



Universidade de Brasília

Instituto de Exatas

Departamento de Estatística

ANÁLISE DE SÉRIES TEMPORAIS NÃO-LINEARES COM MODELOS AUTO-REGRESSIVOS THRESHOLD

Leonardo Melo Bosque

Brasília

Novembro de 2014

Leonardo Melo Bosque – 10/0033768

ANÁLISE DE SÉRIES TEMPORAIS NÃO-LINEARES COM MODELOS AUTO-REGRESSIVOS THRESHOLD

Relatório apresentado à disciplina de Estágio supervisionado II do curso de graduação em Estatística, Departamento de Estatística, Instituto de Exatas, Universidade de Brasília, como parte dos requisitos necessários para o grau de Bacharel em Estatística.

Orientador: Prof. Dr. Raul Yukihiro Matsushita

Brasília

Novembro de 2014

Sumário

1. Introdução.....	5
2. Metodologia	7
2.1 Processos estocásticos.....	7
2.2 Séries temporais lineares e não-lineares	7
2.3 Modelos Auto-regressivos.....	8
2.4 Modelos TAR	8
2.4.1 Modelos SETAR	9
3. Séries Temporais Não-lineares	10
3.1 Não-linearidade por Métodos Gráficos.....	11
3.2 Teste de Keenan para Não-linearidade.....	13
3.3 Teste da Não-linearidade Threshold	14
4. Modelos Threshold	15
4.1 Estimação	16
4.2 Diagnóstico	17
4.3 Previsão	18
5. Aplicação	20
5.1 Dados Meteorológicos	20
5.1.1 Análise Descritiva.....	20
5.1.2 Verificação de não-linearidade	22
5.1.3 Estimação do Modelo.....	24
5.1.4 Diagnóstico	25
5.1.5 Previsão	28
5.2 Dados Financeiros.....	30
5.2.1 Análise Descritiva.....	30
5.1.2 Verificação de não-linearidade	33
5.1.3 Estimação do Modelo.....	35
5.1.4 Diagnóstico	36
6. Conclusão.....	38
7. Referências Bibliográficas	39
A – Programação: simulações.....	41

B – Programação: temperaturas	46
C – Programação: câmbio.....	53

1. Introdução

Explorar o comportamento de fenômenos ao longo do tempo tem sido objeto de estudo de diversos pesquisadores no último século. A análise de dados de qualquer natureza ordenados no tempo constituem o escopo da análise de séries temporais. Alguns exemplos desses tipos de séries são:

- avaliação do preço de um ativo financeiro em uma bolsa de valor;
- medição mensal de índices pluviométricos;
- medição diária do comportamento de marés;
- análise do número de mortes de uma determinada doença em um período de tempo.

Além de descrever o comportamento de determinado fenômeno no tempo, um dos principais objetivos no estudo de séries temporais é poder fornecer previsões futuras para o fenômeno analisado. Para isso existem diversas técnicas estatísticas que fundamentam o estudo desses tipos de série.

Esse trabalho se limita ao estudo de uma técnica aplicada a séries que não possuem um comportamento linear no tempo: os modelos TAR (Threshold Autoregressive Model) também conhecidos como modelos auto-regressivos threshold, ou ainda modelos auto-regressivos com limiares.

Os modelos TAR constituem uma boa saída para situações em que se estuda fenômenos com quebra de regime. Um bom exemplo dessa situação é o comportamento de um ativo financeiro em uma bolsa de valores em períodos de crise ou de grande inconstância do mercado financeiro.

O objetivo deste trabalho é fazer uma abordagem direta e prática sobre o que consistem os modelos auto-regressivos Threshold e abordar um exemplo de seu uso em um caso real. Mais especificamente, o texto objetiva aplicar a metodologia dos modelos TAR para uma base de dados meteorológica e uma financeira, bem como estimar os parâmetros do modelo por meio de recursos computacionais.

Esta monografia se organiza da seguinte maneira. O Capítulo 2 apresenta de modo sucinto os elementos metodológicos que se referem a processos

estocásticos, a definição de um processo não linear, os modelos auto-regressivo e os modelos da classe TAR. O Capítulo 3 apresenta alguns métodos para detecção de não linearidade e o Capítulo 4 discute alguns aspectos relativos a estimação, diagnósticos e previsão. Dois exemplos de aplicações são desenvolvidos e apresentados no Capítulo 5, e a conclusão se encontra no Capítulo 6. Os anexos apresentam os cálculos computacionais que julgamos ser interessantes para que o leitor possa reproduzir boa parte dos resultados mostrados neste trabalho.

Cabe destacar que este texto não descreve profundamente os aspectos teóricos em que os modelos TAR se baseiam e, por isso, sugere-se que o leitor interessado nesse assunto consulte os trabalhos de Tong (1993) e Morettin e Toloí (2006).

Além disso, vale ressaltar que todas as análises serão conduzidas com o apoio do software estatístico *R* versão 3.1.0.

2. Metodologia

2.1 Processos estocásticos

Seja T um conjunto arbitrário. Morettin e Tolo (2006) definem um processo estocástico como sendo uma família $Z = \{Z(t), t \in T\}$, tal que, para cada $t \in T$, $Z(t)$ é uma variável aleatória. Entende-se, portanto, que um processo estocástico seja uma família de variáveis aleatórias definidas em um mesmo espaço de probabilidade $(\Omega, \mathcal{A}, \mathcal{P})$. Tais processos servem de modelos para séries temporais.

De acordo com Brockwell e Davis (2002), séries temporais são constituídas de observações de uma variável aleatória analisadas em um período de tempo específico t .

2.2 Séries temporais lineares e não-lineares

Uma série temporal z_t é dita linear se puder ser escrita na forma

$$z_t = \mu + \sum_{i=1}^n \psi_i a_{t-i} \quad (1)$$

Onde μ é a média do processo z_t ; ψ_i são números reais com $\psi_0 = 1$ e $\{a_t\}$ é uma sequência de variáveis aleatórias independentes e identicamente distribuídas que possuem uma distribuição de probabilidade bem definida. Além disso, assume-se que a distribuição dos a_t é contínua e que $E(a_t) = 0$ e $Var(a_t) = \sigma_a^2$.

Qualquer série temporal que não puder ser escrita da forma da equação 1 é dita ser uma série temporal não-linear. O estudo de séries temporais é repleto de modelos não-lineares. Dentre eles destacam-se os modelos da família ARCH-

GARCH, modelos de volatilidade estocástica e os modelos que serão descritos nesse estudo, os modelos auto-regressivos threshold (TAR).

