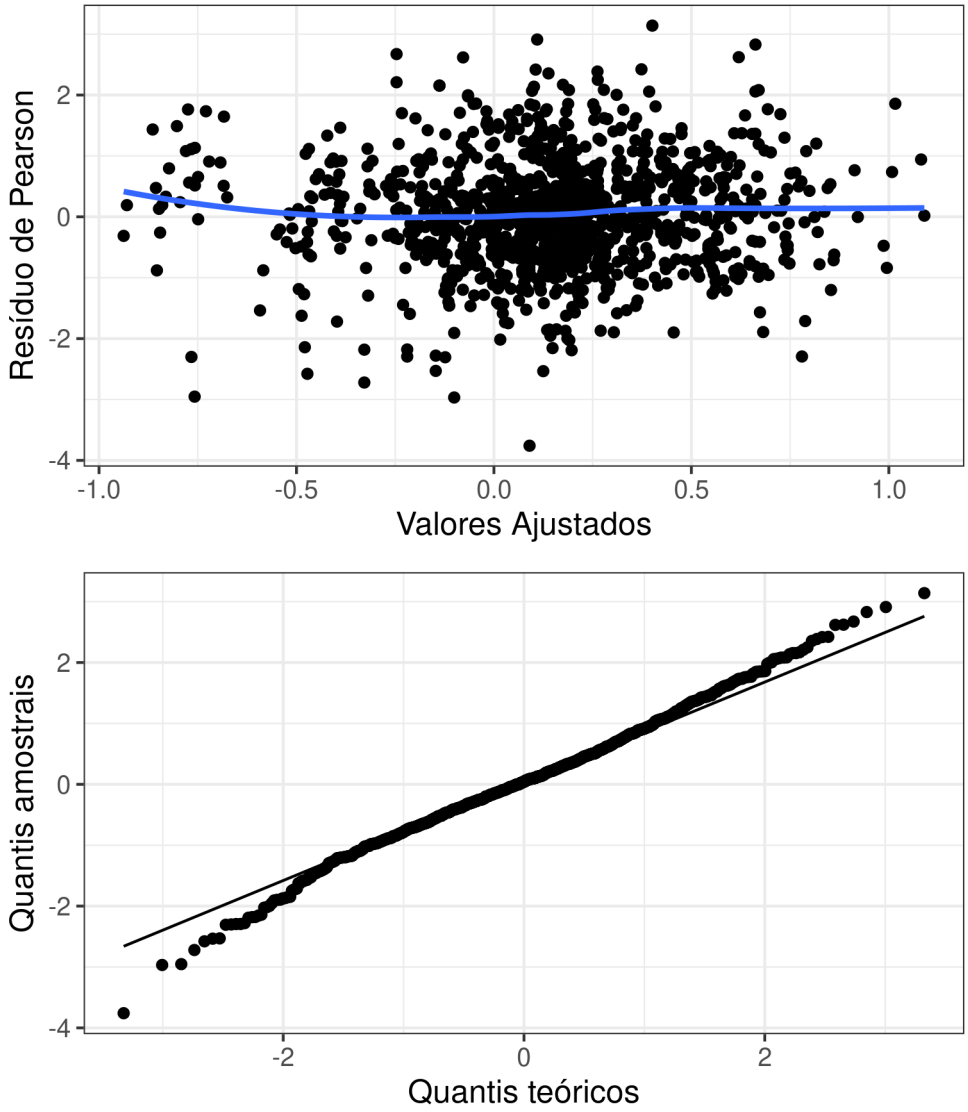
Fonte: Elaborado pelo autor.

$$\begin{array}{c} \\ MECH \\ VECT \\ ALG \\ ANL \\ STAT \end{array} \begin{array}{c} \begin{array}{ccccc} MECH & VECT & ALG & ANL & STAT \end{array} \\ \begin{pmatrix} 1.00 & 0.55 & 0.55 & 0.41 & 0.39 \\ 0.55 & 1.00 & 0.61 & 0.48 & 0.44 \\ 0.55 & 0.61 & 1.00 & 0.71 & 0.66 \\ 0.41 & 0.48 & 0.71 & 1.00 & 0.61 \\ 0.39 & 0.44 & 0.66 & 0.61 & 1.00 \end{pmatrix} \end{array}$$

Foi ajustado um modelo linear para a estrutura de média, enquanto que a estrutura de covariância foi definida como não estruturada. Dado os fins da aplicação, foi escolhida uma função de ligação de covariância inversa, isto é, modelamos a *precisão* no lugar da covariância. A conclusão pela existência de relação significativa entre as variáveis foi feita via um teste de significância (teste *score*) de cada coeficiente de dispersão e uma regularização *Ridge* foi feita na estrutura de covariância. A escolha do valor de γ foi realizada através da divisão do conjunto de dados entre treino (70%) e teste (30%). O procedimento de validação cruzada não pode ser realizado devido ao baixo número de indivíduos na base dados, o que ocasionava uma alta variabilidade dos valores da correlação nas bases de teste - e que, portanto, diferem significativamente da correlação amostral da base completa.

