

UNIVERSIDADE FEDERAL DE MINAS GERAIS – UFMG

BOOTSTRAP ESTACIONARIO EM
MODELOS ARFIMA (p,d,q)

Silma de Souza Evangelista

Belo Horizonte

Junho 2013

Silma de Souza Evangelista

BOOTSTRAP ESTACIONARIO EM
MODELOS **ARFIMA** (p, d, q)

Dissertação apresentada ao curso de Mestrado da
Universidade Federal de Minas Gerais - UFMG,
como requisito para a obtenção do título de Mestre
em Estatística.

Área de concentração: Probabilidade e Estatística

Orientadora: Prof^a Glaura da Conceição Franco

Belo Horizonte

2013

Resumo

Este estudo tem como objetivo utilizar o *bootstrap* estacionário para fazer inferência sobre o parâmetro de memória, d , em modelos ARFIMA e verificar a eficiência do mesmo na região de estacionariedade. O método *bootstrap* estacionário consiste em reamostrar um conjunto de dados utilizando-se as distribuições geométrica e uniforme. O comprimento de cada bloco que compõe a série *bootstrap* é obtido através da distribuição geométrica, com parâmetro p , e o ponto de início de cada bloco é gerado por uma uniforme discreta. Neste trabalho, a estimação do parâmetro de longa dependência é feita através de métodos semiparamétricos e de máxima verossimilhança. São construídos também intervalos de confiança *bootstrap* percentílico e de correção de vício e seu desempenho é analisado por meio do percentual de cobertura dos intervalos. Através de estudos de simulação Monte Carlo verificou-se que valores menores do parâmetro utilizado na distribuição geométrica geram estimativas de d mais próximas do valor real, especialmente, quando se utiliza o procedimento semiparamétrico. Os intervalos de confiança obtidos também estão próximos do nível nominal de 95% fixado, principalmente, quando o intervalo percentílico é utilizado. Além disto, os resultados mostram também que os intervalos de confiança percentílico apresentam coberturas mais próximas ao valor nominal fixado de 95% em relação ao intervalo BC.

Abstract

This study aims to use the stationary bootstrap to make inference about the memory parameter, d , in ARFIMA models and verify its efficiency in the region of stationarity. The method consists of using the stationary bootstrap to resample a data set using the geometric and uniform distributions. The length of each block that composes the bootstrap series is obtained through the geometric distribution and the starting point of each block is generated by a uniform distribution. In this work, the estimation of the memory parameter of ARFIMA models is performed through semiparametric and maximum likelihood methods. Bootstrap percentile and bias corrected confidence intervals are also constructed and their performances are analyzed by the coverage rate of the intervals. Monte Carlo simulation studies showed that lower values of the parameter used in the geometric distribution generate estimates of d closer to the actual value, especially when using the semiparametric procedure. Moreover, the results also show that the percentile confidence intervals have coverage rates closer to the fixed nominal value of 95% than the interval BC.

Agradecimentos

A Deus, por mais essa vitória em minha vida.

À minha orientadora, professora Glaura da Conceição Franco, pela estimosa colaboração na construção da minha dissertação e pelos conhecimentos compartilhados. Gostaria de ratificar que sua ajuda foi de grande valia para o meu aprimoramento e desenvolvimento.

A minha mãe, meu maior ídolo, por acreditar em mim e fazer tudo isso possível.

Aos meus irmãos que sempre me deram força na realização do meu sonho.

1	INTRODUÇÃO	7
2	PRELIMNARES	10
3	PROCESSOS DE MEMÓRIA LONGA	12
3.1	MODELO ARFIMA	15
3.2	ESTIMAÇÃO DO PARÂMETRO DE MEMÓRIA	17
3.2.1	ESTIMADOR DE GEWEKE E PORTER-HUDAK (MÉTODO GPH)	17
3.2.2	MÉTODO DE MÁXIMA VEROSSIMILHANÇA	19
4	TÉCNICA <i>BOOTSTRAP</i>	21
4.1	<i>BOOTSTRAP</i> ESTACIONÁRIO	22
4.2	INTERVALOS DE CONFIANÇA	24
4.2.1	INTERVALO DE CONFIANÇA <i>BOOTSTRAP</i> PERCENTÍLICO	24
4.2.2	INTERVALO DE CONFIANÇA <i>BOOTSTRAP</i> DE CORREÇÃO DE VÍCIO (BC)	24
5	SIMULAÇÃO	26
5.1	RESULTADOS DAS ESTIMATIVAS PONTUAIS	27
5.2	ESTIMATIVA INTERVALAR	31
6	APLICAÇÕES EM SÉRIES REAIS	35
6.1	NÍVEL DO RIO NILO	35
6.2	UMIDADE RELATIVA DO AR (%)	38
7	CONCLUSÃO	41
	REFERÊNCIAS BIBLIOGRÁFICAS	42
	APÊNDICE A	44

1 INTRODUÇÃO

Uma série temporal é definida como sendo um conjunto de observações medidas, sequencialmente, ao longo do tempo. É possível citar como exemplo, as cotações diárias do euro, taxas de juros, o índice anual de inflação de um país, etc. A característica mais relevante deste tipo de estudo é que as observações próximas são dependentes e o interesse está em analisar e modelar esta dependência. O estudo de séries temporais é motivado pelo interesse em investigar o mecanismo gerador das observações ao longo do tempo para descrever sua dinâmica com o objetivo principal de gerar previsões do seu comportamento futuro.

Algumas séries temporais têm como característica a longa dependência, ou seja, mesmo em observações distantes no tempo, a correlação verificada não é desprezível. Esta dependência é visualizada de duas maneiras, sendo uma por meio da função de autocorrelação que apresenta um decaimento hiperbólico e a outra através da função espectral que é ilimitada em frequências próximas de zero. As séries que apresentam esta característica são designadas por séries de memória longa ou longa dependência.

A presença desse fenômeno de memória longa em séries temporais foi primeiramente observada no estudo relacionado ao nível do rio Nilo entre os anos 622 e 1284. Em 1956, Hurst, ao analisar esses dados, constatou uma forte dependência entre as observações, mesmo para tempos bastante distantes entre si.

Hosking (1981) e Granger e Joyeux (1980) introduziram o modelo autoregressivo fracionário integrado de média móvel, denominado **ARFIMA** (p, d, q), onde este é a generalização do modelo **ARIMA** (p, d, q) de Box & Jenkins (1976) em relação ao parâmetro d , podendo naquele assumir valores não inteiros e, nesse, valores inteiros.

A partir de 1980, os modelos de longa dependência despertaram o interesse de muitos pesquisadores de variadas áreas de aplicações. Esses modelos passaram a ser utilizados, por exemplo, nos estudos de economia, física, análise de estudos climáticos, dentre outros.

Existem várias propostas na literatura para a estimação de d (ver Doukhan et al. (2003)). Neste trabalho, utilizaremos o estimador paramétrico, baseado na máxima verossimilhança (Fox e Taqqu, 1986) e o semiparamétrico baseado na equação de regressão usando a função periodograma (Geweke e Poter-Hudak, 1983). A aplicação deste modelo necessita que a série esteja na região de estacionariedade, ou seja, a série se desenvolva no tempo ao redor de uma média e variância constantes e também que os valores das autocovariâncias entre dois períodos não dependam do tempo, apenas da distância (k) que as separam.

A inferência sobre o parâmetro d do modelo **ARFIMA** (p, d, q) , baseada na distribuição assintótica pode ser problemática e pode gerar erros de estimação quanto se tem amostra de tamanho pequena. Os intervalos de confiança exatos são construídos com base em soluções analíticas muitas vezes complicadas de se obter, enquanto intervalos aproximados dependem de aproximações assintóticas nem sempre obtidas. Assim, uma forma de tentar fazer inferência sobre o parâmetro do modelo se dá por meio da aplicação da técnica do *bootstrap* (Efron, 1979). O *bootstrap* é uma ferramenta eficiente tanto na construção de intervalos, quanto para obter o erro-padrão de estimadores ou até mesmo para estimar a distribuição de probabilidade do estimador.

O método *bootstrap* é bastante geral, pois não depende da hipótese sobre a distribuição das estatísticas ou da normalidade dos dados. Em séries temporais, onde as observações apresentam, geralmente, autocorrelação significativa ao longo do tempo, os métodos *bootstrap* mais utilizados são o *bootstrap* nos resíduos e o *bootstrap* em blocos (Efron e Tibshirani, 1993). O método *bootstrap* estacionário foi introduzido por Politis e Romano (1991). Este método é similar às técnicas de *bootstrap* por blocos móveis com reposição, proposto por Künsch (1989) e Liu e Singh (1992). Como será visto, as pseudo-séries são geradas por blocos de comprimentos aleatórios, em que o comprimento de cada bloco tem uma distribuição geométrica. Em contraste, o método *bootstrap* por blocos móveis baseia-se em reamostragem de blocos de tamanho fixo.

O *bootstrap* nos modelos **ARFIMA** ainda é pouco explorado na literatura. A maioria dos trabalhos nesta área utilizam o *bootstrap* nos resíduos do modelo ajustado (Alonso et al., 2000, Franco e Reisen, 2004 e 2007), ou no periodograma (Arteche e Orbe, 2005) e poucos fazem uso do *bootstrap* em blocos, como o trabalho de Lahiri (1993).

Assim, o presente estudo tem a finalidade de utilizar o *bootstrap* estacionário para fazer inferência sobre o parâmetro d e verificar a eficiência do mesmo na região de estacionariedade do modelo **ARFIMA**. Para tanto, estudos de simulação Monte Carlo serão realizados para vários valores do parâmetro da distribuição geométrica que define o tamanho dos blocos e diferentes tamanhos de séries. Além disto, intervalos de confiança *bootstrap* serão construídos e a taxa de cobertura dos mesmos será avaliada.

Este trabalho está organizado da seguinte forma. O Capítulo 2 consiste na apresentação de alguns conceitos básicos sobre série temporal. O Capítulo 3 consiste na apresentação dos processos de memória longa. O Capítulo 4 apresenta as técnicas *bootstrap* e os intervalos de confiança utilizados. O Capítulo 5 apresenta a análise de resultados das simulações. O

Capítulo 6 consiste na aplicação do *bootstrap* estacionário a dados reais e por fim o Capítulo 7 apresenta as considerações finais.

2 PRELIMNARES

Nesta seção serão introduzidos alguns conceitos e definições básicas referentes à análise de séries temporais. De acordo com Morettin (2004), os modelos utilizados para descrever séries temporais são processos estocásticos, isto é, processos controlados por leis probabilísticas.

DEFINIÇÃO 2.1: Seja T um conjunto arbitrário. Um processo estocástico é uma família de variáveis aleatórias $\{Y_t\}_{t \in T}$, sendo que todas elas são definidas em um mesmo espaço de probabilidade (Ω, A, P) . A série temporal é obtida por meio de um processo estocástico.

Muitas das vezes, em estudo de série temporal, deseja-se verificar a existência de alguma dependência entre as observações atuais com os seus valores anteriores. As funções de autocorrelação e autocovariância são ferramentas bastante úteis para quantificar essa dependência. A autocovariância entre y_t e y_{t-k} pode ser expressa da seguinte forma:

$$\gamma(k) = \text{cov}(y_t, y_{t-k}) = E[(y_t - E(y_t))(y_{t-k} - E(y_{t-k}))]$$

Esta função satisfaz as seguintes propriedades:

- 1) $\gamma(0) > 0$
- 2) $\gamma(k) = \gamma(-k)$, ou seja, $\text{cov}(y_t, y_{t+k}) = \text{cov}(y_t, y_{t-k})$
- 3) $|\gamma(k)| \leq \gamma(0)$

A autocorrelação é, simplesmente, a autocovariância padronizada. A autocorrelação é uma ferramenta capaz de medir o comprimento e a memória de um processo, isto é, a extensão para a qual o valor tomado no tempo t depende daquele no momento $t-k$. A função de autocorrelação em relação a um processo estacionário é definida por:

$$\rho(k) = \frac{\gamma(k)}{\gamma(0)} = \frac{\text{cov}(y_t, y_{t-k})}{\sqrt{\text{var}(y_t)}\sqrt{\text{var}(y_{t-k})}}$$

É fácil perceber que $\rho(0) = 1$ e que $\rho(k)$ apresenta propriedades análogas a função de autocovariância. Ao considerar uma sequência de valores y_1, y_2, \dots, y_t a função de

autocorrelação amostral é definida em termos da função de autocovariância amostral como sendo:

$$\hat{\rho}_k = \frac{\hat{\gamma}_k}{\hat{\gamma}_0}, k = 0, 1, 2, \dots$$

onde $\hat{\gamma}_k$ é um estimador não-tendencioso da autocovariância e é dado por:

$$\hat{\gamma}_k = \frac{1}{n} \sum_{t=1}^{n-k} (y_t - \bar{y})(y_{t+k} - \bar{y}), \quad k = 0, 1, 2, \dots, n-1$$

É importante ressaltar que \bar{y} é a média amostral. Então $\hat{\rho}_k$ pode ser definido como sendo:

$$\hat{\rho}_k = \frac{\sum_{t=1}^{n-k} (y_t - \bar{y})(y_{t+k} - \bar{y})}{\sum_{t=1}^{n-k} (y_t - \bar{y})^2}, \quad k = 0, 1, 2, \dots, n-1$$

3 PROCESSOS DE MEMÓRIA LONGA

Em 1976, Box & Jenkins propuseram o modelo auto-regressivo integrado de média móvel, denominado **ARIMA** (p, d, q), para descrever o comportamento de séries caracterizadas pela memória curta, ou seja, os valores atuais são pouco correlacionados com valores do passado. A aplicação dos modelos Box & Jenkins necessitam, inicialmente, transformar séries não estacionárias em estacionárias, se necessário. A metodologia usada por Box & Jenkins sugere a descrição do comportamento da série temporal na forma de polinômios, sendo os valores p e q , respectivamente, o número de termos autoregressivos e de médias móveis dos polinômios. O valor d é um número inteiro que estabelece o número de diferenciações necessárias para tornar a série temporal estacionária.