2.3 Modelos Auto-regressivos

Segundo Franses e van Dijk (2000) e Morettin e Toloí (2006), um modelo auto-regressivo para uma variável y_t consiste em uma combinação linear dos valores dessa variável para $t - p$ valores defasados no tempo. O modelo abaixo exemplifica o que é um modelo auto-regressivo:

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \varepsilon_t, \quad t = 1, \dots, n \quad (2)$$

Os modelos auto-regressivos constituem uma classe de modelos bastante abrangentes e usuais no estudo de séries temporais. Este texto se limitará a uma classe específica de modelos auto-regressivos para séries temporais não-lineares: os modelos TAR.

2.4 Modelos TAR

O modelo TAR (Threshold Autoregressive Model), também conhecido como modelo auto-regressivo com limiar, foi inicialmente proposto por Tong (1980) e extensivamente estudado por Tong e Lim (1980) e Franses e van Dijk (2000). Esses modelos são bastante úteis para processos em que ocorrem trocas de regimes. Dito em outras palavras, o modelo TAR parte da ideia de aproximar uma função não-linear geral para uma função linear por partes.

A formulação do modelo ocorre da seguinte maneira: seja y_t o processo de uma série temporal, z_t uma variável de estado unidimensional e $R_i, i = 1, \dots, k$ uma partição de \mathbb{R} . O modelo TAR pode ser escrito então como a composição de L modelos auto-regressivos da seguinte forma:

$$y_t = \begin{cases} \phi_0^{(1)} + \sum_{i=1}^{k_1} \phi_i^{(1)} y_{t-i} + \varepsilon_t^{(1)}, & \text{se } z_t \in \mathbb{R}^{(1)} \\ \phi_0^{(2)} + \sum_{i=1}^{k_2} \phi_i^{(2)} y_{t-i} + \varepsilon_t^{(2)}, & \text{se } z_t \in \mathbb{R}^{(2)} \\ \vdots \\ \phi_0^{(L)} + \sum_{i=1}^{k_L} \phi_i^{(L)} y_{t-i} + \varepsilon_t^{(L)}, & \text{se } z_t \in \mathbb{R}^{(3)} \end{cases} \quad (3)$$

O valor Z_t é escolhido de acordo com os dados trabalhados e é bastante flexível. Em razão dessa flexibilidade, existe uma vasta possibilidade na escolha de modelos auto-regressivos com limiar para uma determinada situação.

2.4.1 Modelos SETAR

O modelo SETAR (Self Excited Threshold Model) - ou modelo auto-regressivo com limiar auto-excitante – é uma extensão do modelo TAR para $z_t = y_{t-d}$, em que d é um parâmetro de defasamento. O modelo SETAR pode ser escrito então da seguinte maneira:

$$y_t = \begin{cases} \phi_0^{(1)} + \sum_{i=1}^{k_1} \phi_i^{(1)} y_{t-i} + \varepsilon_t^{(1)}, & \text{se } y_{t-d} \in \mathbb{R}^{(1)} \\ \phi_0^{(2)} + \sum_{i=1}^{k_2} \phi_i^{(2)} y_{t-i} + \varepsilon_t^{(2)}, & \text{se } y_{t-d} \in \mathbb{R}^{(2)} \\ \vdots \\ \phi_0^{(L)} + \sum_{i=1}^{k_L} \phi_i^{(L)} y_{t-i} + \varepsilon_t^{(L)}, & \text{se } y_{t-d} \in \mathbb{R}^{(3)} \end{cases} \quad (4)$$

3. Séries Temporais Não-lineares

Conforme exposto na metodologia, uma série temporal é linear se pode ser escrita da forma:

$$z_t = \mu + \sum_{i=1}^n \psi_i a_{t-i} \quad (5)$$

De acordo com Cryer e Chan (2008), séries temporais lineares são essencialmente modelos lineares cuja média condicional é uma função linear dos valores presentes e passados de uma variável avaliada no tempo. Os modelos da família ARIMA são um grande exemplo de modelos para séries lineares.

Os métodos para séries temporais lineares tem se mostrado bastante úteis na prática. Porém tais métodos possuem algumas limitações. Para exemplificar considere a seguinte situação: sabe-se que um processo estacionário normal é caracterizado principalmente pela sua média e sua função de autocovariância. Dessa maneira, tal processo revertido no tempo possui a mesma distribuição do processo original. Essa propriedade é chamada de reversibilidade no tempo.

Entretanto, muitos processos reais se mostram irreversíveis no tempo. Um exemplo disso são preços de fechamento diários de ações. Tais preços de fechamento, em geral, aumentam gradativamente. Porém se há um processo de quebra ou uma crise repentina, os preços podem sofrer um aumento ou queda súbita e substancial em relação ao histórico do processo. Esse cenário exemplifica um mecanismo de dados irreversíveis no tempo.

Sendo assim, para situações como a explanada, processos não-lineares exemplificam melhor o comportamento de uma série.

A avaliação da não-linearidade de um processo pode ser feita por métodos gráficos e comprovada por testes estatísticos.

3.1 Não linearidade por Métodos Gráficos

Em modelos ARIMA é comum assumir que o erro do processo é normal, independente e identicamente distribuído. O pressuposto de normalidade dos erros implica que séries temporais estacionárias também sejam um processo normal. Por exemplo, o par (Y_1, Y_2) possui uma distribuição normal bivariada assim como qualquer par de Y ; a tripla (Y_1, Y_2, Y_3) possui uma distribuição normal trivariada assim como qualquer tripla de Y e assim por diante.

Cryer e Chan (2008) explicam que se o pressuposto da normalidade dos erros é mantido, então uma série temporal não-linear não é normalmente distribuída. A não-linearidade pode ser explorada ao verificar se um número finito de observações de uma série temporal é conjuntamente normal.

Uma saída gráfica para verificar a linearidade de um processo consiste em analisar diagramas de dispersão de Y_t por Y_{t-1} ou Y_{t-2} , e assim por diante. Os dados no diagrama devem seguir um padrão linear, ou seja, estarem dispostos em forma de uma nuvem elíptica com uma densidade decrescente em torno do seu centro. Diagramas que sugerem um afastamento desse padrão sugerem a não-linearidade do processo. Observe as figuras abaixo:

Figura 1: Diagramas de dispersão dos lags 1 ao 6 de um processo ARIMA (1,1,0) simulado com $\phi_1 = 0,7$ e $\sigma^2 = 4$