Jørgensen e Knudsen (2004) demonstraram que o vetor $\boldsymbol{\beta}$ e o vetor $\boldsymbol{\lambda}$ são insensíveis, isto é, ao variar os valores de $\boldsymbol{\lambda}$ haverá uma baixa variação nos valores de $\boldsymbol{\beta}$. Desta forma, a medida de erro tomada na escolha de γ não pôde ser feita através dos valores preditos das notas dos alunos. Como alternativa, tomou-se a matriz de correlação da base de testes para a validação. Assim, para cada valor de γ, foi ajustado um McGLM conforme descrito acima e construída a matriz de correlação estimada. Para cada um dos coeficientes da matriz de correlação, foi calculado o erro quadrático médio e ao final somou-se tais valores.

Figura 6 – Gráficos de diagnóstico para o McGLM ajustado com o parâmetro de regularização ótimo

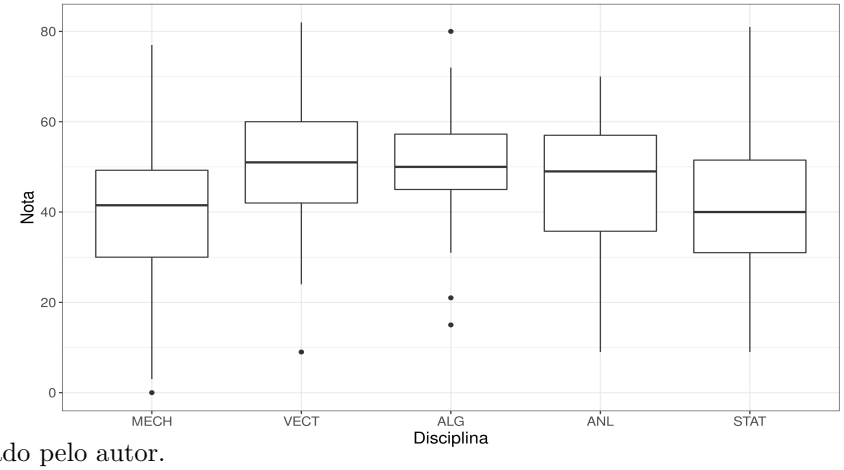


Fonte: Elaborado pelo autor.

A Figura 8 apresenta o resultado da validação. O valor ótimo (na escala logarítmica) foi de 13.94.

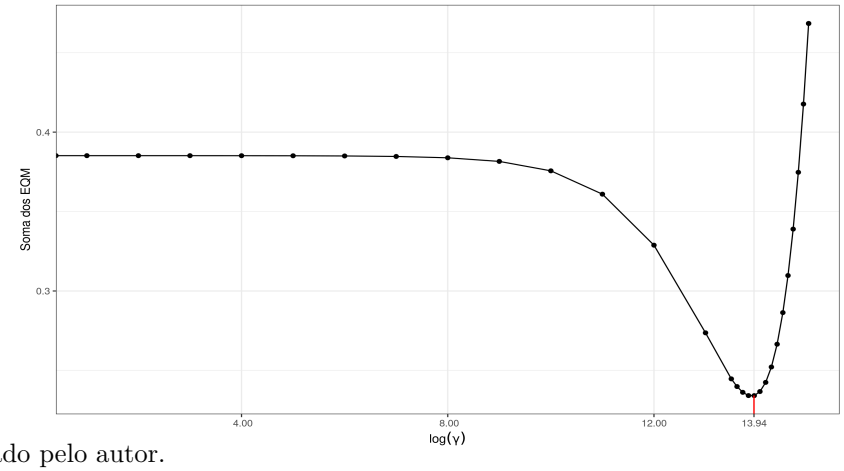
A Tabela 1 apresenta os valores estimados para os parâmetros de dispersão, os seus respectivos erros-padrão e o valor da estatística de teste. Foi considerado um nível de significância de 5%. A Figura 9 mostra a rede estimada através do procedimento. A matriz de correlação estimada pelo modelo regularizado é dada por:

Figura 7 – Boxplot das disciplinas no conjunto de dados `marks`



Fonte: Elaborado pelo autor.

Figura 8 – Soma dos EQM versus os valores de γ



Fonte: Elaborado pelo autor.

$$
\begin{array}{c} \\ MECH \\ VECT \\ ALG \\ ANL \\ STAT \end{array}
\begin{array}{c}
\begin{array}{ccccc} MECH & VECT & ALG & ANL & STAT \end{array} \\
\left(\begin{array}{ccccc}
1.00 & 0.48 & 0.41 & 0.31 & 0.33 \\
0.48 & 1.00 & 0.43 & 0.37 & 0.40 \\
0.41 & 0.43 & 1.00 & 0.49 & 0.41 \\
0.31 & 0.37 & 0.49 & 1.00 & 0.44 \\
0.33 & 0.40 & 0.41 & 0.44 & 1.00
\end{array}\right)
\end{array}
$$

Sabendo que foi testado múltiplas hipóteses, poderia-se utilizar algum tipo de correção dos p-valores. No caso da utilização da correção de Bonferroni, o limiar de rejeição da hipótese nula seria $|Z| > 2.84$. Nesta análise, a mudança seria a não rejeição da hipótese de não relação entre VECT–ALG e ALG–STAT. A rede formada após a correção é mostrada na Figura 10.