Seja Y_t uma série temporal de tamanho n . Um dos procedimentos para tornar uma série estacionária é tomar diferenças sucessivas da série original até se obter uma série estacionária por meio do operador de diferenças $\nabla^d Y_t = (1-B)^d Y_t$, onde o operador de translação para o passado, denotado por B e definido por $B^k Y_t = Y_{t-k}$, representa a defasagem de k períodos de tempos para trás.

Um comportamento importante em estudo é a indicação de não estacionariedade da série original, ou seja, as autocorrelações amostrais apresentam um padrão de decaimento lento. Neste caso, faria sentido modelar a série, pelo menos em uma primeira tentativa, como um processo integrado de ordem um, isto é, $\nabla^1 Y_t = (1-B)^1 Y_t$. Segundo Morettin (2004), se a densidade espectral da série diferenciada tende à zero na frequência zero (não é um ruído branco), ou seja, parece ser um processo “super-diferenciado”, deve-se modelar a série por meio do processo de integração fracionária.

Granger e Joyeux (1980), juntamente com Hosking (1981), introduziram uma classe de modelos que apresentam correlação significativa entre observações distantes em um longo período do tempo, denominado **ARFIMA**, que é uma generalização do modelo **ARIMA** de Box & Jenkins. Os modelos de longa dependência tentam solucionar os problemas em que a série temporal parece ser um processo estocástico não-estacionário sendo que na realidade é um processo estacionário com memória longa.

A propriedade mais importante do modelo **ARFIMA** (p, d, q) é a característica de longa dependência que ocorre quando $d \in (0.0; 0.5)$ e curta dependência quando $d \in (-0.5; 0.0)$. Esses modelos são capazes de descrever tanto a estrutura de memória longa, quanto a estrutura de memória curta que restou na série após a diferenciação fracionária. Portanto, o

tipo de dependência é determinado pelo valor fracionário de d . Os parâmetros p e q modelam as autocorrelações em *lags* de ordens baixas, isto é, captam o comportamento de curto prazo e o parâmetro d modela a estrutura de autocorrelações de ordens altas, isto é, capta o comportamento de memória longa.

A análise e modelagem das séries temporais podem ser realizadas em duas vertentes: no domínio do tempo, utilizando a função de autocorrelação, e no domínio da frequência, utilizando a função de densidade espectral. A análise no domínio do tempo leva em consideração a evolução da série temporal que tem como objetivo medir a relação entre os eventos em unidades de tempo posterior e as suas magnitudes. A função de autocorrelação é a melhor ferramenta para avaliar essa evolução do processo por meio do tempo. A análise no domínio da frequência tem como objetivo verificar a frequência que alguns eventos ocorrem em determinados intervalos de tempo. O método empregado para estimar a função de densidade espectral é denominado de análise espectral (transformada de Fourier da função de autocovariância).

De acordo com Morettin (2008), um processo de memória longa é um processo estacionário em que a função de autocorrelação (ρ_j) decresce hiperbolicamente para zero, isto é,

$$\rho_j \sim Cj^{2d-1}, \quad j \rightarrow \infty, \quad (3.1)$$

onde $C > 0$ é uma constante e $0 < d < 0,5$. A expressão (3.1) garante que a função de autocorrelação tenha um decaimento lento.

A propriedade de memória longa ocorre em séries que apresentam persistência nas autocorrelações amostrais, ou seja, dependência significativa entre os valores observados separados por longo intervalo de tempo. A longa dependência pode ser definida, no domínio do tempo, como a característica na qual a função de autocorrelação não é absolutamente convergente. Formalmente, dizemos que a série tem memória longa se:

$$\lim_{n \rightarrow \infty} \sum_{j=-n}^n |\rho_j| = \infty$$

Já no domínio da frequência, a característica de longa dependência é evidenciada pelo fato da função de densidade espectral ser ilimitada nas frequências próximas do zero. Se Y_t é

um processo estacionário discreto, define-se o espectro de Y_t como sendo a transformada de Fourier da função de autocovariância:

$$f(\omega) = \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} \gamma_k e^{i\omega k}, \quad \omega \in [-\pi, \pi]$$

onde $\gamma_k = \text{cov}(Y_t, Y_{t+k})$, $e^{i\omega k} = \cos\omega k + i \text{sen}\omega k$ e ω é a frequência de Fourier. A função de densidade espectral pode ser escrita também como:

$$f(\omega) = \frac{1}{2\pi} \left[\gamma_0 + 2 \sum_{k=1}^{\infty} \gamma_k \cos(\omega k) \right]$$

onde são consideradas as propriedades $\gamma_{-k} = \gamma_k$, $\text{sen}(-\omega k) = -\text{sen}(\omega k)$ e $\cos(-\omega k) = \cos(\omega k)$

A função de autocovariância pode ser obtida através da função de densidade espectral por meio da seguinte relação:

$$\gamma_k = \int_0^{2\pi} e^{i\omega k} f(\omega) d\omega$$

Devido ao fato que a função espectral e a função autocovariância estão relacionadas, a análise baseada no domínio da frequência é equivalente no domínio do tempo.

Propriedades de $f(\omega)$:

- $f(\omega)$ é uma função contínua real,
- $f(\omega) = -f(\omega)$, para todo ω ,
- $f(\omega) \geq 0$, para todo ω .

A seguir é dado o estimador da função espectral, denominado de função periodograma. Seja um conjunto de n observações Y_1, Y_2, \dots, Y_n de um processo $\{Y_t\}$. A função $I(\omega)$, definida como periodograma, é definida para todo $\omega \in [-\pi, \pi]$ por:

$$I(\omega) = 2[\hat{\gamma}_0 + 2\sum_{k=1}^{n-1} \hat{\gamma}_k \cos(\omega k)]$$

Vale lembrar que $\hat{\gamma}_k$ é um estimador da função de autocovariância e $\hat{\gamma}_0$ é a variância amostral dada por:

$$\hat{\gamma}_0 = \frac{1}{n} \sum_{i=1}^n (Y_i - \bar{Y})^2 .$$

O estimador $I^*(\omega) = \frac{I(\omega)}{4\pi}$ é um estimador não viciado da função $f(\omega)$.

3.1 MODELO ARFIMA

A seguir é apresentado o modelo **ARFIMA** e suas características. A série $\{Y_t\}_{t \in \mathbb{Z}}$, segue o modelo **ARFIMA** (p, d, q) , proposto por Granger e Joyeux (1980), juntamente com Hosking (1981), se satisfaz:

$$\phi(B)(1-B)^d Y_t = \theta(B)u_t, \quad u_t \sim \text{Ruído Branco}(0, \sigma_u^2)$$

onde $\phi(B) = 1 - \phi_1 B^1 - \phi_2 B^2 - \dots - \phi_p B^p$ e $\theta(B) = 1 - \theta_1 B^1 - \theta_2 B^2 - \dots - \theta_q B^q$ são polinômios de graus p e q , respectivamente, e u_t é uma sequência de variáveis aleatórias independentes e identicamente distribuídas com média zero e variância finita, ou seja, um ruído branco.

O termo $(1-B)^d$ é o operador de diferença fracionária e é definido pela expansão binomial:

$$(1-B)^d = \sum_{j=1}^{\infty} \binom{d}{j} (-1)^j B^j .$$

Expandindo o operador de diferença fracionária, temos:

$$(1-B)^d = 1 - dB + \frac{d(d-1)}{2!} B^2 - \frac{d(d-1)(d-2)}{3!} B^3 + \dots$$

Segundo Morettin (2004), uma das suposições mais frequentes que se faz a respeito de uma série temporal é a de que ela é estacionária, ou seja, ela se desenvolve no tempo aleatoriamente ao redor de uma média constante, refletindo alguma forma de equilíbrio estável.

Hosking (1981) demonstrou que se $\{Y_t\}_{t \in \mathbb{Z}}$ é um processo **ARFIMA** (p, d, q) , então as condições de estacionariedade e invertibilidade são:

- i. Y_t é estacionário se $d < 0,5$ e todas as raízes da equação $\phi(B) = 0$ estiverem fora do círculo unitário.
- ii. Y_t é invertível se $d > -0,5$ e todas as raízes da equação $\theta(B) = 0$ estiverem fora do círculo unitário.

Para que a série Y_t seja estacionária e invertível é necessário, então que $d \in (-0,5, 0,5)$.

Segundo Hosking (1981), se Y_t for estacionário e invertível e se $f(\omega)$ representa a função densidade espectral, então:

- i. $\lim_{\omega \rightarrow 0} \omega^{2d} f(\omega)$ existe e é finito;
- ii. $\lim_{j \rightarrow \infty} j^{1-2d} \rho_j$ existe e é finito.

Ao considerar um caso especial quando $p = q = 0$, tem-se o modelo denominado ruído fracionário, **ARFIMA** $(0, d, 0)$ e é representado pelo modelo:

$$(1 - B)^d Y_t = u_t \quad -0,5 < d < 0,5$$

Segundo Morettin (2008), "a razão da escolha do modelo **ARFIMA** para fins de modelagem das séries com comportamento de longa dependência é que o efeito do parâmetro d em observações distantes decai hiperbolicamente conforme a distância aumenta, enquanto os efeitos dos parâmetros de médias móveis (θ) e autorregressivo (ϕ) decaem exponencialmente. Então, d deve ser escolhido com o objetivo de explicar a estrutura de

correlação de ordens altas da série, enquanto os parâmetros θ e ϕ explicam a estrutura de correlação de ordens baixas".

3.2 ESTIMAÇÃO DO PARÂMETRO DE MEMÓRIA

Existem inúmeros métodos na literatura para a estimação do parâmetro de longa dependência, entretanto neste trabalho o enfoque será dado apenas nos seguintes procedimentos:

1. O método de regressão utilizando o periodograma proposto por Geweke e Porter-Hudak (1983);
2. O método de aproximação da função de máxima verossimilhança proposto por Fox e Taqqu (1986).

Segue abaixo uma descrição detalhada destes métodos.

3.2.1 ESTIMADOR DE GEWEKE E PORTER-HUDAK (MÉTODO GPH)

Nesta seção apresentaremos a estimação semiparamétrica no domínio da frequência. Para esse processo, inicialmente, estima-se o parâmetro de diferenciação, d . Os demais parâmetros (auto-regressivos e médias móveis) são estimados no passo seguinte. Esse método foi proposto por Geweke e Porter-Hudak (1983) e se baseia na equação que exhibe relação entre a função densidade espectral de um processo **ARFIMA** (p, d, q) e de um processo **ARMA** (p, q) . O procedimento é detalhado a seguir:

Seja $\{Y_t\}_{t \in \mathbb{Z}}$ um processo estacionário **ARFIMA** (p, d, q) com $d \in (-0.5; 0.5)$. A função densidade espectral do processo é dada por:

$$f_y(\omega) = f_u(\omega) [2 \operatorname{sen}\left(\frac{\omega}{2}\right)]^{-2d} \quad (3.2.1)$$

onde $f_u(\cdot)$ denota a função densidade espectral do processo **ARMA** (p, q) , $\phi(B)U_t = \theta(B)\varepsilon_t$, para todo $t \in \mathbb{Z}$, dada por

$$f_u(\omega) = \frac{\sigma_\varepsilon^2 |\theta(e^{-i\omega})|^2}{2\pi |\phi(e^{-i\omega})|^2} \frac{\sigma_\varepsilon^2}{2\pi}.$$

Logo, temos que a função densidade espectral do processo **ARFIMA** (p, d, q) pode ser escrita como:

$$f_y(\omega) = \frac{\sigma_\varepsilon^2 |\theta(e^{-i\omega})|^2}{2\pi |\phi(e^{-i\omega})|} \left| 2\text{sen}\left(\frac{\omega}{2}\right) \right|^{-2d}.$$

Tomando logaritmo da expressão (3.2.1), temos:

$$\ln f_y(\omega) = \ln f_u(\omega) - d \ln [2\text{sen}\left(\frac{\omega}{2}\right)]^2 \quad (3.2.2)$$

Somando a ambos os lados da expressão (3.2.2) o termo $\ln f_u(0)$ e com alguma álgebra temos a equação:

$$\ln f_y(\omega) = \ln f_u(0) - d \ln [2\text{sen}\left(\frac{\omega}{2}\right)]^2 + \ln \left\{ \frac{f_u(\omega)}{f_u(0)} \right\}$$

Substituindo ω pelas frequências de Fourier $\omega_j = 2\pi j/n$, $j = 0, 1, \dots, n/2$, onde n é tamanho da amostra e adicionando $\ln I(\omega_j)$ em ambos os lados da expressão, onde $I(\omega_j)$ é a função periodograma, temos:

$$\ln I(\omega_j) = \ln f_u(0) - d \ln [2\text{sen}\left(\frac{\omega_j}{2}\right)]^2 + \ln \left\{ \frac{f_u(\omega_j)}{f_u(0)} \right\} + \ln \left\{ \frac{I(\omega_j)}{f_y(\omega_j)} \right\} \quad (3.2.3)$$

O valor máximo de j , ou seja, $j = g(n)$ é escolhido de modo $g(n)/n \rightarrow 0$ quando $n \rightarrow 0$ e $\omega_j \leq \omega_{g(n)}$, onde $\omega_{g(n)}$ é pequeno. O termo $\ln \left\{ \frac{f_u(\omega_j)}{f_u(0)} \right\}$ é considerado desprezível quando se considera as frequências próximas de zero, que serão consideradas para a estimação de d .