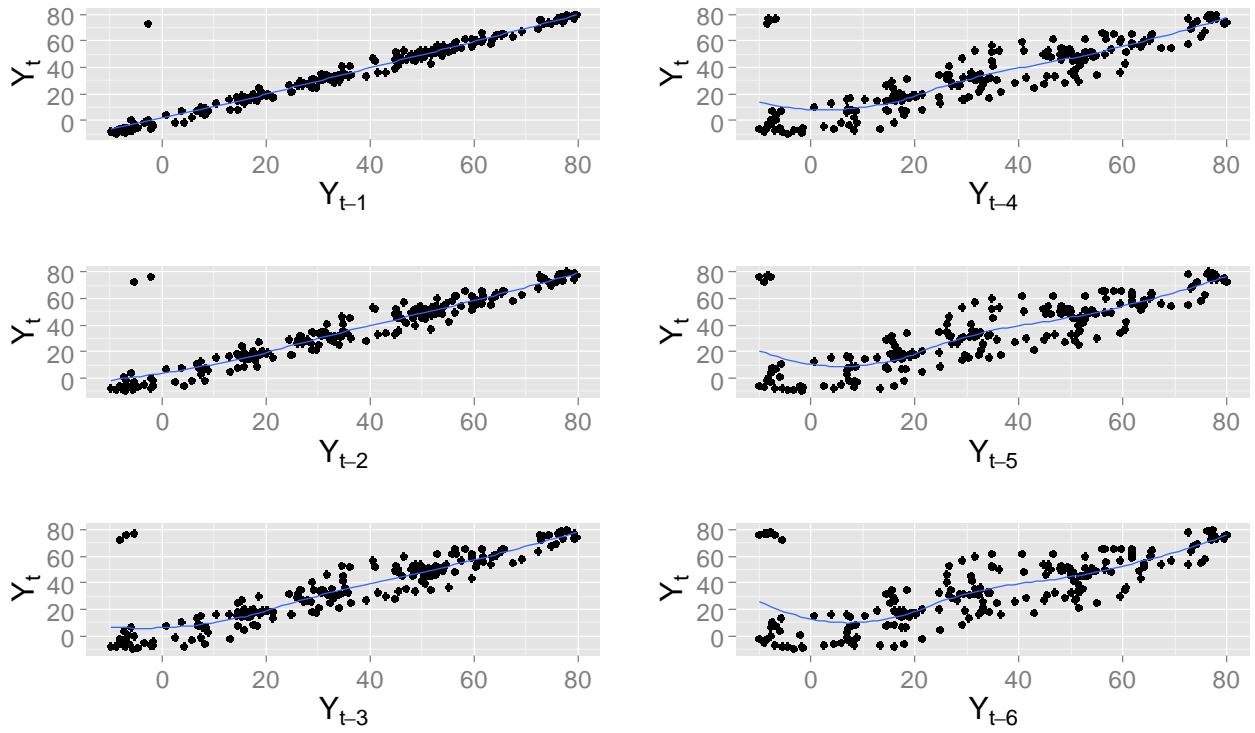
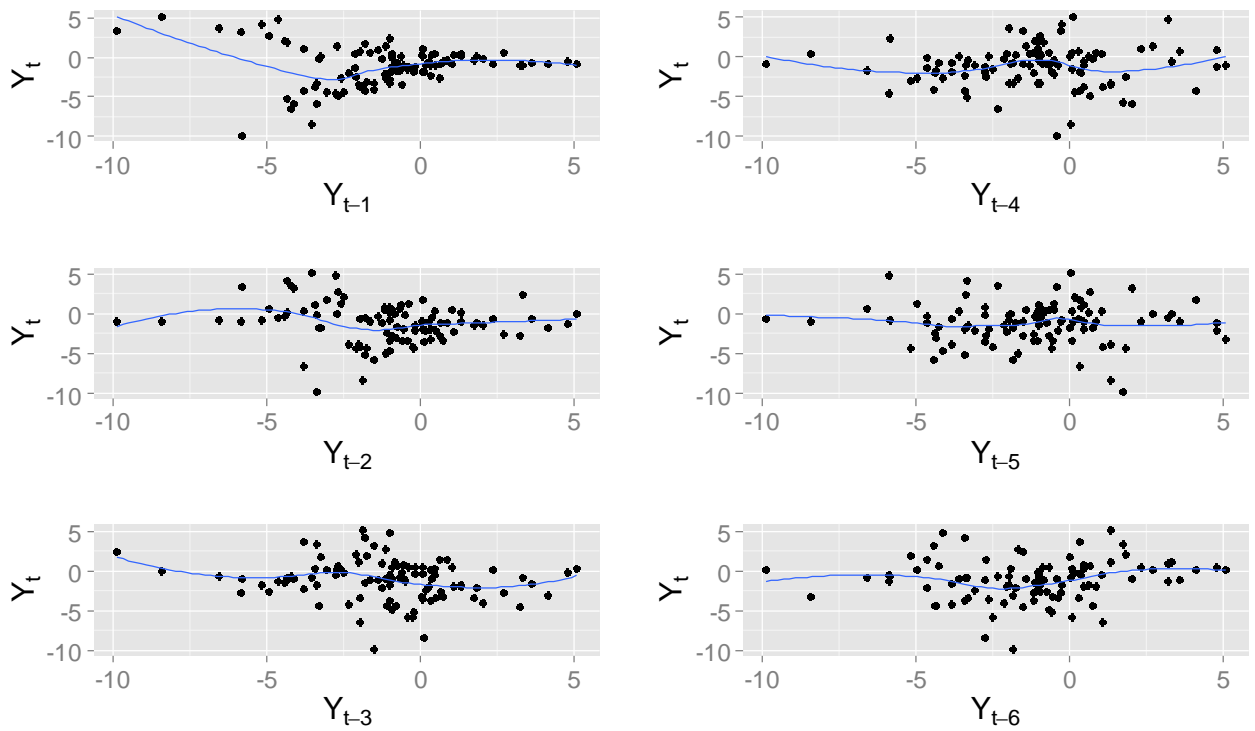


Figura 2: Diagramas de dispersão dos lags de 1 a 6 de um processo TAR(2,1,1) simulado com $\phi_{1,1} = 0,5$ e $\phi_{2,1} = -1,8$



A figura 1 indica o diagrama de dispersão de um processo ARIMA (1,1,0) com $\phi_1 = 0,7$. Os gráficos dos lags 1 a 6 indicam que a série temporal analisada possui um caráter linear. Na figura 2, percebe-se que o processo TAR simulado não apresenta um caráter linear. Isso se dá pelo fato dos diagramas de dispersão dos lags 1 a 6 não estarem dispostos em torno de uma reta, como ocorre na figura 1.

3.2 Teste de Keenan para Não-linearidade

Diversos testes já foram implementados para analisar a linearidade de uma série temporal. Dentre eles, o teste de Keenan consiste em uma boa alternativa para avaliar se um processo requer uma modelagem de séries temporais não-lineares como destaca Keenan (1985). A hipótese nula do teste é de que o processo analisado seja linear a siga algum processo auto-regressivo.