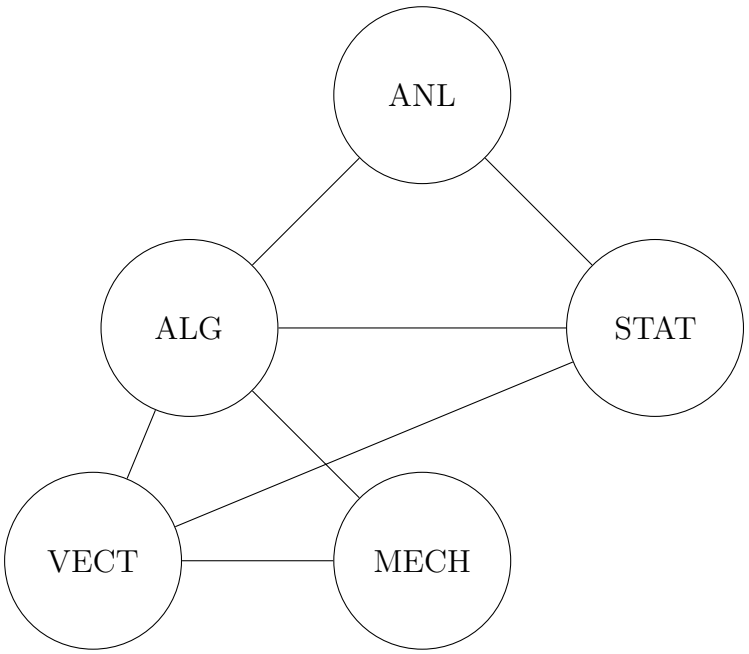
Estudos com medidas repetidas geralmente apresentam uma matriz de correlação positiva (DIGGLE et al., 2002). Pode-se interpretar a rede construída através da associação

Tabela 1 – Tabela com os coeficientes de dispersão estimados para o conjunto de dados `marks`. * indica que o coeficiente é significativo ao nível de 5%. ** indica que o coeficiente é significativo na correção de Bonferroni

Coeficiente	Estimativa	Erro padrão	Estatística Z	Relação
$\hat{\tau}_{10}$	0.00644	0.00032	20.28**	
$\hat{\tau}_{11}$	-0.00215	0.00040	−5.34**	MECH–VECT
$\hat{\tau}_{12}$	-0.00124	0.00041	−3.03**	MECH–ALG
$\hat{\tau}_{13}$	-0.00027	0.00041	-0.66	MECH–ANL
$\hat{\tau}_{14}$	-0.00055	0.00042	-1.33	MECH–STAT
$\hat{\tau}_{15}$	-0.00111	0.00041	−2.73*	VECT–ALG
$\hat{\tau}_{16}$	-0.00068	0.00041	-1.67	VECT–ANL
$\hat{\tau}_{17}$	-0.00123	0.00041	−2.99**	VECT–STAT
$\hat{\tau}_{18}$	-0.00201	0.00040	−5.02**	ALG–ANL
$\hat{\tau}_{19}$	-0.00097	0.00041	−2.35*	ALG–STAT
$\hat{\tau}_{110}$	-0.00164	0.00041	−4.01**	STAT–ANL

Fonte: Elaborado pelo autor.

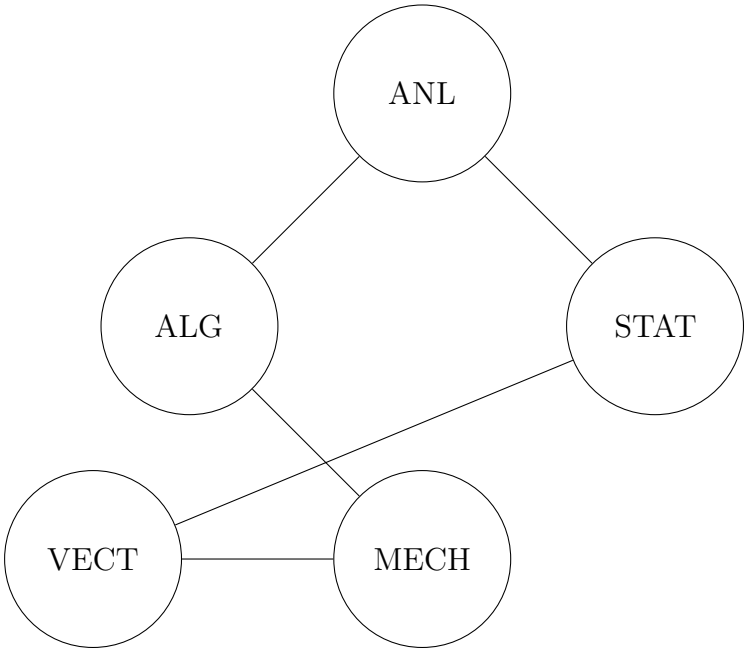
Figura 9 – Rede estimada pelo McGLM através da modelagem da estrutura de covariância



Fonte: Elaborado pelo autor.

entre as variáveis, por exemplo, um aluno com uma alta nota em Análise tende a ter uma nota alta em Álgebra.