Assim, obtemos uma forma aproximada para a equação (3.2.3), dada por:

$$\ln I(\omega_j) \approx \ln f_u(0) - d \ln [2\text{sen}\left(\frac{\omega_j}{2}\right)]^2 + \ln \left\{ \frac{I(\omega_j)}{f_y(\omega_j)} \right\}. \quad (3.2.4)$$

A equação (3.2.4) pode ser expressa como uma equação de regressão da seguinte forma:

$$y_j = a + bx_j + \varepsilon_j, \text{ para todo } 1 \leq j \leq g(n)$$

onde: $y_j = \ln I(\omega_j)$, $x_j = \ln[2\text{sen}\left(\frac{\omega_j}{2}\right)]^2$, $\varepsilon_j = \ln\left\{\frac{I(\omega_j)}{f_y(\omega_j)}\right\}$, $a = \ln f_u(0)$ e $b = -d$

A estimação de d , proposto por Geweke e Porter-Hudak (1983), é obtida utilizando o método de mínimos quadrados, denominado aqui por \hat{d}_{GPH} , é dado por:

$$\hat{d}_{GPH} = \frac{\sum_{j=1}^{g(n)} (x_j - \bar{x}) y_j}{\sum_{j=1}^{g(n)} (x_j - \bar{x})^2}$$

onde \bar{x} corresponde a média de x_j .

Geweke e Porter-Hudak (1983) demonstraram que:

$$d_{GPH} \xrightarrow{D} N\left(d, \frac{\pi^2}{6 \sum_{j=1}^{g(n)} (x_j - \bar{x})^2}\right)$$

em que $g(n) = n^\alpha$ ($0 \leq \alpha \leq 1$).

3.2.2 MÉTODO DE MÁXIMA VEROSSIMILHANÇA

Nesta seção apresentaremos a estimação paramétrica do parâmetro d . O estimador de máxima verossimilhança, proposto por Fox e Taquq (1986), é baseado em uma aproximação da função de verossimilhança sugerida por Whittle (1953). Vale a pena ressaltar que nesse método todos os parâmetros são estimados conjuntamente.

Suponha $\{Y_t\}_{t \in \mathbb{Z}}$ um processo Gaussiano, logo a função de verossimilhança de Y_t proveniente do processo **ARFIMA** (p, d, q) é dada por:

$$L(y, \eta) = (2\pi)^{-\frac{n}{2}} |\Sigma_n(\eta)|^{-\frac{1}{2}} \exp^{-\frac{1}{2} y' [\Sigma_n(\eta)]^{-1} y}$$

onde $\eta = (d, \phi_1, \dots, \phi_p, \theta_1, \dots, \theta_q, \sigma^2)$ é o vetor de parâmetros desconhecidos do modelo, $y = (y_1, \dots, y_n) \in \mathfrak{R}^n$, y' denota o vetor transposto do vetor y . O termo $\Sigma_n(\eta)$ representa a

matriz quadrada $n \times n$, sendo que $\Sigma_n(\eta) = [\gamma(k)]_{k=0}^n$. Vale à pena lembrar que $\gamma(k)$ corresponde à função de autocovariância do processo.

A função de verossimilhança exata necessita do cálculo da matriz inversa da autocovariância e por isso a aproximação da função de verossimilhança sugerida por Whittle é preferida. Computacionalmente, a maximização da função exata apresenta um elevado custo no tempo.

Em 1986, Fox e Taquq fizeram uso do método máxima verossimilhança aproximada, proposto por Whittle (1953), para estimar o vetor de parâmetros desconhecidos. Este método consiste em substituir a matriz $\Sigma_n^{-1}(\eta)$, que não é fácil de ser calculada, pela matriz aproximada $A_n(\eta)$ cujos elementos são fáceis de calcular. Então, ao fazer o uso desta aproximação, a estimação de η é obtida maximizando a seguinte função de máxima verossimilhança:

$$L(y, \eta) = \left(\frac{1}{\sqrt{2\pi\sigma}} \right)^n \exp \left\{ -\frac{z' A_n(\eta) z}{2n\sigma} \right\}$$

onde $z = (y_1 - \bar{y}, \dots, y_n - \bar{y})$, \bar{y} é a média amostral e $A_n(\eta) = [\alpha(k)]_{k=1}^n$ é uma matriz $n \times n$ sugerido por Whittle (1953) com a finalidade de aproximar a função de covariância $\Sigma_n(\eta)$. Segundo Whittle (1953) a matriz $\Sigma_n(\eta)$ pode ser aproximada por:

$$\alpha(k) = \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} \frac{1}{f(\omega, \eta)} e^{i\omega k} d\omega$$

onde $f(\omega, \eta)$ corresponde a função densidade espectral caracterizada pelo vetor de parâmetro desconhecido. Assim, o estimador de máxima verossimilhança é determinado por meio da minimização da função de Whittle que é dada por:

$$FT(\eta) = \frac{1}{2n} \sum_{j=1}^{n-1} \ln f(\omega_j; \eta) + \frac{I(\omega_j)}{f(\omega_j; \eta)}.$$

Ver Fox e Taquq (1986) e Beran (1994), para um estudo mais completo esse estimador.

4 TÉCNICA *BOOTSTRAP*

O método *bootstrap*, introduzido por Efron (1979), é uma ferramenta poderosa de reamostragem que pode ser empregada para aproximar a distribuição teórica pela distribuição empírica de uma amostra finita de observações. Esse método se baseia na construção de distribuições amostrais por reamostragem, e é muito utilizado para estimar intervalo de confiança para os parâmetros, construir intervalo de predição, viés e a variância dos estimadores, entre outras aplicações.

A técnica *bootstrap* consiste de um sorteio com reposição das observações de uma amostra, gerando “pseudo-séries”, de tamanho igual à original. A partir dessas “pseudo-séries”, é possível estimar características da população, tais como a média, variância, etc.

O *bootstrap* em séries temporais requer algumas modificações, devido ao fato que as observações são correlacionadas. Existem basicamente duas formas de aplicação do *bootstrap* nestes casos: o *bootstrap* de blocos móveis, Künsh (1989) e Liu e Singh (1992), e o *bootstrap* nos resíduos do modelo ajustado (Franco e Reisen, 2004).

A técnica não-paramétrica do *bootstrap* por blocos móveis consiste em reproduzir blocos, de comprimento q , que sejam independentes, mas que preservem a dependência dos dados de um bloco. Sorteiam-se k blocos amostrados com reposição, agregando-os para formar a pseudo-série. Esse processo é repetido por B vezes, gerando B “pseudo-séries”. A escolha do tamanho do bloco (q) é problema muito discutido na literatura (Hall *et al.*, 1995), pois um comprimento pequeno produz amostras que não capturam corretamente a dependência das observações. Já a escolha de um comprimento grande acarreta na diminuição da eficiência das estatísticas.

Em relação à técnica *bootstrap* residual, é necessário garantir a hipótese de independência dos resíduos, que são usados para gerar as “pseudo-séries”. Inicialmente ajusta-se um modelo para a série em questão e calcula-se os resíduos do modelo ajustado. O *bootstrap* é realizado nestes resíduos e, desta forma, são geradas B pseudo-séries utilizando os parâmetros do modelo original e os resíduos *bootstrap*. Este procedimento é dito ser dependente do modelo, pois as séries *bootstrap* são obtidas utilizando-se os parâmetros estimados para o modelo. Para maiores detalhes, ver Franco e Reisen (2004).

Neste trabalho utilizaremos um aperfeiçoamento na técnica de blocos móveis, em que o tamanho do bloco não é fixado *a priori*. O procedimento é baseado no trabalho de Politis e Romano (1991), descrito na próxima seção.

4.1 *BOOTSTRAP* ESTACIONÁRIO

Um inconveniente que pode ocorrer com o *bootstrap* por bloco é que a série temporal resultante não é estacionária. Politis e Romano (1991) propuseram o *bootstrap* estacionário para superar este problema.

Politis e Romano (1991) propuseram um procedimento similar ao *bootstrap* em blocos, que consiste em sortear com reposição dados pertencentes a uma amostra retirada anteriormente, de modo a formar uma pseudo-série.

Considere $Y = (Y_1, Y_2, \dots, Y_{n-1}, Y_n)$ a amostra aleatória disponível da variável aleatória, de tamanho n (finita), com função de distribuição desconhecida descrita por F e $\theta = S(Y)$ a estatística de interesse.

A idéia do *bootstrap* estacionário é reamostrar os dados originais em vários blocos, onde cada bloco é formado por um número aleatório de observações consecutivas, denotado pelo índice L_i , e o índice I_i representa a posição que em o bloco iniciará. Em cada bloco há preservação da estrutura de dependência das observações originais. A equação (3.1) ilustra como se dá a construção dos blocos,

$$B_{I_i, L_i} = \{Y_{I_i}, Y_{I_i+1}, \dots, Y_{I_i+L_i-1}\} \quad (4.1)$$

Ambos os índices L_i e I_i são variáveis aleatórias com distribuição geométrica e uniforme discreta, respectivamente. O tamanho do bloco não é constante. A fim de alcançar a estacionariedade para a série de tempo reamostrada, os dados originais são estruturados em forma de “círculo”, de modo que o final da série seja conectado ao início da mesma. Caso $k = I_i + L_i - 1 > n$, denota-se $Y_{n+k} = Y_k$. Vale destacar que k representa o valor da posição no qual a série iniciará, adicionada ao comprimento do bloco.

A Figura 4.1 ilustra a construção dos blocos.

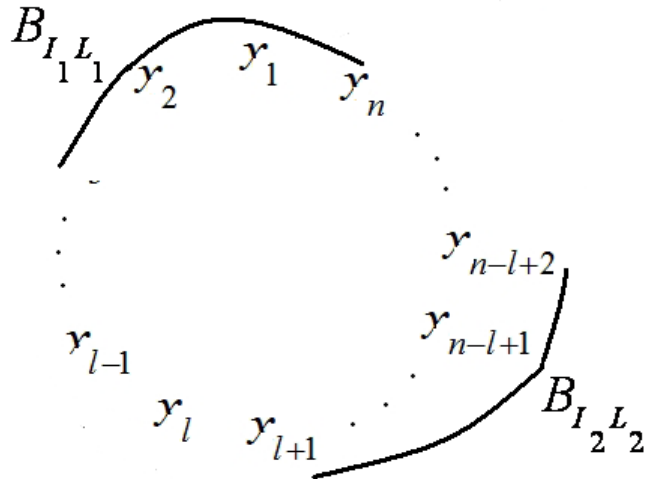


Figura 4.1: Ilustração da construção de blocos

A pseudo-série consiste na junção de vários blocos de comprimentos aleatórios, isto é, $Y^* = B_{I_1, L_1}, B_{I_2, L_2}, \dots$. As primeiras L_1 observações da pseudo-série são determinadas pelo bloco B_{I_1, L_1} de observações $Y_{I_1}, Y_{I_1+1}, \dots, Y_{I_1+L_1-1}$, as próximas L_2 observações são obtidas pelo segundo bloco, B_{I_2, L_2} . Esse mecanismo é interrompido quando o comprimento da pseudo-série atinge o tamanho da original. Caso o número de observações da pseudo-série ultrapasse o tamanho da inicial, o último bloco reamostrado é cortado até obter a série *bootstrap* do tamanho da inicial. Para cada pseudo-série tem-se a correspondente estimativa *bootstrap* da estatística de interesse, ou seja, $\theta^* = S(Y^*)$. Replica-se esse mecanismo B número de vezes e encontrará a distribuição empírica do estimador, que é facilmente visualizada por um histograma.