Seja Y_1, \dots, Y_n as observações. O teste de Keenan pode ser implementado da seguinte maneira:

- i. Efetue a regressão de Y_t sobre Y_{t-1}, \dots, Y_{t-m} incluindo o intercepto, onde m seja um número inteiro pré-estabelecido. Calcule os valores preditos $\{\hat{Y}_t\}$ e os resíduos $\{\hat{e}_t\}$ para $t = m + 1, \dots, n$; e seja $RSS = \sum \hat{e}_t^2$ a soma de quadrados dos resíduos.
- ii. Faça a regressão Y_t^2 sobre Y_{t-1}, \dots, Y_{t-m} incluindo o intercepto e calcule os resíduos $\{\hat{\xi}_t\}$ para $t = m + 1, \dots, n$.
- iii. Efetue a regressão de \hat{e}_t sobre os resíduos $\hat{\xi}_t$ sem incluir o intercepto para $t = m + 1, \dots, n$. A estatística do teste de Keenan, denotada por \hat{F} , é obtida ao se multiplicar $(n - 2m - 2)/(n - m - 1)$ pela estatística F obtida ao se testar que a última função de regressão seja igual a zero.

Considere agora:

$$\eta = \eta_0 \sqrt{\sum_{t=m+1}^n \hat{\xi}_t^2}, \quad (6)$$

em que η_0 é o coeficiente de regressão. A estatística do teste é dada pela seguinte expressão:

$$\hat{F} = \frac{\eta^2(n - 2m - 2)}{RSS - \eta^2}. \quad (7)$$

3.3 Teste da Não linearidade Threshold

Os testes que permitem concluir a não-linearidade de um processo, em geral não são sensíveis para detectar se a não-linearidade está associada a um modelo com limiar (modelo Threshold). Tsay (2010) e Cryer e Chan (2008) apresentam uma saída para situações em que se deseja verificar se uma determinada série seja não-linear e esteja associada a um modelo com limiar.

O teste da não-linearidade Threshold é basicamente um teste da razão de verossimilhança com um modelo Threshold como uma alternativa específica. A hipótese nula é de que o processo seja um modelo $AR(p)$ contra a hipótese alternativa de que o processo seja um modelo Threshold de dois regimes de ordem p com variância dos ruídos constante (ou seja, $\sigma_1 = \sigma_2 = \sigma$).

A estatística do teste é a seguinte:

$$T_n = (n - p) \log \left\{ \frac{\hat{\sigma}^2(H_0)}{\hat{\sigma}^2(H_1)} \right\}, \quad (8)$$

Em que $(n - p)$ é o tamanho efetivo da amostra, $\hat{\sigma}^2(H_0)$ é o estimador de máxima verossimilhança da variância dos ruídos do processo $AR(p)$ e $\hat{\sigma}^2(H_1)$ é o estimador de máxima verossimilhança do modelo Threshold com limiar pré fixado em algum intervalo finito.

4. Modelos Threshold

Conforme explanado na metodologia, a essência dos modelos Threshold é descrever um modelo não-linear por meio de submodelos lineares. Dessa forma, por seguir o princípio da modelagem por partes, os modelos com limiares constituem a classe mais simples de modelos não-lineares para séries temporais, como destaca Cryer e Chan (2008).

A especificação dos modelos Threshold se baseia em se determinar a quantidade de submodelos (ou regimes) a serem utilizados conjuntamente à um mecanismo que dite quando esses submodelos são operacionais.

No modelo proposto por Tong, o que dita qual regime será analisado é simplesmente a posição da variável de estado em relação ao limiar em questão. Para exemplificar considere o seguinte modelo TAR de primeira ordem com dois regimes:

$$y_t = \begin{cases} \phi_{1,0} + \phi_{1,1}y_{t-1} + \varepsilon_t & \text{se } y_{t-1} \leq r \\ \phi_{2,0} + \phi_{2,1}y_{t-1} + \varepsilon_t & \text{se } y_{t-1} > r \end{cases} \quad (9)$$

Na situação acima os ϕ 's são os parâmetros autoregressivos, y_{t-1} é a variável de estado, r é o parâmetro de limiar (ou Threshold) e $\{\varepsilon_t\}$ é o erro associado ao modelo. Pode-se observar que para valores de y_{t-1} menores ou iguais a r , o regime superior que irá descrever o processo; para valores de y_{t-1} superiores a r , o regime será descrito pelo regime inferior.

O modelo TAR pode ser estendido para ordens maiores e com a variável de estado associada à um parâmetro de defasamento. Na literatura esses modelos são chamados de modelos SETAR (Self Excited Threshold Model) - ou modelo auto-regressivo com limiar auto-excitante. Observe:

$$y_t = \begin{cases} \phi_{1,0} + \phi_{1,1}y_{t-1} + \dots + \phi_{1,p_1}y_{t-p_1} + \varepsilon_t & \text{se } y_{t-d} \leq r, \\ \phi_{2,0} + \phi_{2,1}y_{t-1} + \dots + \phi_{2,p_2}y_{t-p_2} + \varepsilon_t & \text{se } y_{t-d} > r. \end{cases} \quad (10)$$

Cryer e Chan (2008) destacam que as ordens p_1 e p_2 de cada regime não precisam ser iguais. Além disso, o parâmetro de defasamento d pode ser maior do que o máximo das ordens de cada regime. O modelo definido pela equação acima é denotado como um modelo TAR(2; p_1, p_2) com defasamento d .

4.1 Estimação

Em razão da distribuição estacionária de um modelo TAR não ter uma fórmula fechada e uma solução direta, a estimação dos parâmetros de um modelo Threshold é condicionada à otimização dos parâmetros p, d e r . Onde p é a ordem do processo, d é o parâmetro de defasamento e r é o limiar do modelo.

Para modelos lineares ARMA é comum fazer a estimação dos parâmetros autoregressivos por meio da minimização do critério de informação de Akaike (AIC), conforme destaca Cryer e Chan (2008). Para valores de r e d fixos, a estimação dos parâmetros de um modelo TAR se dá essencialmente em modelar dois modelos AR de ordens p_1 e p_2 , de forma que o AIC seja:

$$AIC(p_1, p_2, r, d) = -2l(r, d) + 2(p_1 + p_2 + 2), \quad (11)$$

Em que o número de parâmetros, excluindo r, d, σ_1 e σ_2 seja igual a $p_1 + p_2 + 2$; e $l(r, d)$ é o logaritmo natural da função de máxima verossimilhança do modelo.

O método de estimação proposto calcula os parâmetros do modelo ao se minimizar o AIC dado acima. Essa minimização deve seguir a restrição de que o parâmetro Threshold (ou parâmetro de limiar) seja procurado em um intervalo que garanta que ambos os regimes possuam dados suficientes para a estimação do modelo.