Figura 10 – Rede estimada pelo McGLM através da modelagem da estrutura de covariância após a correção de Bonferroni



Fonte: Elaborado pelo autor.

4 Considerações Finais e Trabalhos Futuros

Neste trabalho foi proposto um esquema de estimação baseado em equações de estimação combinadas com estratégias de penalização que permite lidar com o problema de seleção de covariáveis e *high dimensional data*. O método foi implementado no *software* `R` no pacote `mcglm`. A implementação foi validada via estudos de simulação e via aplicações em dados reais.

A seleção do parâmetro de *tuning*, embora tenha sido feita via validação cruzada é um procedimento intensivo do ponto de vista computacional. Outras estratégias para a obtenção de tal valor complementam este trabalho, assim como outros métodos de penalização das estruturas de media e covariância.

Referências

BICKEL, P. J. et al. Regularization in statistics. *Test*, Springer, 2006. Disponível em: <https://link.springer.com/article/10.1007/BF02607055>.

BONAT, W. H. Multiple response variables regression models in r: The mcglm package. *Journal of Statistical Software*, v. 84, n. 4, p. 1–30, 2018. Disponível em: <http://138.232.16.156/index.php/jss/article/view/v084i04>.

BONAT, W. H. et al. Extended poisson–tweedie: Properties and regression models for count data. *Statistical Modelling*, v. 18, n. 1, p. 24–49, 2018. Disponível em: <https://doi.org/10.1177/1471082X17715718>.

BONAT, W. H.; JØRGENSEN, B. Multivariate covariance generalized linear models. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, v. 65, n. 5, p. 649–675, 2016. Disponível em: <https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/rssc.12145>.

BONAT, W. H. et al. Modelling multiple outcomes in repeated measures studies: Comparing aesthetic eyelid surgery techniques. *Statistical Modelling*, v. 21, n. 6, p. 564–582, 2021. Disponível em: <https://doi.org/10.1177/1471082X20943312>.

DIGGLE, P. et al. *Analysis of Longitudinal Data*. OUP Oxford, 2002. (Oxford Statistical Science Series). ISBN 9780198524847. Disponível em: <https://books.google.com.br/books?id=kKLbyWycRwcC>.

FRANK lldiko E.; FRIEDMAN, J. H. A statistical view of some chemometrics regression tools. *Technometrics*, Taylor & Francis, v. 35, n. 2, p. 109–135, 1993. Disponível em: <https://www.tandfonline.com/doi/abs/10.1080/00401706.1993.10485033>.

FRIEDMAN, J. H.; HASTIE, T.; TIBSHIRANI, R. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, v. 33, n. 1, p. 1–22, 2010. Disponível em: <https://www.jstatsoft.org/index.php/jss/article/view/v033i01>.

FU, W. J. Penalized estimating equations. *Biometrics*, v. 59, n. 1, p. 126–132, 2003. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1111/1541-0420.00015>.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009. (Springer series in statistics). ISBN 9780387848846. Disponível em: <https://books.google.com.br/books?id=eBSgoAEACAAJ>.

HOERL, A. E.; KENNARD, R. W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, Taylor & Francis, v. 12, n. 1, p. 55–67, 1970. Disponível em: <https://www.tandfonline.com/doi/abs/10.1080/00401706.1970.10488634>.

INAN, G.; ZHOU, J.; WANG, L. *PGEE: Penalized Generalized Estimating Equations in High-Dimension*. [S.l.], 2017. R package version 1.5. Disponível em: <https://CRAN.R-project.org/package=PGEE>.

JØRGENSEN, B.; KNUDSEN, S. J. Parameter orthogonality and bias adjustment for estimating functions. *Scandinavian Journal of Statistics*, v. 31, n. 1, p. 93–114, 2004. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9469.2004.00375.x>.

LIANG, K.-Y.; ZEGER, S. L. Inference Based on Estimating Functions in the Presence of Nuisance Parameters. *Statistical Science*, Institute of Mathematical Statistics, v. 10, n. 2, p. 158 – 173, 1995. Disponível em: <https://doi.org/10.1214/ss/1177010028>.

MARDIA, K.; KENT, J.; BIBBY, J. *Multivariate analysis*. London [u.a.]: Acad. Press, 1979. (Probability and mathematical statistics). ISBN 0124712509. Disponível em: <http://gso.gbv.de/DB=2.1/CMD?ACT=SRCHA&SRT=YOP&IKT=1016&TRM=ppn+02434995X&sourceid=fbw_bibsonomy>.

MARTINEZ-BENEITO, M. A. A general modelling framework for multivariate disease mapping. *Biometrika*, v. 100, n. 3, p. 539–553, 06 2013. ISSN 0006-3444. Disponível em: <https://doi.org/10.1093/biomet/ast023>.

NELDER, J. A.; WEDDERBURN, R. W. M. Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, v. 135, n. 3, p. 370–384, 1972. Disponível em: <https://rss.onlinelibrary.wiley.com/doi/abs/10.2307/2344614>.