Na construção do *bootstrap* estacionário é importante escolher o parâmetro da distribuição geométrica (p) tendendo a zero, segundo Politis e Romano (1991). Para que o erro médio quadrático da variância seja minimizado é necessário que

$$p \propto n^{-\frac{1}{3}}$$

sendo que n corresponde ao tamanho da série original. Politis e Romano (1991) verificaram que o *bootstrap* estacionário é menos sensível à má especificação do tamanho dos blocos quando comparado ao *bootstrap* em blocos com repetição e ao *bootstrap* circular em blocos.

4.2 INTERVALOS DE CONFIANÇA

Nesta subseção serão apresentadas duas formas para se construir intervalos de confiança *bootstrap* para o parâmetro de memória d do modelo **ARFIMA**, o intervalo percentílico e o intervalo de correção de vício.

4.2.1 INTERVALO DE CONFIANÇA *BOOTSTRAP* PERCENTÍLICO

Uma forma usual de se obter intervalos de confiança de um estimador, $\theta^* = S(Y^*)$, é por meio do método *bootstrap* percentílico, onde este estabelece os intervalos de confiança com os percentis $(\alpha/2)$ e $(1-\alpha/2)$ da distribuição empírica do estimador. Esse método foi proposto por Efron & Tibshirani (1993). Na prática, são geradas B pseudo-séries independentes, $Y_1^*, Y_2^*, \dots, Y_B^*$, e depois se estima a estatística de interesse, $\theta^* = S(Y^*)$, para cada pseudo-série. Em seguida, esses valores estimados são ordenados e toma-se o percentil $100(\alpha/2)$ como o limite inferior e o percentil $100(1-\alpha/2)$ como o limite superior do intervalo. Pode-se definir o intervalo como:

$$IC_{(100-\alpha)\%} : [\hat{\theta}_{100,(\alpha/2)}^* ; \hat{\theta}_{100,(1-\alpha/2)}^*].$$

4.2.2 INTERVALO DE CONFIANÇA *BOOTSTRAP* DE CORREÇÃO DE VÍCIO (BC)

Na construção do intervalo de confiança BC o limite superior e inferior do intervalo são os percentis da distribuição *bootstrap* ajustados para corrigir o vício de estimação e assimetria da distribuição. Para determinar o intervalo de confiança BC, por exemplo, é necessário calcular o percentil da distribuição, mas não, exatamente, os percentis tradicionais de 2,5% e 97,5%. Esse método pretende corrigir estes valores para possíveis vícios na estimação do parâmetro.

O método consiste em calcular a probabilidade (p_o) de uma estimativa *bootstrap* ser inferior à estimativa da estatística da original, ou seja,

$$p_o = P[\hat{\theta}_i^* \leq \hat{\theta}].$$

Por meio do valor da probabilidade (p_o) é encontrado o parâmetro responsável pela correção do vício z_o que representa a inversa da normal no ponto p_o

$$z_o = \Phi^{-1}(p_o).$$

Por fim, seleciona-se o nível de confiança $(1 - \alpha)100\%$ para determinar o valor de $z_{\frac{\alpha}{2}}$.

$$P_{inferior} = \Phi\left(2z_o + z_{\frac{\alpha}{2}}\right)$$

$$P_{superior} = \Phi\left(2z_o + z_{\frac{(1-\alpha)}{2}}\right).$$

O intervalo corrigido pelo vício (BC) é definido como:

$$IC_{BC} = \left[\hat{\theta}^{*(P_{inferior})} , \hat{\theta}^{*(P_{superior})} \right].$$

5 SIMULAÇÃO

Por meio do método Monte Carlo foram simuladas inicialmente 1000 séries do modelo **ARFIMA** (p, d, q) , com tamanhos $n = 100$ e 500 . O *software* estatístico R versão (2.15.1) foi o escolhido para a construção da linguagem de programação. Para a geração das séries usou-se o pacote **fracdiff.sim**. A estimação de d através do método semiparamétrico GPH foi feita utilizando-se a função **fdGPH** e a estimação por máxima verossimilhança foi feita utilizando-se a função **fracdiff**.

As estatísticas calculadas para avaliar o desempenho dos estimadores dos métodos *bootstrap* foram a média das estimativas e o Erro Quadrático Médio (EQM). Os intervalos de confiança (IC) foram comparados por meio do Percentual de Cobertura (PC), em relação ao nível de 95%.

O processo para a obtenção das séries *bootstrap* e suas respectivas estimativas é dado pelo seguinte esquema:

1. Inicialmente, foram geradas séries originais de tamanho $n = 100$ e 500 .
2. Para cada uma das séries geradas, foram geradas 1000 séries *bootstrap*, baseadas no método *bootstrap* estacionário.
3. O método *bootstrap* estacionário, como visto na Seção 3, consiste na reamostragem da série original em inúmeros blocos aleatórios, sendo que o início da série é determinado pela distribuição uniforme discreta e o comprimento pela distribuição geométrica.
4. A distribuição uniforme discreta utilizada foi: $U \sim (1, n)$, pois são as possíveis alternativas para o início da série. Em relação à distribuição geométrica foram utilizados $p = 0,5, 0,05, 0,005$ e $0,0005$.
5. Para cada uma das séries *bootstrap* geradas, foi estimado o parâmetro d via GPH e MV (máxima verossimilhança).
6. Foram construídos intervalos de confiança percentílico e *BC* para o parâmetro em estudo.
7. Este procedimento foi repetido 1000 vezes e registrou-se a estimativa média do parâmetro; o EQM médio; o número de intervalos de confiança que, em cada situação, continham o parâmetro de interesse. Neste caso, o percentual de cobertura nominal é determinada pelo quociente entre o número de intervalos que continham o verdadeiro valor do parâmetro e o número total de intervalos determinados.

Para testar a metodologia proposta, foram realizadas simulações de Monte Carlo com $d = 0,3$ e $0,45$ para os seguintes modelos: **ARFIMA (0, d , 0)**, **ARFIMA (1, d , 0)** com $\phi = 0,4$ e **ARFIMA (0, d , 1)** com $\theta = 0,4$.

5.1 RESULTADOS DAS ESTIMATIVAS PONTUAIS

Nesta seção serão apresentados os resultados das estimativas pontuais para o parâmetro d . A Tabela 1 mostra os resultados da estimação de d para o modelo **ARFIMA (0, d , 0)**, com $n = 100$ e 500 e $d = 0,3$ e $0,45$. Vale ressaltar que a série *bootstrap* é formada por vários blocos da série original e o comprimento de cada bloco é determinado pela distribuição geométrica de parâmetro (p).

A Tabela 1 apresenta a média e o EQM das estimativas pontuais de d em relação aos dois tipos de estimadores: GPH e MV. Os valores em negrito representam as melhores estimativas para o parâmetro obtidas no procedimento *bootstrap*.

Em relação aos resultados referentes à estimação do parâmetro d para a série original, observa-se que o método GPH apresenta estimativas menos viciadas, mas o método MV possui menor EQM. Este é um resultado já conhecido na literatura (ver Franco e Reisen, 2007). Os resultados são consistentes uma vez que à medida que o tamanho da série aumenta a variabilidade do estimador diminui, ou seja, as observações vão ficando cada vez mais concentradas em torno do parâmetro na medida em que a série vai tornando cada vez maior.

Quanto ao procedimento *bootstrap*, pode-se perceber que à medida que o valor de p diminua, os resultados obtidos ficam mais próximos dos valores encontrados pelo método de Monte Carlo para a série original. Apesar de que esses resultados referentes ao *bootstrap* sempre apresentaram valores abaixo do valor da série original. A seguir é feita uma análise mais detalhada dos resultados para cada valor de p .

Se o comprimento de cada bloco da série *bootstrap* tem distribuição geométrica com parâmetro igual a $0,5$, o valor médio de d , em ambos tamanhos de amostra e métodos, ficaram muito subestimado.

Para o caso em que o comprimento de cada bloco da série *bootstrap* tenha distribuição geométrica com $p = 0,05$, o método de máxima verossimilhança parece ser mais preciso (erro quadrático médio pequeno) do que o método GPH quando analisado $n = 500$.

Se o tamanho de cada bloco segue uma distribuição geométrica com $p = 0,005$, observa-se que para a série com $n = 100$ o melhor método de estimação foi o GPH e para $n = 500$ foi o de máxima verossimilhança.

Se o comprimento de cada bloco tem distribuição geométrica com $p = 0,0005$, nota-se que tanto para $n = 100$ e 500 o melhor método de estimação foi desempenhado pelo GPH.

É importante destacar que o comprimento dos blocos para a construção da série *bootstrap* vai aumentando à medida que p diminui. Assim, é necessário muita cautela para se decidir qual o melhor valor de p , já que um valor de p muito pequeno (por exemplo, $p = 0,0005$) pode estar simplesmente reproduzindo a série original. Neste caso a variabilidade inerente à distribuição subjacente do estimador de d utilizado não estaria sendo considerada. Isto invalidaria o procedimento *bootstrap* para a realização de inferências com relação ao parâmetro de interesse, como a construção de intervalos de confiança e testes de hipóteses, como será visto na próxima subseção.

TABELA 1
Estimação pontual do parâmetro d em relação ao modelo **ARFIMA (0, d , 0)**

	d = 0,3				d = 0,45			
	n = 100		n = 500		n = 100		n = 500	
	Original	Bootstrap	Original	Bootstrap	Original	Bootstrap	Original	Bootstrap
$p = 0,5$								
GPH								
\hat{d}	0,313	0,021	0,307	0,005	0,469	0,033	0,463	0,008
EQM	0,082	0,164	0,029	0,116	0,086	0,261	0,029	0,224
MV								
\hat{d}	0,248	0,091	0,292	0,128	0,376	0,156	0,434	0,220
EQM	0,010	0,050	0,001	0,031	0,010	0,095	0,001	0,055
$p = 0,05$								
GPH								
\hat{d}	0,291	0,175	0,308	0,166	0,450	0,299	0,468	0,268
EQM	0,099	0,125	0,028	0,052	0,099	0,135	0,028	0,069
MV								
\hat{d}	0,243	0,209	0,287	0,273	0,374	0,336	0,429	0,424
EQM	0,010	0,019	0,001	0,004	0,011	0,022	0,001	0,003
$p = 0,005$								
GPH								
\hat{d}	0,311	0,282	0,297	0,254	0,463	0,430	0,455	0,405
EQM	0,080	0,092	0,027	0,040	0,081	0,094	0,028	0,041
MV								
\hat{d}	0,241	0,233	0,290	0,287	0,369	0,361	0,433	0,431
EQM	0,010	0,013	0,001	0,002	0,011	0,014	0,001	0,002
$p = 0,0005$								
GPH								
\hat{d}	0,309	0,306	0,302	0,296	0,459	0,456	0,460	0,452
EQM	0,086	0,087	0,031	0,033	0,084	0,086	0,031	0,033
MV								
\hat{d}	0,243	0,240	0,287	0,287	0,370	0,367	0,429	0,430
EQM	0,011	0,011	0,001	0,002	0,012	0,013	0,001	0,001

A Tabela 2 mostra o desempenho dos estimadores para o modelo **ARFIMA(1,d,0)**. Ao inserir o termo **AR** no modelo, observa-se que o parâmetro d é sempre superestimado, e que o método MV apresenta piores estimativas. Este também é um resultado conhecido na literatura (ver Franco e Reisen, 2007). Deve-se observar que a rotina disponível no R para o cálculo do estimador MV possui valor máximo de 0,5, e por este motivo os resultados para $d = 0,45$ ficam comprometidos. Nota-se que com o aumento do tamanho da série analisada houve uma melhora no valor estimado de d , ou seja, os resultados são consistentes

TABELA 2

Estimação pontual do parâmetro d em relação ao modelo **ARFIMA (1, d, 0)**, com $\phi = 0,4$

	d = 0,3				d = 0,45			
	n = 100		n = 500		n = 100		n = 500	
	Original	Bootstrap	Original	Bootstrap	Original	Bootstrap	Original	Bootstrap
p = 0,5								
GPH								
\hat{d}	0,370	0,033	0,324	0,007	0,527	0,042	0,479	0,010
EQM	0,098	0,158	0,028	0,115	0,095	0,253	0,029	0,223
MV								
\hat{d}	0,455	0,208	0,492	0,257	0,482	0,256	0,497	0,319
EQM	0,025	0,018	0,037	0,004	0,001	0,046	0,002	0,019
p = 0,05								
GPH								
\hat{d}	0,379	0,244	0,315	0,173	0,537	0,360	0,471	0,273
EQM	0,089	0,111	0,032	0,050	0,094	0,118	0,033	0,067
MV								
\hat{d}	0,457	0,428	0,492	0,492	0,483	0,468	0,497	0,496
EQM	0,026	0,020	0,037	0,037	0,001	0,001	0,002	0,002
p = 0,005								
GPH								
\hat{d}	0,381	0,353	0,313	0,270	0,540	0,505	0,469	0,419
EQM	0,095	0,101	0,030	0,040	0,094	0,100	0,029	0,040
MV								
\hat{d}	0,457	0,451	0,492	0,491	0,483	0,480	0,497	0,497
EQM	0,026	0,024	0,037	0,036	0,001	0,001	0,002	0,002
p = 0,0005								
GPH								
\hat{d}	0,384	0,381	0,321	0,314	0,546	0,542	0,479	0,471
EQM	0,086	0,087	0,029	0,030	0,086	0,087	0,029	0,031
MV								
\hat{d}	0,457	0,454	0,491	0,492	0,483	0,481	0,497	0,497
EQM	0,026	0,025	0,037	0,037	0,001	0,001	0,002	0,002

Em relação ao procedimento *bootstrap*, pode-se constatar que os resultados obtidos foram similares aos resultados para o modelo **ARFIMA (0, d, 0)**. Observou-se também que à medida que o valor de p diminui os resultados ficaram mais próximos dos valores obtidos no Monte Carlo para a série original. Entretanto, esses resultados relacionados ao *bootstrap*

apresentaram valores abaixo do valor da série original. Isto fez com que os valores do *bootstrap* ficassem mais próximos dos verdadeiros valores do parâmetro ($d = 0,3$ ou $0,45$) quando p diminui.