Cryer e Chan (2008) explicam que adicionar 2 ao critério de Akaike proposto se baseia na simples ideia de tratar o limiar do modelo como um parâmetro adicional. Tal fato adiciona um grau de liberdade a mais no modelo. Sendo assim, Tong (1990) defende que essa adição de um grau de liberdade é assintoticamente equivalente ao se adicionar três graus de liberdade ao modelo.

4.2 Diagnóstico

O diagnóstico para modelos TAR se dá essencialmente em um tripé: análise de resíduos, análise da função de autocorrelação dos resíduos e na verificação se os resíduos são ruído branco por métodos inferenciais.

Os resíduos são definidos como a diferença entre os valores dos dados e os valores preditos do modelo estimado. Ou seja, para um modelo TAR temos:

$$\hat{\varepsilon}_t = Y_t - \{\hat{\phi}_{1,0} + \hat{\phi}_{1,1}Y_{t-1} + \dots + \hat{\phi}_{1,p}Y_{t-\hat{a}}\}I(Y_{t-\hat{a}} \leq \hat{r}) - \{\hat{\phi}_{2,0} + \hat{\phi}_{2,1}Y_{t-1} + \dots + \hat{\phi}_{2,p}Y_{t-\hat{a}}\}I(Y_{t-\hat{a}} > \hat{r}), \quad (12)$$

Em que $I(\cdot)$ é uma variável indicadora que é igual a 1 se, e somente se, a condição expressada for verdadeira e 0 caso contrário.

Uma variação dos resíduos também é utilizada: os resíduos padronizados. A padronização dos resíduos se dá na divisão destes pelo seu apropriado desvio padrão:

$$\hat{e}_t = \frac{\hat{\varepsilon}_t}{\hat{\sigma}_1 I(Y_{t-\hat{a}} \leq \hat{r}) + \hat{\sigma}_2 I(Y_{t-\hat{a}} > \hat{r})}, \quad (13)$$

ou seja, os resíduos do primeiro (ou segundo) regime são normalizados pelo desvio padrão dos ruídos estimados do primeiro (ou segundo) regime.

Assim como no caso linear, o gráfico dos resíduos padronizados deve apresentar um caráter aleatório e tais resíduos devem ser aproximadamente independentes e identicamente distribuídos caso o modelo TAR analisado esteja adequado para a situação em questão. A presença de outliers ou algum outro tipo de padrão no gráfico de resíduos fornece pistas para a especificação de algum outro modelo mais apropriado (CRYER e CHAN, 2008).

O pressuposto de independência dos erros do modelo pode ser verificado ao se analisar a função de autocorrelação (FAC) amostral dos resíduos padronizados. Caso o interesse seja verificar se a variância dos erros não seja constante, analisar a função de autocorrelação amostral do quadrado dos resíduos é indicado (CRYER e CHAN, 2008).

A fim de confirmar a existência de uma estrutura de independência dos resíduos do modelo, o teste de Box-Pierce-Ljung é indicado. As hipóteses do teste consistem nas seguintes:

H_0) *Os resíduos são iid*

H_1) *Os resíduos não são iid*

Sob essas hipóteses o teste é construído com base na estatística a seguir:

$$Q(K) = n(n + 2) \sum_{j=1}^K \frac{\hat{r}_j^2}{(n - j)}, \quad (14)$$

em que:

$$\hat{r}_j^2 = \frac{\sum_{t=k+1}^n \hat{a}_t \hat{a}_{t-k}}{\sum_{t=1}^n \hat{a}_t^2}. \quad (15)$$

E a estatística $Q(K)$ tem distribuição χ^2 com $K - p - q$ graus de liberdade.

4.3 Previsão

Além de tentar descrever o comportamento de um determinado processo, um dos objetos de estudo em séries temporais consiste em fazer previsões para valores futuros do processo analisado.

Uma das dificuldades de se tentar fazer previsões para modelos TAR é fato de não existir um método que forneça uma fórmula fechada para tais estimativas. As previsões para os modelos de séries temporais não-lineares são comumente realizadas por métodos numéricos. Um método simples para prever valores futuros de um modelo Threshold segue o seguinte princípio:

$$Y_{t+1} = h(Y_t, e_{t+1}). \quad (16)$$

Dados $Y_t = y_t, Y_{t-1} = y_{t-1}, \dots$, temos $Y_{t+1} = h(y_t, e_{t+1})$. Então a realização de Y_{t+1} da previsão de um passo à frente pode ser obtida por meio de e_{t+1}

proveniente da distribuição dos erros e computando $h(y_t, e_{t+1})$. Ao se repetir esse processo B vezes, obtém-se uma amostra aleatória de B valores preditos um passo à frente. A média desses B valores constitui numa estimativa para a previsão do próximo valor do processo.

O método de previsão previamente exposto é bastante simples e limitado. Alternativas mais rebuscadas podem ser encontradas em Franses e Dijk (2000, pg 117).

5. Aplicação

A técnica de modelagem de séries temporais com modelos auto-regressivos Threshold possui uma grande aplicabilidade em diversas áreas do conhecimento. Será apresentado nesse texto a aplicação da metodologia dos modelos TAR para duas realidades distintas: inicialmente será apresentada uma modelagem para dados meteorológicas e em seguida para dados financeiros.