PAULA, A. G. *Modelos de regressão com apoio computacional*. São Paulo: [s.n.], 2013. Disponível em: <https://www.ime.usp.br/~giapaula/texto_2013.pdf>.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2022. Disponível em: <https://www.R-project.org/>.

SCUTARI, M. Learning bayesian networks with the bnlearn R package. *Journal of Statistical Software*, v. 35, n. 3, p. 1–22, 2010.

SPELLMAN, P. T. et al. Comprehensive identification of cell cycle–regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Molecular Biology of the Cell*, v. 9, n. 12, p. 3273–3297, 1998. PMID: 9843569. Disponível em: <https://doi.org/10.1091/mbc.9.12.3273>.

TIBSHIRANI, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, v. 58, n. 1, p. 267–288, 1996. Disponível em: <https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.2517-6161.1996.tb02080.x>.

TIBSHIRANI, R. et al. Strong rules for discarding predictors in lasso-type problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, v. 74, n. 2, p. 245–266, 2012. Disponível em: <https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2011.01004.x>.

WEDDERBURN, R. W. M. Quasi-likelihood functions, generalized linear models, and the gauss-newton method. *Biometrika*, [Oxford University Press, Biometrika Trust], v. 61, n. 3, p. 439–447, 1974. ISSN 00063444. Disponível em: <http://www.jstor.org/stable/2334725>.

ZEVIANI, W. M. *EACS: Estatística Aplicada à Ciência do Solo*. [S.l.], 2019. Disponível em: <http://leg.ufpr.br/~walmes/pacotes/EACS>.

ZOU, H.; HASTIE, T. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, v. 67, n. 2, p. 301–320, 2005. Disponível em: <https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2005.00503.x>.

Apêndices

APÊNDICE A - Funções criadas pelo autor

```
ridge <- function(gamma, beta, intercept = FALSE) {
    output <- list()
    output$first <- -2 * gamma * abs(beta) * beta / (abs(beta) + 1e-6)
    output$second <- -2 * gamma * beta / beta
    return(output)
}

none <- function(gamma, beta) {
    output <- list(first = integer(length(beta)),
                   second = integer(length(beta)))
    return(output)
}

mc_penalization <- function(gamma, beta, penalization, list_X) {
    tamanhos <- cumsum(unlist(lapply(list(list_X), ncol)))
    output <- do.call(penalization,
                      args = list(gamma = gamma,
                                  beta = beta))
    idx <- c(1, tamanhos[-length(tamanhos)] + 1)
    output$first[idx] <- 0
    output$second[idx] <- 0
    return(output)
}

mc_penalization_cov <- function(gamma, beta, penalization, list_Z) {
    tamanhos <- cumsum(unlist(lapply(list(list_Z), length)))
    output <- do.call(penalization,
                      args = list(gamma = gamma,
                                  beta = beta))
    idx <- c(1, tamanhos[-length(tamanhos)] + 1)
    output$first[idx] <- 0
    output$second[idx] <- 0
    return(output)
}
```