A Tabela 3 mostra o desempenho dos estimadores para o modelo **ARFIMA (1,d,0)**. Neste caso, a inserção do termo **MA** (médias móveis) causa uma subestimação do parâmetro d . Mas ao aumentar o tamanho da série analisada houve uma melhora no valor estimado para d pelo método GPH. Vale destacar que o método de máxima verossimilhança apresentou as piores estimativas.

TABELA 3

Estimação pontual do parâmetro d em relação ao modelo **ARFIMA (1, d, 0,4)**, com $\theta = 0,4$.

	d = 0,3				d = 0,45			
	Original n = 100	Bootstrap	Original n = 500	Bootstrap	Original n = 100	Bootstrap	Original n = 500	Bootstrap
p = 0,5								
GPH								
\hat{d}	0,234	0,0003	0,296	0,002	0,397	0,015	0,452	0,006
EQM	0,093	0,176	0,030	0,118	0,084	0,276	0,030	0,226
MV								
\hat{d}	0,018	0,011	0,035	0,006	0,103	0,035	0,180	0,060
EQM	0,081	0,085	0,071	0,087	0,126	0,175	0,074	0,154
p = 0,05								
GPH								
\hat{d}	0,240	0,116	0,293	0,151	0,401	0,240	0,450	0,255
EQM	0,084	0,140	0,030	0,057	0,086	0,154	0,032	0,074
MV								
\hat{d}	0,018	0,015	0,033	0,027	0,110	0,085	0,178	0,162
EQM	0,081	0,082	0,072	0,078	0,122	0,141	0,076	0,087
p = 0,005								
GPH								
\hat{d}	0,236	0,210	0,295	0,249	0,399	0,367	0,455	0,401
EQM	0,093	0,104	0,029	0,041	0,091	0,105	0,027	0,041
MV								
\hat{d}	0,019	0,017	0,035	0,031	0,108	0,101	0,181	0,173
EQM	0,080	0,081	0,071	0,071	0,124	0,129	0,074	0,079
p = 0,0005								
GPH								
\hat{d}	0,234	0,231	0,292	0,286	0,400	0,396	0,450	0,442
EQM	0,089	0,091	0,030	0,032	0,089	0,090	0,030	0,032
MV								
\hat{d}	0,018	0,016	0,035	0,033	0,111	0,108	0,180	0,179
EQM	0,081	0,082	0,071	0,072	0,121	0,123	0,074	0,076

Verifica-se que o padrão observado anteriormente se manteve, ou seja, com a diminuição do valor de p os resultados ficaram mais próximos dos valores obtidos para a série original. Devido ao fato de que o valor médio de d , para a série original, está subestimado e

que o método *bootstrap* também subestima o valor do parâmetro da série original, pode-se concluir que as estimativas estão bem piores em relação ao verdadeiro valor.

5.2 ESTIMATIVA INTERVALAR

Nesta subseção são apresentados os limites dos intervalos de confiança para d e as taxas de cobertura estimadas para os modelos de longa dependência.

A Tabela 4 mostra os limites dos intervalos de confiança para d e as taxas de cobertura estimadas para o modelo **ARFIMA (0, d , 0)**. Vale lembrar que os intervalos para a série *bootstrap* foram construídos com 95% de confiança, isto é, teoricamente o percentual de cobertura deveria estar próximo deste valor.

Em geral, o intervalo de confiança percentílico obteve taxas de cobertura mais próximas do nível nominal fixado de 95% para o GPH quando utilizado $p = 0,05$ e $p = 0,005$. Vale frisar que os resultados para o estimador MV foram muito ruins, com taxas de cobertura bem abaixo do nível de 95%. Em relação à taxa de cobertura, verifica-se que o intervalo percentílico mostrou melhores percentuais de cobertura que o BC. Pode-se observar que a amplitude dos intervalos diminui quando o tamanho da série aumenta.

Apesar dos resultados obtidos com $p = 0,0005$ terem apresentado as melhores estimativas pontuais (como verificado na seção anterior), seus resultados foram ruins em relação às estimativas intervalares. Esse problema está relacionado com o fato de que ao considerar um valor muito pequeno para p , a série formada por meio do *bootstrap* pode estar idêntica à série original. Se a série *bootstrap* for equivalente a série original todos os valores médios de d estarão bem próximos, isto implica que a amplitude do intervalo será muito pequena.

É importante relembrar que os resultados das estimativas pontuais ao considerar $p = 0,05$ foram piores que $p = 0,005$. No caso em que o comprimento de cada bloco tem distribuição geométrica com $p = 0,005$, verifica-se que o método GPH apresentou probabilidade de cobertura dentro do esperado, quando $n = 500$ e $d = 0,3$ e $0,45$.

As Tabelas 5 e 6 mostram os resultados obtidos dos limites dos intervalos de confiança para d e as taxas de cobertura estimadas por estes procedimentos para os modelos **ARFIMA (1, d , 0)**, com $\phi = 0,4$ e **ARFIMA (0, d , 1)**, com $\theta = 0,4$, respectivamente.

Nota-se que o comportamento observado anteriormente se manteve, ou seja, de um modo geral o intervalo de confiança percentílico apresentou as melhores taxas de cobertura para o método de estimação GPH e $p = 0,05$ ou $0,005$. Em relação os tipos de intervalos, o intervalo percentílico mostrou melhores probabilidades de cobertura que o BC.

TABELA 4

Intervalos de confiança bootstrap para d e taxas de cobertura no modelo **ARFIMA (0, d , 0)**

<i>Série Bootstrap</i>					
		$d = 0,3$		$d = 0,45$	
		$n = 100$	$n = 500$	$n = 100$	$n = 500$
$p = 0,5$					
GPH					
IC percentílico	[-0,597 ; 0,563]	[-0,350 ; 0,320]	[-0,585 ; 0,576]	[-0,346 ; 0,323]	
	100%	93,3%	100%	0%	
IC BC	[-0,392 ; 0,721]	[-0,156 ; 0,444]	[-0,309 ; 0,776]	[-0,106 ; 0,458]	
	100%	97,5%	99,9%	56,6%	
MV					
IC percentílico	[<0,001 a 0,244]	[0,051 a 0,201]	[0,006 a 0,318]	[0,140 a 0,299]	
	11,3%	0,0%	0,0%	0,0%	
IC BC	[-0,668 a 0,432]	[-0,317 a 0,268]	[-0,596 a 0,481]	[-0,314 a 0,270]	
	64,1%	49,9%	56,1%	30,9%	
$p = 0,05$					
GPH					
IC percentílico	[-0,432 a 0,666]	[-0,188 a 0,473]	[-0,314 a 0,790]	[-0,091 a 0,578]	
	98,2%	98,2%	96,0%	91,8%	
IC BC	[-0,551 a 0,591]	[-0,245 a 0,421]	[-0,513 a 0,654]	[-0,150 a 0,504]	
	96,1%	78,7%	82,5%	63,1%	
MV					
IC percentílico	[0,063 a 0,326]	[0,200 a 0,339]	[0,186 a 0,427]	[0,356 a 0,471]	
	64,8%	82,4%	43,2%	83,2%	
IC BC	[-0,772 a 0,321]	[-0,365 a 0,249]	[-0,654 a 0,452]	[-0,291 a 0,303]	
	54,9%	46,0%	53,3%	29,4%	
$p = 0,005$					
GPH					
IC percentílico	[-0,153 a 0,549]	[-0,072 a 0,499]	[-0,011 a 0,689]	[0,074 a 0,648]	
	85,9%	93,0%	84,9%	92,4%	
IC BC	[-0,538 a 0,433]	[-0,250 a 0,367]	[-0,441 a 0,528]	[-0,133 a 0,486]	
	75,1%	71,8%	66,9%	64,1%	
MV					
IC percentílico	[0,132 a 0,294]	[0,232 a 0,332]	[0,271 a 0,407]	[0,382 a 0,466]	
	48,1%	79,6%	27,8%	75,9%	
IC BC	[-0,609 a 0,301]	[-0,296 a 0,281]	[-0,474 a 0,441]	[-0,169 a 0,412]	
	52,7%	50,6%	51,5%	48,0%	
$p = 0,0005$					
GPH					
IC percentílico	[0,256 a 0,327]	[0,119 a 0,414]	[0,402 a 0,474]	[0,273 a 0,566]	
	7,6%	68,8%	9,8%	68,2%	
IC BC	[-0,198 a 0,370]	[-0,132 a 0,322]	[-0,077 a 0,476]	[0,003 a 0,459]	
	60,8%	60,5%	55,5%	60,9%	
MV					
IC percentílico	[0,226 a 0,263]	[0,259 a 0,309]	[0,353 a 0,385]	[0,406 a 0,449]	
	13,8%	44,9%	11,5%	43,2%	
IC BC	[-0,233 a 0,296]	[-0,128 a 0,300]	[-0,089 a 0,445]	[0,016 a 0,447]	
	43,8%	48,6%	44,1%	49,2%	

Obs.: Em negrito estão as taxas de cobertura mais próximas do nível nominal de 95%

TABELA 5

Intervalos de confiança bootstrap para d e taxas de cobertura no modelo ARFIMA (1, d , 0)

	Série Bootstrap			
	$d = 0,3$		$d = 0,45$	
	$N = 100$	$n = 500$	$n = 100$	$n = 500$
$p = 0,5$				
GPH				
IC percentílico	[-0,584 a 0,576] 100%	[-0,347 a 0,321] 94,6%	[-0,576 a 0,585] 100%	[-0,345 a 0,324] 0,0%
IC _{BC}	[-0,371 a 0,741] 100%	[-0,146 a 0,449] 97,6%	[-0,296 a 0,780] 99,8%	[-0,102 a 0,458] 55,6%
MV				
IC percentílico	[0,025 a 0,371] 96,2%	[0,174 a 0,339] 98,2%	[0,067 a 0,412] 12%	[0,232 a 0,403] 0,0%
IC _{BC}	[-0,554 a 0,493] 62,4%	[-0,318 a 0,270] 50,2%	[-0,541 a 0,503] 58,0%	[-0,314 a 0,271] 30,6%
$p = 0,05$				
GPH				
IC percentílico	[-0,368 a 0,734] 99,2%	[-0,179 a 0,480] 98,9%	[-0,257 a 0,853] 97,9%	[-0,084 a 0,582] 92,4%
IC _{BC}	[-0,543 a 0,617] 96,2%	[-0,234 a 0,427] 80,3%	[-0,476 a 0,700] 84,5%	[-0,141 a 0,509] 63,6%
MV				
IC percentílico	[0,321 a 0,474] 37,2%	[0,464 a 0,494] 0,0%	[0,409 a 0,487] 75,2%	[0,494 a 0,498] 0,0%
IC _{BC}	[-0,666 a 0,434] 67,4%	[-0,309 a 0,311] 56,9%	[-0,521 a 0,594] 69,5%	[-0,178 a 0,443] 55,1%
$p = 0,005$				
GPH				
IC percentílico	[-0,081 a 0,626] 88,2%	[-0,062 a 0,517] 94,8%	[0,060 a 0,767] 88,3%	[0,085 a 0,665] 94,3%
IC _{BC}	[-0,482 a 0,479] 81,9%	[-0,254 a 0,379] 74,0%	[-0,383 a 0,590] 75,4%	[-0,130 a 0,496] 66,2%
MV				
IC percentílico	[0,404 a 0,470] 7,8%	[0,481 a 0,497] 0,0%	[0,460 a 0,486] 20,6%	[0,496 a 0,498] 0,0%
IC _{BC}	[-0,525 a 0,377] 58,1%	[-0,284 a 0,299] 53,0%	[-0,370 a 0,546] 60,5%	[-0,137 a 0,456] 54,5%
$p = 0,0005$				
GPH				
IC percentílico	[0,333 a 0,401] 6,1%	[0,132 a 0,430] 72,2%	[0,488 a 0,559] 6,9%	[0,288 a 0,580] 72,0%
IC _{BC}	[-0,148 a 0,418] 69,5%	[-0,108 a 0,343] 67,3%	[-0,021 a 0,550] 64,7%	[0,029 a 0,484] 65,6%
MV				
IC percentílico	[0,446 a 0,462] 7%	[0,488 a 0,494] 0,0%	[0,478 a 0,484] 2,9%	[0,497 a 0,498] 0,0%
IC _{BC}	[-0,150 a 0,373] 51,1%	[-0,111 a 0,325] 52,3%	[-0,001 a 0,540] 54,2%	[0,040 a 0,483] 54,3%