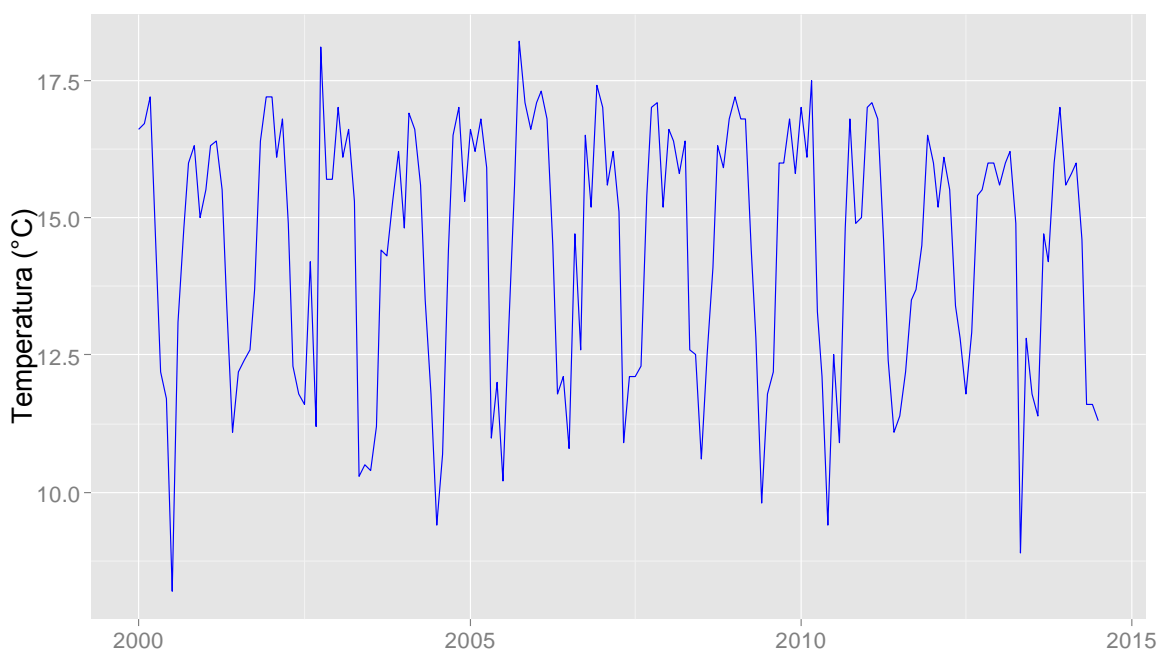
5.1 Dados Meteorológicos

5.1.1 Análise Descritiva

A série temporal analisada foi obtida no Banco de Dados Meteorológicos para Ensino e Pesquisa (BDMEP) do Instituto Nacional de Meteorologia (INMET). As informações são referentes às temperaturas mínimas mensais registradas no Distrito Federal entre Janeiro de 2000 e Julho de 2014. Tais dados podem ser facilmente acessados no endereço eletrônico <http://www.inmet.gov.br/portal/index.php?r=bdmep/bdmep> .

Observe o gráfico da série:

Figura 3: Temperaturas mínimas mensais entre janeiro de 2000 e julho de 2014



É de se esperar que os meses que compreendem o inverno no Distrito Federal apresentem os menores índices de temperatura mínima registrados. Com base na série acima, essa suspeita é confirmada. Além disso, é possível perceber uma forte sazonalidade na série. Esse caráter sazonal é composto por ciclos assimétricos e por uma componente cíclica não-linear.

Com o intuito de conhecer um comportamento geral do processo no período analisado, algumas estatísticas descritivas da série foram calculadas:

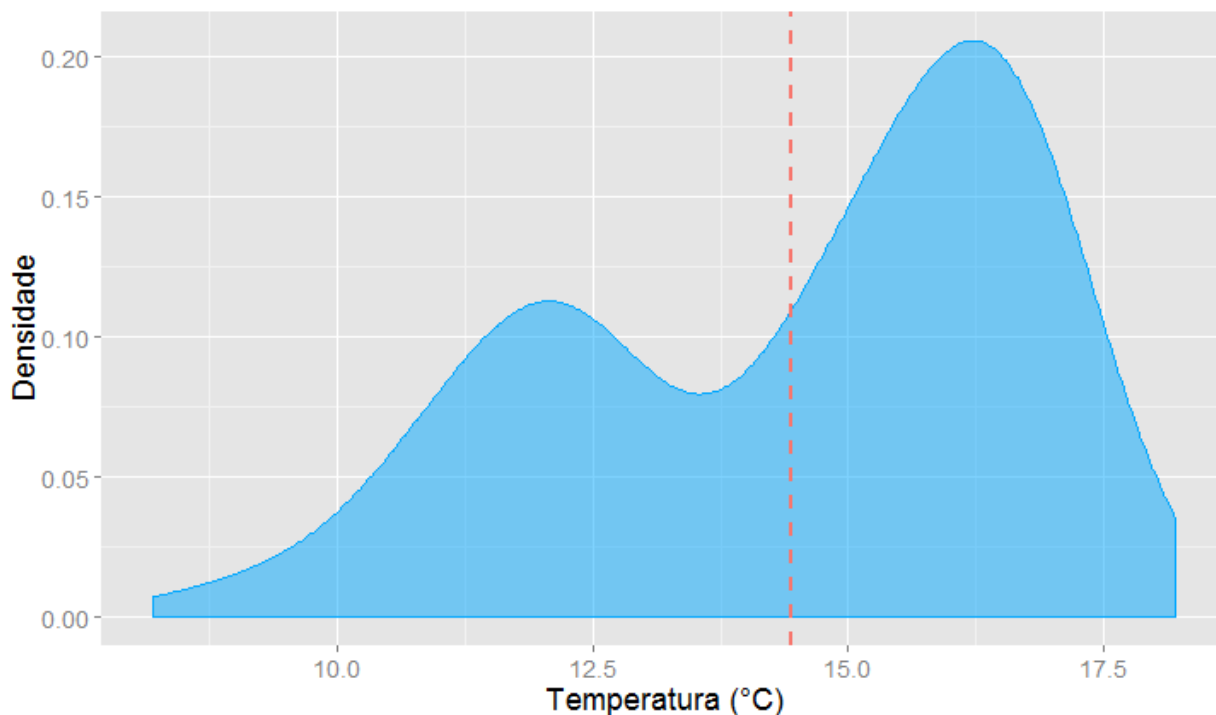
Tabela 1: Estatísticas descritivas da série analisada

Estatística	Valor (°C)
Mínimo	8,20
1º Quartil	12,40
Mediana	15,20
Média	14,45
3º Quartil	16,35
Máximo	18,20

Percebe-se que a menor temperatura mínima do período foi de 8,20°C, a temperatura mínima média foi de 14,45°C e o maior valor para as temperaturas mínimas do DF foi de 18,20°C.

A distribuição das temperaturas mínimas no período em questão é dada da seguinte maneira:

Figura 4: Densidade das temperaturas mínimas no DF entre janeiro de 2000 e julho de 2014