APÊNDICE B - Script R para estudo de simulação

```
##------------------------------------------------------------------------------
## Pacotes necessários
require(devtools)
require(data.table)
require(ggplot2)
require(glmnet)

## Utilizado na versão:
## https://github.com/vriffel/mcglm/archive/02fe265b4fc493ccbd92cc54385e1eb5a7795073.zip
load_all("mcglm/R/")
##------------------------------------------------------------------------------

##------------------------------------------------------------------------------
## Lê o conjunto de dados
csv <- "https://raw.githubusercontent.com/walmes/EACS/master/data-raw/teca_qui.csv"
teca <- read.csv2(file = csv, dec = ".")

## Transforma os dados
teca <- transform(teca,
                  lk = log(k),
                  lca = log(ca),
                  lmg = log(mg + 0.1))

teca <- plyr::arrange(teca, loc, cam)
teca <- teca[, c(1, 2, 16, 17, 18)]
##------------------------------------------------------------------------------

                                        # Para estrutura de média
##------------------------------------------------------------------------------
## Cria estrutura de covariância
Z0 <- mc_id(teca)
Z_ns <- mc_ns(teca, id = "loc")

## Objeto com as fórmulas
form <- list(lk ~ cam,
             lca ~ cam,
             lmg ~ cam)

## Cria lista para armazenar as estimativas
est_mcglm <- vector("list", 1000)
```

```
## Faz a simulação para McGLM
k <- 1
for(i in seq(0.01, 2000, length.out = 1000)) {
    est_mcglm[[k]] <- coef(mcglm(linear_pred = form,
                                 matrix_pred = list(Z0, Z0, Z0),
                                 data = teca,
                                 penalization = list("none", "none", "ridge"),
                                 gamma = list(0, 0, i),
                                 control_algorithm = list(max_iter = 50)))$Estimate
    print(k)
    k <- k + 1
}


## Cria lista para armazenar as estimativas
est_glm <- vector("list", 1000)


## Faz a simulação para glmnet
k <- 1
for(i in seq(0.001, 60, length.out = 1000)) {
    est_glm[[k]] <- coef(glmnet(x = model.matrix(~ cam, data = teca),
                                y = teca[, "lmg"],
                                data = teca,
                                alpha = 0,
                                lambda = i))[-2]
    print(k)
    k <- k + 1
}
##------------------------------------------------------------------------------

##------------------------------------------------------------------------------
## Gráficos

## Monta data.table com as estimativas do McGLM
est_mc <- do.call(rbind, est_mcglm)
da <- data.table(est_mc)[, 1:9]
colnames(da) <- c("beta10", "beta11", "beta12",
                  "beta20", "beta21", "beta22",
                  "beta30", "beta31", "beta32")
db <- melt(da, variable.name = "Parâmetro", value.name = "Estimativa")
```

```
## Monta data.table com as estimativas do glmnet
est_glm <- do.call(rbind, est_glm)
da_glmnet <- data.table(est_glm)
colnames(da_glmnet) <- c("beta30", "beta31", "beta32")
db_glmnet <- melt(da_glmnet, variable.name = "Parâmetro", value.name = "Estimativa")

## Junta os data.table
db_glmnet$id <- "glmnet"
db_glmnet$x <- rep(seq(0.01, 60, length.out = 1000), 3)
db$id <- "mcglm"
db$x <- rep(seq(0.01, 2000, length.out = 1000), 9)
dc <- rbind(db, db_glmnet)

p <- ggplot(dc) +
    geom_line(aes(x = x,
                  y = Estimativa, group = Parâmetro, colour = Parâmetro)) +
    geom_hline(yintercept = 0, colour = "red", size = 0.5, linetype = "dashed") +
    xlab(expression(gamma)) +
    facet_wrap(~ id, scales = "free_x") +
    theme_bw() +
    theme(text = element_text(size = 15))
##-----------------------------------------------------------------------------

                                        # Para estrutura de covariância
##-----------------------------------------------------------------------------
## Lista para armazenar as estimativas
est_mcglm7 <- vector("list", 1000)

## Faz a simulação
k <- 1
for(i in seq(0.01, 2000, length.out = 1000)) {
    est_mcglm7[[k]] <- coef(mcglm(linear_pred = form,
                                  matrix_pred = list(c(Z0, Z_ns), c(Z0, Z_ns), c(Z0)),
                                  data = teca,
                                  penalization_cov = list("ridge", "none", "none"),
                                  gamma_cov = list(i, 0, 0),
                                  control_algorithm = list(max_iter = 50)))$Estimate
    print(k)
    k <- k + 1
}
```

```
## Monta data.table com as estimativas
est_mc7 <- do.call(rbind, est_mcglm7)[, 10:21]
da <- data.table(est_mc7)
colnames(da) <- c("rho12", "rho13", "rho23",
                  "tau11", "tau12", "tau13", "tau14",
                  "tau21", "tau22", "tau23", "tau24",
                  "tau31")
db <- melt(da, variable.name = "Parâmetro", value.name = "Estimativa")

## Gráfico
p2 <- ggplot(db) +
    geom_line(aes(x = rep(seq(0.01, 2000, length.out = 1000), 12),
                  y = Estimativa, group = Parâmetro, colour = Parâmetro)) +
    geom_hline(yintercept = 0, colour = "red", size = 0.5, linetype = "dashed") +
    xlab(expression(gamma)) +
    theme_bw() +
    theme(text = element_text(size = 15))
##------------------------------------------------------------------------------
```