Obs.: Em negrito estão as taxas de cobertura mais próximas do nível nominal de 95%

TABELA 6

Intervalos de confiança bootstrap e taxas de cobertura para d no modelo **ARFIMA (0; d; 1)**

	Série Bootstrap			
	d = 0,3		d = 0,45	
	N = 100	n = 500	n = 100	n = 500
$p = 0,5$				
GPH				
IC percentílico	[-0,618 a 0,543]	[-0,352 a 0,316]	[-0,604 a 0,557]	[-0,349 a 0,319]
	100%	89,0%	100%	0,0%
IC _{BC}	[-0,436 a 0,684]	[-0,160 a 0,440]	[-0,340 a 0,756]	[-0,109 a 0,453]
	100%	98,6%	99,9%	54,1%
MV				
IC percentílico	[<0,001 a 0,083]	[<0,001 a 0,04]	[<0,001 a 0,152]	[0,005 a 0,129]
	0,0%	0,0%	0,0%	0,0%
IC _{BC}	[-1,22 a -0,144]	[-0,491 a 0,150]	[-0,834 a 0,300]	[-0,339 a 0,256]
	0,0%	30,4%	41,2	28,3%
$p = 0,05$				
GPH				
IC percentílico	[-0,495 a 0,611]	[-0,202 a 0,458]	[-0,376 a 0,7334]	[-0,100 a 0,564]
	97,6%	97,3%	92,9%	89,2%
IC _{BC}	[-0,570 a 0,572]	[-0,249 a 0,417]	[-0,535 a 0,626]	[-0,162 a 0,494]
	97,1%	80,0%	82,9%	62,3%
MV				
IC percentílico	[<0,001 a 0,062]	[0,0006 a 0,069]	[0,004 a 0,185]	[0,085 a 0,227]
	0,0%	0,0%	0,0%	0,0%
IC _{BC}	[-1,101 a -0,036]	[-0,402 a 0,229]	[-0,758 a 0,345]	[-0,272 a 0,334]
	15,3%	44,3%	44,9%	34,9%
$p = 0,005$				
GPH				
IC percentílico	[-0,210 a 0,489]	[-0,082 a 0,497]	[-0,064 a 0,630]	[0,065 a 0,646]
	79,7%	93,3%	77,6%	93,2%
IC _{BC}	[-0,551 a 0,401]	[-0,264 a 0,369]	[-0,479 a 0,481]	[-0,144 a 0,486]
	69,8%	71,5%	59,4%	64,3%
MV				
IC percentílico	[0,001 a 0,039]	[0,003 a 0,067]	[0,029 a 0,152]	[0,107 a 0,219]
	0,0%	0,0%	0,0%	0,0%
IC _{BC}	[-0,899 a 0,106]	[-0,325 a 0,271]	[-0,549 a 0,381]	[-0,161 a 0,429]
	30,2%	51,7%	46,5%	53,0%
$p = 0,0005$				
GPH				
IC percentílico	[0,180 a 0,254]	[0,110 a 0,404]	[0,342 a 0,415]	[0,263 a 0,557]
	9,7%	67,5%	8,7%	66,8%
IC _{BC}	[-0,248 a 0,333]	[-0,137 a 0,320]	[-0,131 a 0,424]	[-0,004 a 0,455]
	54,6%	61,2%	44,9%	59,1%
MV				
IC percentílico	[0,011 a 0,024]	[0,013 a 0,048]	[0,094 a 0,127]	[0,146 a 0,199]
	0,0%	0,0%	0,0%	0,0%
IC _{BC}	[-0,479 a 0,179]	[-0,160 a 0,276]	[-0,130 a 0,388]	[-0,001 a 0,438]
	36,3%	46,3%	38,7%	48,2%

Obs.: Em negrito estão as taxas de cobertura mais próximas do nível nominal de 95%

6 APLICAÇÕES EM SÉRIES REAIS

Nesta seção será apresentada a aplicação do *bootstrap* estacionário em dados reais com a finalidade de fazer inferência sobre o parâmetro de longa dependência.

As séries utilizadas foram:

- Nível do rio Nilo entre os anos 622 e 1284.
- Umidade relativa do ar (%) entre 1º de janeiro a 31 de dezembro na cidade de São Paulo, Brasil.

6.1 NÍVEL DO RIO NILO

A série sobre o nível mínimo anual do rio Nilo é bastante utilizada nos estudos envolvendo a característica de longa dependência. Esses dados foram coletados nos anos de 622 a 1284, totalizando 663 observações.

O gráfico da série nível mínimo anual do rio Nilo e a função de autocorrelação da mesma estão apresentados nas Figuras 6.1.1 e 6.1.2, respectivamente. Ao observar o comportamento do nível mínimo do rio Nilo, à primeira vista, pode-se inferir que a série parece ser um processo estocástico não-estacionário. Visualmente, ela não se desenvolve no tempo de forma aleatória ao redor de uma média constante, não refletindo nenhum tipo de equilíbrio estável. Já ao analisar a Figura 6.1.2, verifica-se que função de autocorrelação decresce hiperbolicamente para zero, ou seja, os dados apresentam memória longa. O comportamento que indica a não estacionariedade da série pode ser explicado pela presença da característica de longa dependência entre as observações.

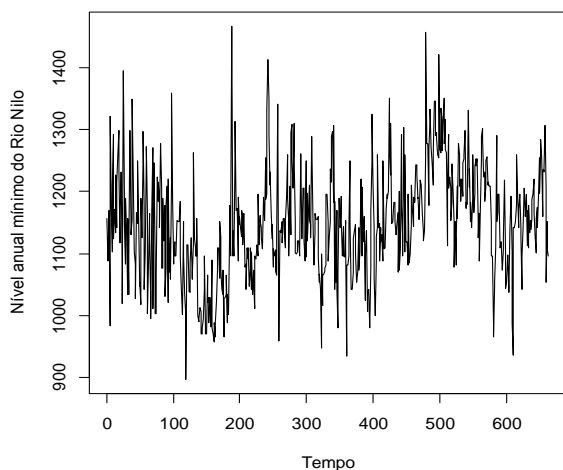


Figura 6.1.1: Gráfico da série nível anual mínimo do rio Nilo.

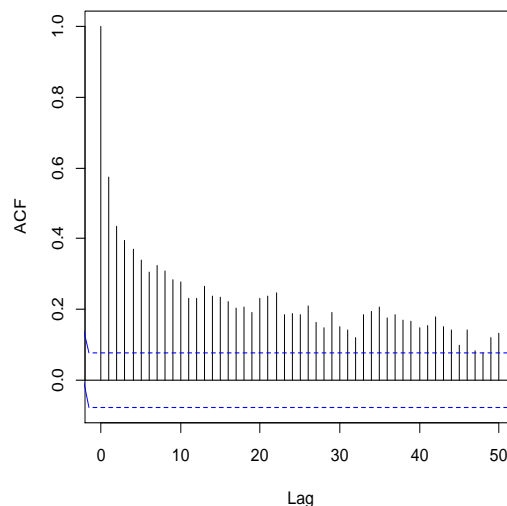


Figura 6.1.2: Gráfico da função de autocorrelação da série nível anual mínimo do rio Nilo.

Como visto anteriormente a série em estudo será modelada pelo modelo **ARFIMA**. Em relação aos resultados obtidos na simulação, iremos utilizar somente o estimador GPH, uma vez que o MV não reportou resultados confiáveis. Neste método, inicialmente, diferenciamos a série de modo a retirar o efeito de longa dependência do processo e, posteriormente, estimam-se os parâmetros do modelo pelos métodos usuais para os modelos **ARMA**. Por meio das funções de autocorrelação e autocorrelação parcial identificaremos a ordem do modelo **ARMA**. De acordo com os valores de p e q , essas funções descreverão um comportamento específico. Quando o processo é um ruído branco, a função de autocorrelação e a função de autocorrelação parcial não apresentam nenhum lag com correlação significativa, ou seja, as funções são iguais à zero para todas as defasagens.

A Tabela 7 apresenta o resultado da estimativa pontual do parâmetro de memória longa em relação ao método GPH. O parâmetro estimado satisfaz as condições de estacionariedade e invertibilidade. O valor estimado pelo método GPH foi de 0,395.

É importante destacar que o modelo **ARFIMA (0, d , 0)**, como visto na análise de resíduos (Figuras 6.1.3 a 6.1.5), apresentou o melhor desempenho e por isso foi necessário a utilização de um bandwidth ($g(n) = n^\alpha$) com $\alpha > 0,5$. Então, para a estimação do parâmetro de memória para a série do rio Nilo optou-se pela utilização α de 0,7.

TABELA 7
Estimação pontual do parâmetro d para a série do rio Nilo

Método	\hat{d}	EQM
GPH	0,395	0,07

A Tabela 8 apresenta a estimação pontual e intervalar (intervalo *bootstrap* percentílico) do parâmetro de memória em relação ao método de *bootstrap* estacionário. Nota-se que o comportamento observado nas simulações se manteve, ou seja, com a diminuição do valor de p os resultados ficaram mais próximos dos valores obtidos para a série original.

TABELA 8
Estimação pontual e intervalar do parâmetro d em relação aos métodos *bootstrap*

	<i>Bootstrap</i>			
	$p = 0,5$	$p = 0,05$	$P = 0,005$	$p = 0,0005$
GPH				
\hat{d}	0,048	0,358	0,394	0,393
EQM	0,126	0,09	0,005	0,001
IC _{95%}	[-0,107 a 0,186]	[0,173 a 0,526]	[0,256 a 0,535]	[0,313 a 0,466]

Os gráficos da série diferenciada em relação ao tempo e da função de autocorrelação da mesma estão apresentados, respectivamente, pelas Figuras 6.1.3 e 6.1.4. Ao analisar o comportamento da série devidamente diferenciada, observa-se que a série se desenvolve no tempo de forma aleatória ao redor de uma média constante, ou seja, é estacionária. E por meio da análise das funções de autocorrelações, há indícios que o melhor modelo a ser ajustado é **ARFIMA (0, d , 0)**.

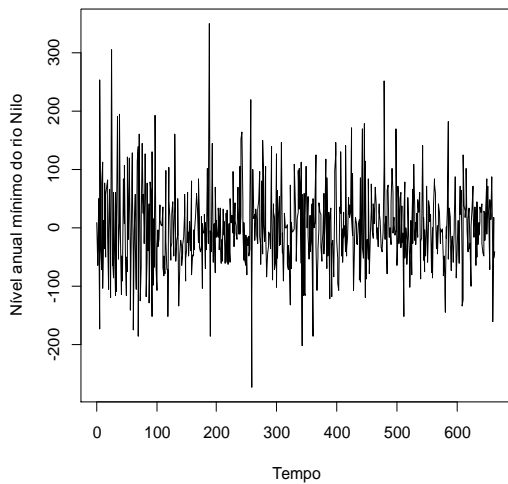


Figura 6.1.3: Gráfico da série do rio Nilo diferenciada

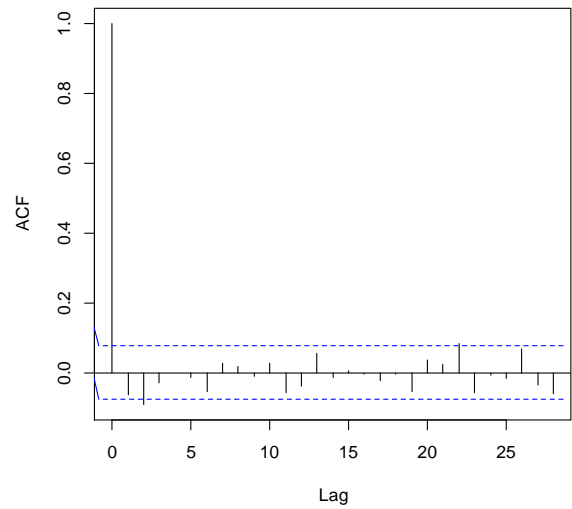


Figura 6.1.4: Gráfico da função de autocorrelação da série do rio Nilo diferenciada

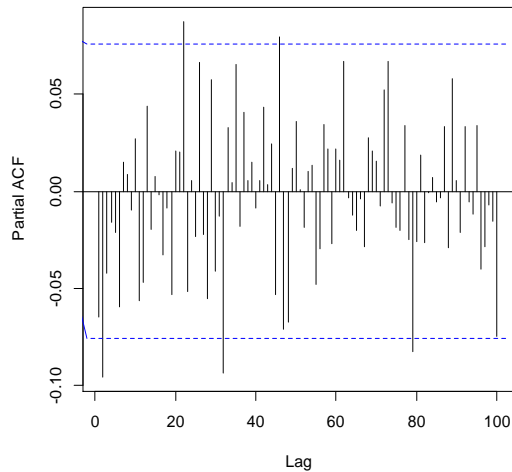


Figura 6.1.5: Gráfico da função de autocorrelação parcial da série do rio Nilo diferenciada

6.2 UMIDADE RELATIVA DO AR (%)

A série sobre a umidade relativa do ar está disponível em <http://www.ime.usp.br/~pam>. Esses dados foram coletados diariamente, ao meio dia, na cidade de São Paulo entre 01 de janeiro a 31 de dezembro, totalizando 365 observações.