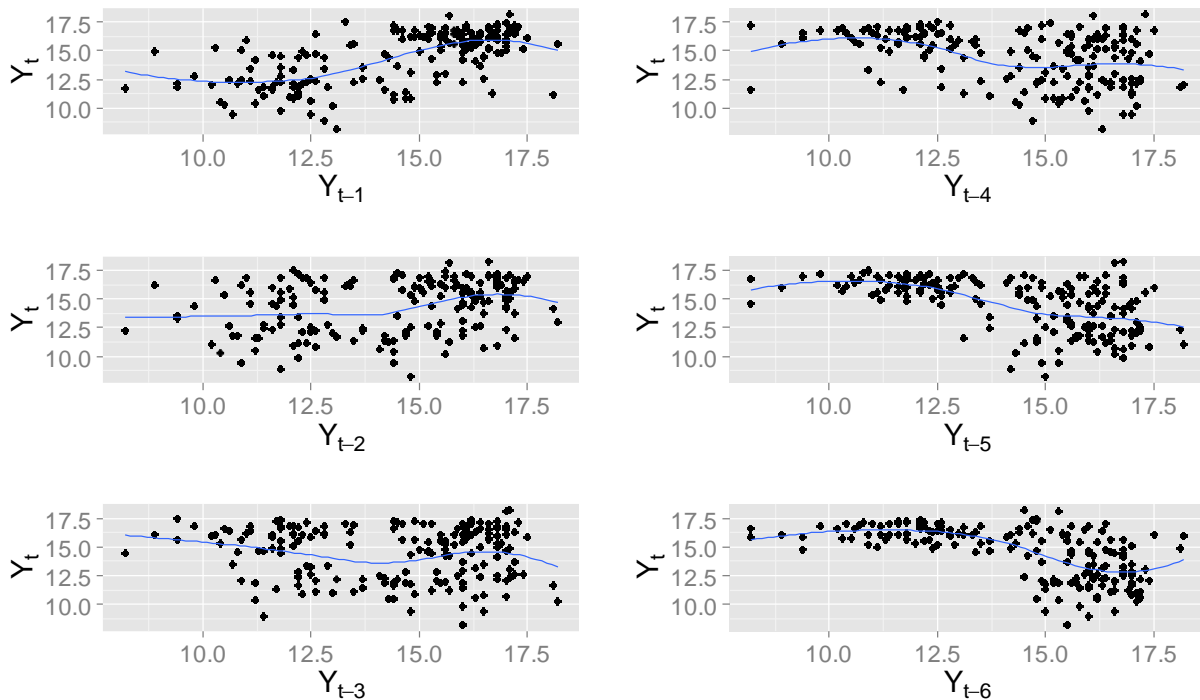
Nota-se a presença de dois picos distintos de temperaturas mínimas: há uma concentração de meses com temperaturas mínimas em torno de 12°C e em seguida uma maior concentração de meses com temperaturas mínimas de aproximadamente 16°C.

A presença desses picos é um indicativo de que o processo possa ser explicado por meio de dois regimes diferentes.

5.1.2 Verificação de não-linearidade

Antes de começar a propor um modelo para a série é necessário avaliar a linearidade do processo. Conforme explanado no capítulo 3, essa análise se dá por meio de métodos gráficos e testes estatísticos. Para a análise gráfica considere os diagramas de dispersão:

Figura 5: Diagramas de dispersão dos lags 1 a 6 da série das temperaturas mínimas do DF



É possível perceber uma não-linearidade acentuada na curva de regressão não-paramétrica dos lags 1,5 e 6. Tal fato é um indicativo de que a série analisada não seja linear. Para confirmar essa suspeita foi realizado o Teste de Keenan para não-linearidade. As hipóteses do teste de Keenan são as seguintes:

H_0) A série é linear e segue algum processo auto – regressivo ;

H_1) A série é não – linear .

A estatística do teste foi 9,9319 e o seu p-valor associado foi de aproximadamente 0,0019. Com esses resultados a hipótese nula é rejeitada para um nível de significância de 5%. Ou seja, pode-se concluir que a série das temperaturas mínimas segue um processo não-linear.

Já que existem evidências suficientes de que o processo seja não-linear, é necessário verificar se um modelo threshold poderia descrever os dados. Para isso, o teste da não-linearidade threshold foi aplicado sob as hipóteses:

H_0) A série segue algum processo $AR(p)$;

H_1) A série segue um modelo TAR com erros iid .

O teste threshold resultou numa estatística de 45,5934 e um p-valor de $6,4635 \cdot 10^{-5}$. Para um nível de significância de 5%, a hipótese nula é rejeitada, portanto conclui-se que existem evidências de que a série das temperaturas mínimas possa ser descrita por um modelo TAR.

Vale ressaltar que esses testes foram computados por meio das funções *keenan.test* e *tlrt* do pacote *TSA* no software livre *R*. Os detalhes da programação estão contidos no anexo ao final desse texto.

5.1.3 Estimação do Modelo

A modelagem da série das temperaturas mínimas no Distrito Federal foi realizada por uma estimação inicial de um modelo TAR (8,4,2). O limiar (threshold) estimado para o processo é de aproximadamente 17,1 sob um parâmetro de defasamento de 6 meses. Dessa forma, os regimes superior e inferior apresentaram as seguintes estimativas:

Tabela 2: Estimativas dos coeficientes do regime inferior

Coeficientes	Estimativa	Erro padrão	Valor t	p-valor
ϕ_0	28,0629	2,5579	10,9710	0,0000
ϕ_1	0,2595	0,0810	3,2031	0,0017
ϕ_2	0,0947	0,0776	1,2212	0,2240
ϕ_3	-0,2921	0,0764	-3,8212	0,0002
ϕ_4	-0,3153	0,0797	-3,9540	0,0001
ϕ_5	-0,0961	0,0785	-0,8809	0,3798
ϕ_6	-0,0349	0,0796	-0,4381	0,6620
ϕ_7	-0,2925	0,0771	-3,7953	0,0002
ϕ_8	-0,2920	0,0773	-3,7797	0,0002

Tabela 3: Estimativas dos coeficientes do regime superior

Coeficientes	Estimativa	Erro padrão	Valor t	p-valor
ϕ_0	14,6599	3,7296	3,9307	0,0111
ϕ_1	0,5061	0,1807	2,8003	0,0380
ϕ_2	0,2044	0,1817	1,1250	0,3117
ϕ_3	-0,3997	0,2863	-1,3960	0,2215
ϕ_4	-0,3072	0,3642	-0,8434	0,4375

Tendo em vista os resultados das tabelas acima, para um nível de confiança de 95% para os coeficientes, o seguinte modelo TAR(1,8,2) é proposto:

$$\hat{y}_t = \begin{cases} 14,6599 + 0,5061y_{t-1} & \text{se } y_{t-6} \leq 17,1 \\ 28,0629 + 0,2595y_{t-1} - 0,2921y_{t-3} & \\ -0,3153y_{t-4} - 0,2925y_{t-7} - 0,2920y_{t-8} & \text{se } y_{t-6} > 17,1. \end{cases} \quad (17)$$