APÊNDICE C - Análise do conjunto de dados yeastG1

```
##------------------------------------------------------------------------------
require("PGEE")
require("devtools")
require("ggplot2")
require("gridExtra")
require("stringr")
require("data.table")
require("ggfortify")
require("bnlearn")

## Utilizado na versão:
## https://github.com/vriffel/mcglm/archive/02fe265b4fc493ccbd92cc54385e1eb5a7795073.zip
load_all("mcglm/R/")
##------------------------------------------------------------------------------

##------------------------------------------------------------------------------
data(yeastG1)
str(yeastG1)
ncol(yeastG1); nrow(yeastG1)



## Descritiva
da <- as.data.table(yeastG1)
db <- da[, mean(y), by = time]

## Variáveis padronizadas
apply(da, 2, var)
apply(da, 2, mean)

## 4 observações por indivíduo, sem dados faltantes
table(da$id)

## Gráfico de perfis
ggplot(yeastG1) +
    geom_line(aes(x = time, group = id, y = y)) +
    geom_line(data = db,
              mapping = aes(x = time, y = V1),
              size = 2,
              colour = "red") +
    xlab("Tempo") + ylab("Nível") +
```

```
    theme_bw() +
    theme(text = element_text(size = 15))

ggsave("grafico_de_perfis.png", width = 25, height = 15, units = "cm")

## PCA
pca <- prcomp(da[, -c(1:3)])

autoplot(pca, data = da, colour = "y")
autoplot(pca, data = da, colour = "y", x = 1, y = 3)

aux <- data.frame(pc = 1:96, cs = cumsum(pca.ve))

ggplot(aux) +
    geom_point(aes(x = pc, y = cs)) +
    geom_line(aes(x = pc, y = cs)) +
    xlab("Componente principal") + ylab("Quantidade explicada acumulada") +
    theme_bw()
ggsave("prcomp.png", width = 25, heigh = 15, units = "cm")
##-----------------------------------------------------------------------------

##-----------------------------------------------------------------------------
## Validação cruzada

## Para ajustar modelo
list_initial <- list(regression = list(rep(1, ncol(yeastG1) - 1)),
                     power = list(0),
                     tau = list(c(1, 0, 0, 0, 0, 0, 0)),
                     rho = 0)

## Sequência de indivíduos para remover
seq_remover <- seq(1, 283, by = 28)

## Sequência de gammas
gamma_seq <- c(seq(0, 10, length.out = 15),
               seq(15, 100, by = 10),
               seq(120, 200, by = 20))

## Para salvar resultados
aux <- list(gamma = vector("list", length(gamma_seq)),
            mse = vector("list", length(gamma_seq)))
```

```
## Número de folds
k <- 10

## Para salvar os resultados nos slots
l <- 1

for(j in gamma_seq) {
    mse <- 0 ## Zera MSE
    for (i in 1:k) {
        idx1 <- seq_remover[i] ## Indexador do valor a ser removido
        idx2 <- seq_remover[i + 1] - 1 ## Indexador do valor a ser removido

        ## cat(idx1, ":", idx2, "\n", sep = "") ## Verificar indexadores

        ## Divide em treino e teste
        treino <- da[!(id %in% c(idx1:idx2)), ]
        teste <- da[id %in% c(idx1:idx2), ]

        ## Cria estrutura de covariância
        Z0 <- mc_id(treino)
        Z_ns <- mc_ns(id = "id",
                      data = treino)

        ## Ajusta McGLM para o j-ésimo gamma e i-ésima fold
        fit <- mcglm(linear_pred = c(y ~ . -id),
                     matrix_pred = list(c(Z0, Z_ns)),
                     data = treino,
                     penalization = "ridge",
                     gamma = j,
                     control_algorithm = list(tuning = 0.5,
                                              max_iter = 50),
                     control_initial = list_initial)

        ## Prediz resultado com McGLM ajustado
        betas <- coef(fit)
        betas <- betas[grepl("beta", betas[, 2]), 1]
        predito <- as.matrix(cbind(1, teste[, -c(1:2)])) %*% as.matrix(betas)

        ## Calcúla MSE para a i-ésima fold e soma com a anterior
        mse <- mse + mean((predito - teste$y)^2)
        cat("Finalizei a fold ", i, ". Estou no gamma ", j, ".\n", sep = "")
```

```
    }
    aux$mse[[l]] <- mse
    aux$gamma[[l]] <- j
    l <- l + 1
}

out <- data.frame(mse = unlist(aux$mse),
                  gamma = unlist(aux$gamma))

ggplot(out, aes(x = gamma, y = mse)) +
    geom_line() +
    geom_point() +
    geom_segment(x = 2.8571429,
                 y = 6.176272,
                 xend = 2.8571429,
                 yend = 0,
                 colour = "red") +
    xlab(expression(gamma)) + ylab("EQM") +
    theme_bw() +
    theme(text = element_text(size = 15)) +
    scale_x_discrete(limits = c(50, 100, 150, 190, 2.86))
ggsave("validacao_cruzada_media.png", width = 25, height = 15, units = "cm")

gamma_otimo <- 2.8571429

## Ajusta modelo para toda a base
Z0 <- mc_id(yeastG1)
Z_ns <- mc_ns(id = "id",
              data = yeastG1)
fit <- mcglm(linear_pred = c(y ~ . -id),
             matrix_pred = list(c(Z0, Z_ns)),
             data = yeastG1,
             penalization = "ridge",
             gamma = gamma_otimo,
             control_algorithm = list(tuning = 0.5,
                                      max_iter = 50))

summary(fit)
plot(fit)

residuos <- data.table(as.matrix(fitted(fit)),
                       as.matrix(residuals(fit, type = "pearson")))
```

```
colnames(residuos) <- c("V1", "V2")

p1 <- ggplot(residuos, aes(x = V1, y = V2)) +
    geom_point() +
    geom_smooth(se = F, method = "loess") +
    xlab("Valores Ajustados") + ylab("Resíduo de Pearson") +
    theme_bw() +
    theme(text = element_text(size = 12))

p2 <- ggplot(residuos, aes(sample = V2)) +
    stat_qq() +
    stat_qq_line() +
    ylab("Quantis amostrais") + xlab("Quantis teóricos") +
    theme_bw() +
    theme(text = element_text(size = 12))

a <- grid.arrange(p1, p2, nrow = 2)
b <- arrangeGrob(a)
ggsave("residuos.png", plot = b, width = 13, height = 15, units = "cm")
##------------------------------------------------------------------------------
```