O gráfico da série umidade relativa do ar (%) e a função de autocorrelação da mesma estão apresentados nas Figuras 6.2.1 e 6.2.2, respectivamente, e verifica-se o mesmo comportamento da série do rio Nilo. Observa-se que a série não se desenvolve no tempo de forma aleatória ao redor de uma média constante. Para a função de autocorrelação, nota-se um decaimento lento. Há indícios de que um possível modelo a ser ajustado é o **ARFIMA**.

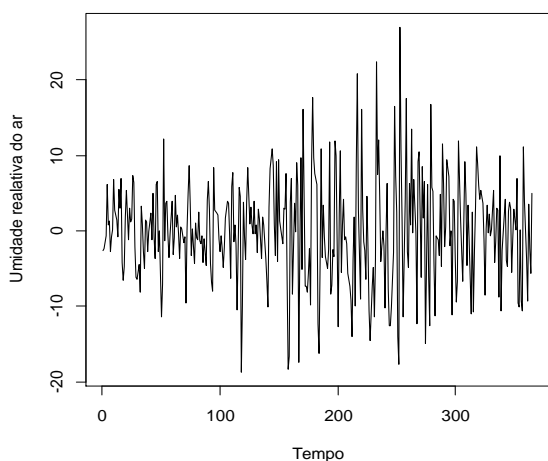


Figura 6.2.1: Gráfico da série umidade relativa do ar (%)

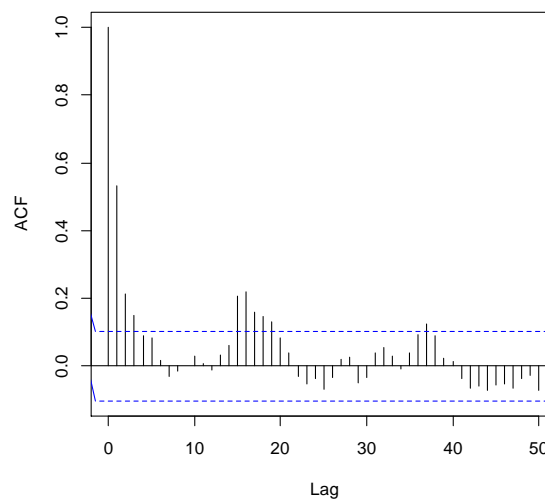


Figura 6.2.2: Gráfico da função de autocorrelação da série umidade relativa do ar (%)

A Tabela 9 apresenta a estimação pontual do parâmetro de longa dependência da série umidade relativa do ar (%). Como mencionado, anteriormente, o parâmetro d foi estimado apenas pelo método GPH. Nota-se que a estimação pontual obtida para o parâmetro de memória longa foi de 0,467. Neste caso, o α utilizado foi de 0,5.

TABELA 9

Estimação pontual do parâmetro d para a série umidade relativa do ar (%)

Método	\hat{d}	EQM
GPH	0,467	0,187

Uma vez determinado o grau d , o passo seguinte é identificar os graus p e q dos polinômios $\phi(B)$ e $\theta(B)$ do modelo ARMA aplicado à série diferenciada. Para averiguar se

o modelo **ARFIMA (0, d, 0)** ajustado está adequado é necessário aplicar o filtro $(1 - B)^{0,467B}$ à série analisada e verificar se a série filtrada apresenta um comportamento de ruído branco. Caso contrário, utilizam-se as informações das funções de autocorrelações da série filtrada para identificar o modelo adequado.

As funções de autocorrelação e autocorrelação parcial da série umidade relativa diferenciada estão apresentadas pelas Figuras 6.2.3 e 6.2.4, respectivamente. Após verificação dos possíveis modelos a serem ajustados a esta série, o que apresentou o melhor desempenho foi o **ARFIMA (1, d, 2)**.

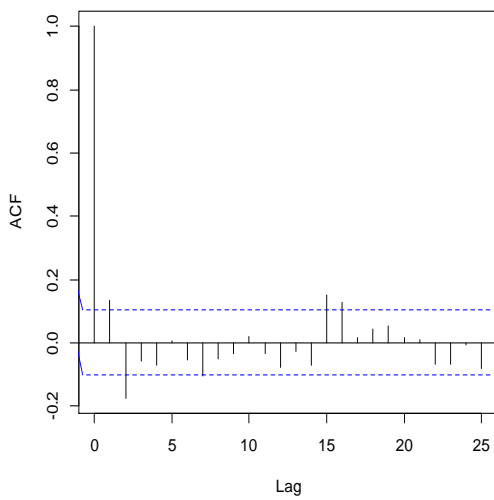


Figura 6.2.3: Gráfico da função de autocorrelação da série diferenciada da umidade relativa do ar (%)

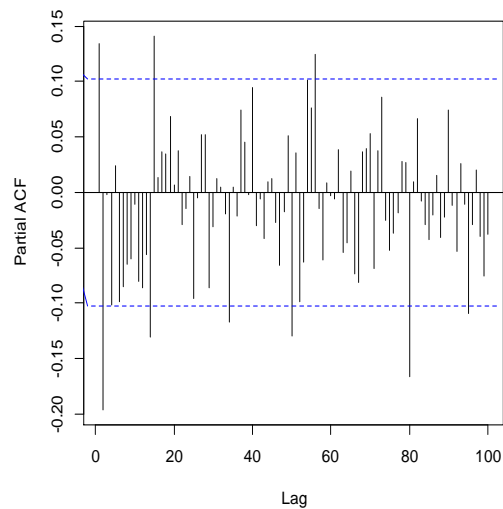


Figura 6.2.4: Gráfico da função de autocorrelação parcial da série diferenciada da umidade relativa do ar (%)

A Tabela 10 apresenta os valores dos coeficientes do modelo ajustado para a série umidade relativa do ar (%). O modelo ajustado está apresentado na equação a seguir:

$$(1 - 0,676B)(1 - B)^{0,467B} y_t = (1 + 0,524B + 0,308B^2)u_t$$

TABELA 10
Ajuste do modelo para a série umidade relativo do ar (%)

	Coefficiente	Erro padrão
<i>Constante</i>	-0,009	0,184
<i>D</i>	0,467	0,187
<i>AR</i>	0,676	0,098
<i>MA1</i>	-0,524	0,106
<i>MA2</i>	-0,308	0,058

As Figuras 6.2.5 e 6.2.6 apresentam as funções de autocorrelação e autocorrelação parcial para a série de resíduos estimados. Como as funções indicam que o processo gerador de \hat{u}_t é um ruído branco, pode se dizer que o modelo escolhido para a série está adequado.

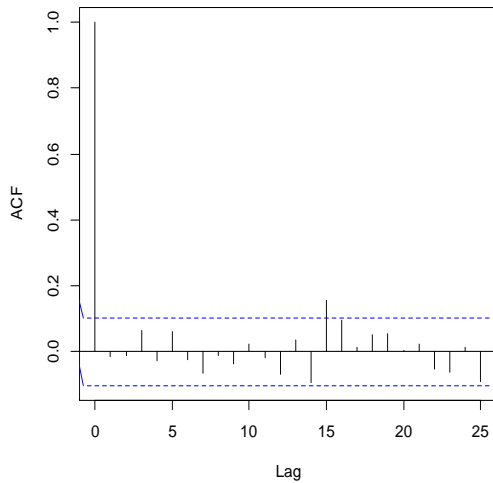


Figura 6.2.5: Gráfico da função de autocorrelação para a série de resíduos estimados.

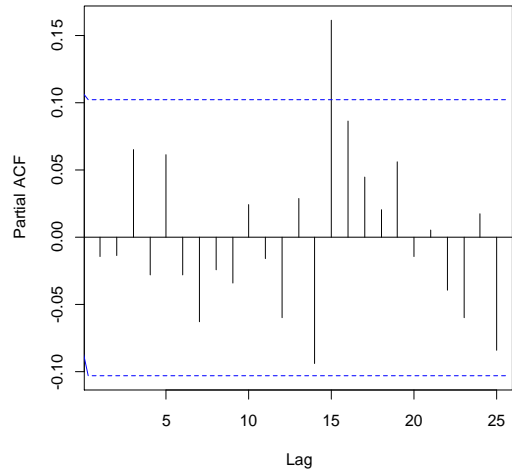


Figura 6.2.6: Gráfico da função de autocorrelação parcial para a série de resíduos estimados.

A Tabela 11 mostra o desempenho dos estimadores para o modelo **ARFIMA (1, d , 2)**. Neste caso, a inserção dos termos **MA** (médias móveis) e **AR** (autorregressivos) causou subestimação do parâmetro d .

Verifica-se que o padrão observado anteriormente se manteve, ou seja, com a diminuição do valor de p os resultados ficaram mais próximos dos valores obtidos para a série real.

TABELA 11

Estimação pontual e intervalar do parâmetro d em relação aos métodos *bootstrap*

	<i>Bootstrap</i>			
	$p = 0,5$	$p = 0,05$	$p = 0,005$	$p = 0,0005$
GPH				
\hat{d}	0,006	0,094	0,273	0,437
EQM	0,248	0,168	0,068	0,007
IC _{95%}	[-0,400 a 0,359]	[-0,277 a 0,410]	[0,350 a 0,458]	[0,168 a 0,466]

7 CONCLUSÃO

Neste trabalho foi realizado um estudo sobre a utilização da técnica de *bootstrap* estacionário para a inferência sobre o parâmetro d em modelos com longa dependência.

O efeito do parâmetro p da distribuição geométrica (que define os tamanhos dos blocos nas séries *bootstrap*) sobre as estimativas pontuais e intervalares de d foram avaliados através de simulações Monte Carlo. Os resultados mostram que deve-se tomar certo cuidado para a escolha de p , já que valores muito pequenos levam a uma aproximação muito boa para as estimativas pontuais, mas resultam em péssimos estimadores intervalares. Isto ocorre porque valores pequenos de p acarretam em blocos grandes, o que faz com que as séries *bootstrap* sejam muito semelhantes às séries originais. Desta forma, a variabilidade obtida nas repetições *bootstrap* é muito pequena e não é suficiente para reproduzir a distribuição empírica dos estimadores utilizados.

Os resultados mostram que o método *bootstrap* estacionário empregado aproxima de forma satisfatória apenas o comportamento dos estimadores GPH para modelos **ARFIMA** $(0,d,0)$, ou seja, que não contenham componentes autorregressivos ou médias móveis. Os estimadores de máxima verossimilhança para d possuem maior vício, e o mesmo é carregado para as séries *bootstrap*, acarretando em um pior desempenho para este método. Além disto, os resultados mostram também que os intervalos de confiança percentílico apresentam coberturas mais próximas ao valor nominal fixado de 95% em relação ao intervalo BC. O intervalo de confiança percentílico obteve as melhores probabilidades de cobertura para o método de estimação GPH.

Assim, é importante destacar que os melhores comprimentos de blocos foram obtidos com o parâmetro p igual a 0,05 e 0,005. E vale a pena ressaltar que devido ao fato do valor estimado do parâmetro d por meio do *bootstrap* sempre está subestimado em relação ao valor real, os resultados referentes ao intervalo de confiança *bootstrap* de correção de vício ficam comprometidos.

Vale lembrar que na literatura encontramos o *bootstrap* estacionário apenas para o caso em que a série é fracamente estacionária, o que pode não ocorrer para séries de memória longa, principalmente se o valor de d for grande.

REFERÊNCIAS BIBLIOGRÁFICAS

- Alonso, A., Peña, D. and Romo, J. (2000). Sieve bootstrap prediction intervals. In: OMPSTAT' 2000. *Proceedings in Computational Statistics*, 181-186.
- Arteche, J. and Orbe, J. (2005). Bootstrapping the log-periodogram regression. *Economics Letters*, 86, 79-85.
- Beran, J. (1994). *Statistics for Long-Memory Processes*. New York: Chapman & Hall.
- Box, G. E. P. and Jenkins, G. M (1976) *Times Series Analysis: Forecasting and Control*. San Francisco: Holden-Day.
- Doukhan, P., Oppenheim, G. and Taqqu, M. S (2003). *Theory and Applications of Long-Range Dependence*. Boston: Birkhauser.
- Efron, B. (1979). Bootstrap methods: another look at the jackknife. *The Annals of Statistics* 7: 1-25.
- Efron, B. & Tibshirani, R. (1993). *An Introduction to the Bootstrap*. London: Chapman and Hall.
- Fox, R. and Taqqu, M.S. (1986). Large-sample properties of parameter estimates for strongly dependent stationary Gaussian time series. *The Annals of Statistics*, 14, 517–532.
- Franco, G.C. and Reisen, V.A. (2004). Bootstrap techniques in semiparametric estimation methods for ARFIMA models: a comparison study. *Computational Statistics*, 19, 243-259.
- Franco, G. C., Reisen, V.A. (2007) Bootstrap approaches and confidence intervals for stationary and non-stationary long range dependence processes. *Physica A*, 375, 546-562.
- Geweke, J. and Porter-Hudak, S. (1983), The estimation and application of long memory time series model, *Journal of Time Series Analysis* 4, 221–238.
- Granger, C.M. G. and Joyeux, R. (1980) An Introduction to long Memory Times Series Models and Fractional Differencing. *Journal of Time Series Analysis*, Vol. 1, 15-29.
- Hall.P., Horowitz, J. and Jing, B. (1995), On Blocking Rules for the Bootstrap with Dependent Data, *Biometrika* 82, 561-574.
- Hosking, J. (1981). Fractional differencing. *Biometrika*, 68(1), 165-176.
- Hurst, H.E. (1951). “Long-term storage capacity of reservoirs”. *Transactions of American Society in Civil Engineers*, Vol. 116, pp. 165-167.
- Kunsch, H.R. (1989) The Jackknife and The Bootstrap for General Stationary Observations. *The Annals of Statistics*, 17, 1217-1241.

Lahiri, S. N. (1993). On the moving block bootstrap under long range dependence. *Statist. Probab. Lett.*, 18, 405-413.

Liu, R.Y., Singh, K. Moving Blocks Jackknife and Bootstrap Capture Weak Dependence. In LePage, R. And Billard, L. (orgs.), *Exploring the Limits of Bootstrap*, New York, John Wiley, 225-248, 1992.

Morettin, P. A. e Tolo, C. M. C (2004). *Análise de Séries Temporais*. São Paulo: Edgard Blücher.

Morettin, P. A. *Econometria financeira: um curso em séries temporais financeiras*. 1ed. São Paulo: Blucher, 2008.

Politis, D. N. and Romano, J. P. (1991) The Stationary Bootstrap, *Journal of the American Statistical Association*, 1303 - 1313, 1991.

Whittle, P. (1953). Estimation and information in stationary time series. *Arkiv for Matematik*, Vol. 2, pp. 423-434

APÊNCICE A

Segue abaixo o programa escrito na linguagem R para a implementação do método bootstrap estacionário, assim como a construção dos intervalos de confiança percentílico e bootstrap BC.

```
rm(list=ls(all=TRUE))
library(boot)
set.seed(300)
library(fracdiff)
n <- 500
p <- 0.5
d <- 0.45
r <- 1000
REPLICAS <- 1000
alpha=.05
conf= .95
Ind_GPH_boot_perct = Ind_MV_boot_perct = Ind_GPH_boot_BC = Ind_MV_boot_BC = 0
MEMORY.LONG = mGPH_BOOT = d_GPH_BOOT = d_hat_MEAN_BOOT <- rep ( )
d_hat_MEAN =EQM_GPH_BOOT <- rep();
Serie_New = Serie_boot = comp_gerado = inicio_S_new = Posicao_S_End = Reiniciar <- list()
serie = Serie_boot= aux_1 = AUX_BOOT = AUX_EQM = AUX2_BOOT = AUX_MAX <- NULL
GPH = EQM_GPH = mGPH = d_MAX.VEROS = EQM_MAX.VEROS <- rep (0,r)
MAX.VEROS_BOOT =d_MEAN_BOOT=EQM_MEAN<- rep (0,r)
d_MEAN_GPH_BOOT = EQM_MEAN_GPH_BOOT = d_MAX.VEROS_BOOT <- rep ( )
EQM_MEAN_MAX =d_MEAN_MAX_BOOT= EQM_MEAN_MAX_BOOT<-rep ( )
AUX_IC_GPH = AUX_IC_MV = AUX_CIp <- NULL
LS_BC = LI_BC = LI_MV = LS_MV = LI_GPH = LS_GPH = IC_GPH = LI_BC_MV=
LS_BC_MV<- rep()
IC_GPH_BOOT= IC_GPH_B = LIMITE_SUPERIOR_GPH = LIMITE_INFERIOR_GPH <- rep()
IC_MV_BOOT = IC_MV_B = LIMITE_SUP_MV = LIMITE_INFERIOR_MV<- rep()

for (i in 1: r ){
  MEMORY.LONG <- fracdiff.sim(n,ar=0.4 , ma=0, d = d )
  mGPH<- fdGPH(MEMORY.LONG$series)
  GPH[i]= mGPH$d
  EQM_GPH[i] =(GPH[i]-d)^2
  MAX.VEROS<-fracdiff(MEMORY.LONG$series, nar = 0, nma = 0)
  d_MAX.VEROS[i]=MAX.VEROS$d
```

EQM_MAX.VEROS[i]=(d_MAX.VEROS[i]-d)^2

===== BOOTSTRAP ESTACIONÁRIO

```
for (j in 1: REPLICAS) {
  while(length(serie) < n) {
    inicio_S_new <- ceiling      (round(runif(1,1,n)))
    comp_gerado <- rgeom(1,p)
    Posicao_S_End <- (inicio_S_new + comp_gerado - 1)
    Reiniciar <- (Posicao_S_End - n )
    if(Posicao_S_End <= n) {Serie_New = c(MEMORY.LONG$series
[inicio_S_new : Posicao_S_End])}
    else if (Posicao_S_End > n){Serie_New = c(MEMORY.LONG$series
[inicio_S_new : n], MEMORY.LONG$series[1:Reiniciar])}
    aux_1 = Serie_New
    serie=c(serie, aux_1) }
    Serie_boot [[j]] = serie[1:n]
    serie = aux_1 <- NULL
    mGPH_BOOT <- fdGPH(Serie_boot[[j]])
    AUX_BOOT <- mGPH_BOOT$d
    d_GPH_BOOT <- c( d_GPH_BOOT, AUX_BOOT)
    d_MEAN_BOOT <-mean(d_GPH_BOOT)
    EQM_GPH_BOOT=(AUX_BOOT - d)^2
    AUX_EQM = c(AUX_EQM, EQM_GPH_BOOT )
    EQM_MEAN = mean(AUX_EQM )
    AUX2_BOOT = MAX.VEROS_BOOT$d
    d_MAX.VEROS_BOOT <- c(d_MAX.VEROS_BOOT, AUX2_BOOT)
    d_MEAN_M_BOOT = mean(d_MAX.VEROS_BOOT)
    EQM_MAX_BOOT =(AUX2_BOOT - d)^2
    AUX_MAX = c(AUX_MAX,EQM_MAX_BOOT )
    EQM_MEAN_MAX = mean (AUX_MAX)
  }
  d_MEAN_GPH_BOOT=c(d_MEAN_GPH_BOOT, d_MEAN_BOOT)
  EQM_MEAN_GPH_BOOT =c (EQM_MEAN_GPH_BOOT, EQM_MEAN)
  d_MEAN_MAX_BOOT=c( d_MEAN_MAX_BOOT,d_MEAN_M_BOOT)
  EQM_MEAN_MAX_BOOT= c(EQM_MEAN_MAX_BOOT, EQM_MEAN_MAX)
```

```
# INTERVALO DE CONFIANÇA BOOTSTRAP PERCENTÍLICO GPH
```

```
IC_GPH_B <- quantile( d_GPH_BOOT, probs = c(2.5, 97.5)/100 )  
LI_GPH[i]=IC_GPH_B[1]  
LS_GPH[i]=IC_GPH_B[2]  
IC_GPH= cbind( LI_GPH , LS_GPH)  
if( LI_GPH[i] < d & LS_GPH[i] > d )  
{Ind_GPH_boot_perct = Ind_GPH_boot_perct + 1}
```

```
# INTERVALO DE CONFIANÇA BOOTSTRAP PERCENTÍLICO MV
```

```
IC_MV_B <- quantile( d_MAX.VEROS_BOOT, probs = c(2.5, 97.5)/100 )  
LI_MV [i]<- IC_MV_B[1]  
LS_MV [i]<- IC_MV_B[2]  
if( LI_MV[i]< d & LS_MV[i] > d )  
{Ind_MV_boot_perct <- Ind_MV_boot_perct + 1}
```

```
# INTERVALO DE CONFIANÇA BOOTSTRAP DE CORREÇÃO DE VÍCIO GPH
```

```
BC_GPH=function(GPH,d_GPH_BOOT,conf) {  
alpha=c((.5-conf/2),(.5+conf/2))  
nBoot=length(d_GPH_BOOT)  
z0=qnorm(sum(d_GPH_BOOT < GPH)/nBoot)  
zCI=qnorm(alpha)  
p=pnorm(2*z0+zCI)  
CIp=quantile(d_GPH_BOOT, probs=p)  
return(list(probs=p,CIp=CIp)) }  
A= BC_GPH( GPH, d_GPH_BOOT,conf)  
LI_BC [i]<- A$CIp[[1]]  
LS_BC [i]<- A$CIp[[2]]  
if( LI_BC[i]< d & LS_BC[i] > d )  
{Ind_GPH_boot_BC <- Ind_GPH_boot_BC + 1}
```

```
#INTERVALO DE CONFIANÇA BOOTSTRAP DE CORREÇÃO DE VÍCIO MV
```

```
BC_GPH=function(d_MAX.VEROS,d_MAX.VEROS_BOOT,conf) {  
alpha=c((.5-conf/2),(.5+conf/2))
```

```

nBoot_MV=length(d_MAX.VEROS_BOOT)
z0_MV=qnorm(sum(d_MAX.VEROS_BOOT < d_MAX.VEROS)/nBoot_MV)
zCI_MV=qnorm(alpha)
p_MV=pnorm(2*z0_MV+zCI_MV)      # Prob para ajuste do BC.
CIp_MV=quantile(d_GPH_BOOT, probs=p_MV)
return(list(probs=p_MV,CIp_MV=CIp_MV)) }
A2= BC_GPH( d_MAX.VEROS,d_MAX.VEROS_BOOT,conf)
LI_BC_MV [i]<- A2$CIp_MV[[1]]
LS_BC_MV [i]<- A2$CIp_MV[[2]]
if( LI_BC_MV[i]< d & LS_BC_MV[i] > d )
{Ind_MV_boot_BC <- Ind_MV_boot_BC + 1}
d_MEAN_GPH_BOOT = c(d_MEAN_GPH_BOOT, d_MEAN_BOOT)
EQM_MEAN_GPH_BOOT = c( EQM_MEAN_GPH_BOOT, EQM_MEAN)
d_MEAN_MAX_BOOT = c( d_MEAN_MAX_BOOT,d_MEAN_M_BOOT)
EQM_MEAN_MAX_BOOT = c(EQM_MEAN_MAX_BOOT, EQM_MEAN_MAX)
AUX_BOOT = d_GPH_BOOT = EQM_MEAN = AUX_EQM <- NULL
AUX2_BOOT = d_MAX.VEROS_BOOT= AUX_MAX= EQM_MEAN_MAX <- NULL
}

Media_d_MEAN_GPH_BOOT = mean(d_MEAN_GPH_BOOT)
Media_d_MEAN_GPH_BOOT
Media_EQM_MEAN_GPH_BOOT = mean(EQM_MEAN_GPH_BOOT)
Media_EQM_MEAN_GPH_BOOT
Media_d_MEAN_MAX_BOOT = mean( d_MEAN_MAX_BOOT)
Media_d_MEAN_MAX_BOOT
Media_EQM_MEAN_MAX_BOOT = mean(EQM_MEAN_MAX_BOOT)
Media_EQM_MEAN_MAX_BOOT
Média_GPH = mean(GPH);Média_GPH
Média_EQM_GPH = mean (EQM_GPH);Média_EQM_GPH
Média_d_MAX.VEROS = mean(d_MAX.VEROS);Média_d_MAX.VEROS
Média_EQM_MAX.VEROS = mean (EQM_MAX.VEROS);Média_EQM_MAX.VEROS
Ind_MV_boot_perct
Ind_GPH_boot_perct
Ind_GPH_boot_BC
Ind_MV_boot_BC
Média_LI_GPH = mean( LI_GPH); Média_LI_GPH
Média_LS_GPH = mean( LS_GPH); Média_LS_GPH
Média_LI_MV = mean( LI_MV);Média_LI_MV

```

Média_LS_MV = mean(LS_MV); Média_LS_MV

Média_LI_BC = mean(LI_BC);Média_LI_BC

Média_LS_BC = mean(LS_BC);Média_LS_BC

Média_LI_BC_MV= mean(LI_BC_MV);Média_LI_BC_MV

Média_LS_BC_MV= mean(LS_BC_MV);Média_LS_BC_MV