APÊNDICE D - Análise do conjunto de dados marks

```
##------------------------------------------------------------------------------
require("PGEE")
require("devtools")
require("ggplot2")
require("gridExtra")
require("stringr")
require("data.table")
require("ggfortify")
require("bnlearn")

## Utilizado na versão:
## https://github.com/vriffel/mcglm/archive/02fe265b4fc493ccbd92cc54385e1eb5a7795073.zip
load_all("mcglm/R/")
##------------------------------------------------------------------------------

##------------------------------------------------------------------------------
data(marks)

marks$id <- 1:nrow(marks)

da <- as.data.table(melt(marks, id.vars = "id"))

da[, var(value), by = variable]
tapply(da$value, da$variable, summary)

ggplot(da) +
    geom_boxplot(aes(x = variable, y = value)) +
    xlab("Disciplina") + ylab("Nota") +
    theme_bw() +
    theme(text = element_text(size = 15))
ggsave("descritiva_marks.png", width = 12, height = 10, units = "cm")

ggplot(da) +
    geom_raster(aes(x = variable, y = id, fill = value)) +
    theme_bw()

cor(marks[, -6])
ggcorrplot::ggcorrplot(cor(marks[, -6]))

## Para ajustar modelo
```

```
list_initial <- list(regression = list(rep(1, ncol(marks) - 1)),
                     power = list(0),
                     tau = list(c(1, rep(0, 10))),
                     rho = 0)


Z0 <- mc_id(da)
Z_ns <- mc_ns(id = "id",
              data = da)


## Sequência de gammas
gamma_seq <- c(0,
               exp(1:13),
               exp(seq(13.5, 14.5, length.out = 10)),
               exp(seq(14.6, 15, by = 0.1)),
               exp(15))


## Para salvar resultados
aux <- list(gamma = vector("list", length(gamma_seq)),
            mse = vector("list", length(gamma_seq)))


## Cria estrutura de covariância
Z0 <- mc_id(da)
Z_ns <- mc_ns(id = "id",
              data = da)

l <- 1


set.seed(123)
idx <- sample(1:82, 30)
treino <- da[!(id %in% idx)]
teste <- marks[marks$id %in% idx,]
m_teste <- cor(teste[-6])
m_teste
cor(marks[,-6])

Z0 <- mc_id(treino)
Z_ns <- mc_ns(id = "id",
```

```
              data = treino)

for(j in gamma_seq) {
    ## Ajusta McGLM para o j-ésimo gamma e i-ésima fold
    fit <- mcglm(linear_pred = c(value ~ . -id),
                 matrix_pred = list(c(Z0, Z_ns)),
                 covariance = "inverse",
                 data = treino,
                 penalization_cov = "ridge",
                 gamma_cov = j,
                 control_algorithm = list(tuning = 0.2,
                                          max_iter = 100))

    ## Prediz a matriz de correlação com McGLM ajustado
    taus <- coef(fit)
    taus <- taus[grepl("tau", taus[, 2]), 1]
    m <- matrix(nc = 5, nr = 5)
    m[cbind(1:5, 1:5)] <- coef(fit)[6, 1]
    m[lower.tri(m)] <- coef(fit)[7:16, 1]
    m[upper.tri(m)] <- t(m)[upper.tri(m)]
    m <- solve(m)
    m <- cov2cor(m)
    print(m)
    print(m_teste)
    ## Calcúla MSE para a i-ésima fold e soma com a anterior
    mse <- sum((m - m_teste)^2)
    cat("Estou no gamma ", j,
        ". O MSE foi de ", mse,
        ".\n", sep = "")
    aux$gamma[[l]] <- j
    aux$mse[[l]] <- mse
    l <- l + 1
}

out <- data.frame(mse = unlist(aux$mse),
                  gamma = unlist(aux$gamma))

ggplot(out) +
    geom_line(aes(x = log(gamma), y = mse)) +
    geom_point(aes(x = log(gamma), y = mse)) +
    ylab("Soma dos EQM") + xlab(expression(log(gamma))) +
    theme(text = element_text(size = 15)) +
```

```
    theme_bw() +
    geom_segment(x = 13.94,
                 y = 0.2341,
                 xend = 13.94,
                 yend = 0,
                 colour = "red") +
    scale_x_discrete(limits = c(4, 8, 12, 13.94))
ggsave("validacao_cruzada_marks.png", width = 25, height = 15, units = "cm")


Z0 <- mc_id(da)
Z_ns <- mc_ns(id = "id",
              data = da)

fit_final <- mcglm(linear_pred = c(value ~ . -id),
                   matrix_pred = list(c(Z0, Z_ns)),
                   covariance = "inverse",
                   data = da,
                   penalization_cov = "ridge",
                   gamma_cov = out[which.min(out$mse), 2],
                   control_algorithm = list(tuning = 0.2,
                                            max_iter = 100))

summary(fit_final)

taus <- coef(fit_final)
taus <- taus[grepl("tau", taus[, 2]), 1]
m <- matrix(nc = 5, nr = 5)
m[cbind(1:5, 1:5)] <- coef(fit_final)[6, 1]
m[lower.tri(m)] <- coef(fit_final)[7:16, 1]
m[upper.tri(m)] <- t(m)[upper.tri(m)]
m <- solve(m)
m <- cov2cor(m)
print(round(m, 2))
round(cor(marks[,-6]), 2)
##------------------------------------------------------------------------------
```