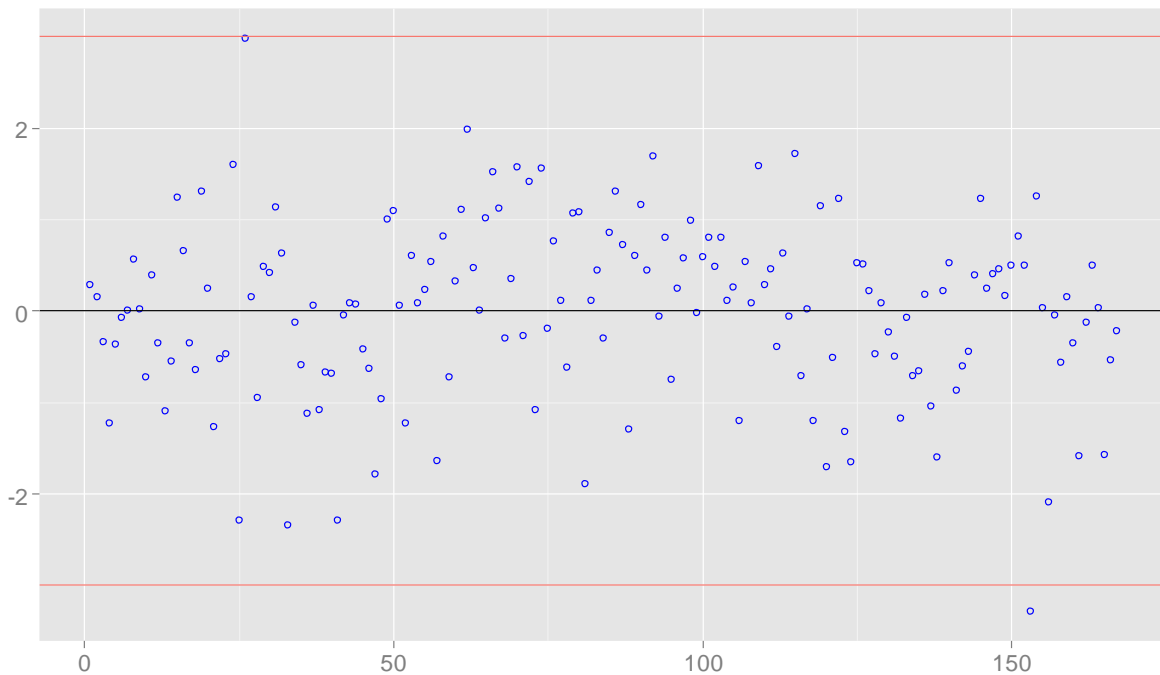
Segue, portanto, da equação 16 que a série das temperaturas mínimas do Distrito Federal entre janeiro de 2000 e julho de 2014 é descrita por dois regimes distintos: para valores de $y_{t-6} \leq 17,1$ o primeiro regime é levado em consideração, para o caso contrário é utilizado o segundo regime.

É importante ressaltar que o parâmetro de defasamento $d = 6$ foi escolhido por um conhecimento empírico. As temperaturas do Distrito Federal em geral se diferenciam em dois grandes blocos: períodos de seca e em períodos chuvosos. O espaço de tempo que comumente separa o final de um período e o início do mesmo período é de 6 meses. Por essa razão foi escolhido o parâmetro de defasamento em questão.

5.1.4 Diagnóstico

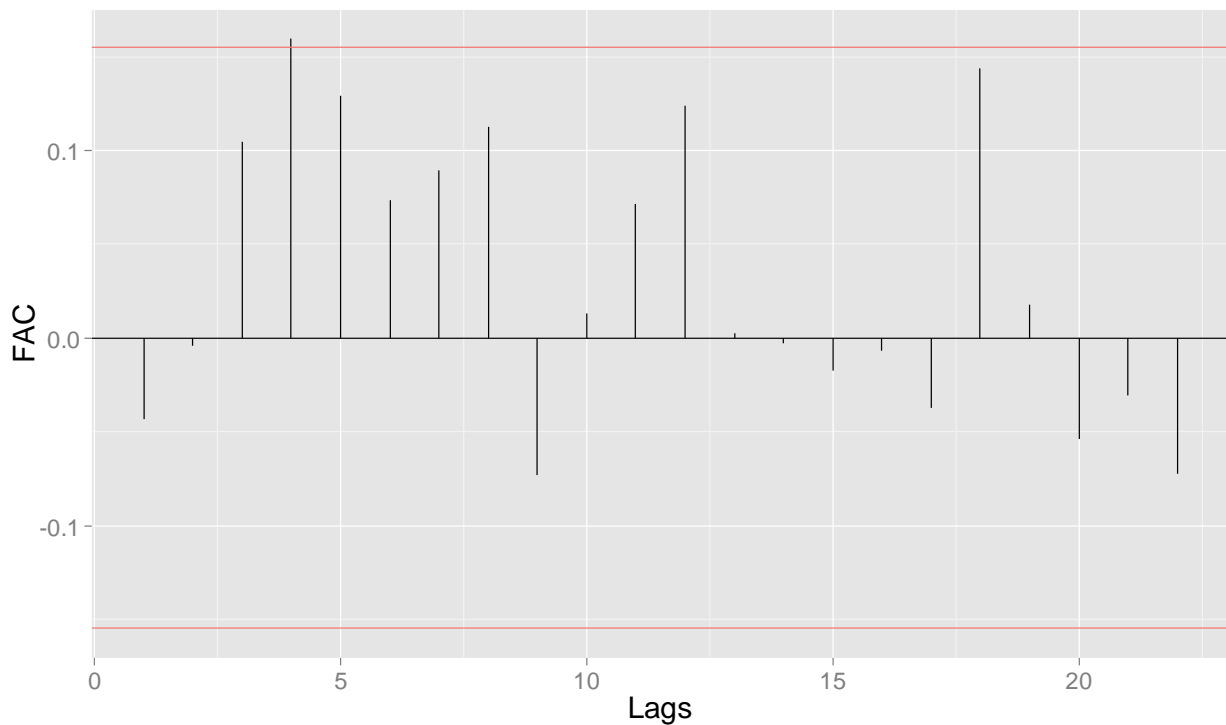
Com o objetivo de testificar a viabilidade do modelo foram realizadas algumas análises de diagnóstico. Primeiramente considere o gráfico dos resíduos padronizados:

Figura 6: Resíduos padronizados do modelo TAR(2,1,8)



O gráfico acima descreve a evolução da série residual. A linhas vermelhas representam as bandas de 95% para os resíduos padronizados. Com base nessas linhas, observam-se dois possíveis pontos discrepantes, um próximo ao tempo 25 e outro próximo ao tempo 150. Para avaliação da correlação serial desses resíduos, aplicaremos o teste de Box-Pierce-Ljung (MORETTIN e TOLOI, 2006), que se baseia na soma dos quadrados das autocorrelações residuais mostradas na Figura 7.”

Figura 7: Função de autocorrelação (FAC) para os resíduos do processo analisado



Nesse gráfico, com exceção do lag 3, os demais lags do gráfico da FAC dos resíduos se concentram dentro das bandas de confiança. A fim de concluir se existe autocorrelação residual no processo, foi realizado o teste de Box-Pierce-Ljung sob as seguintes hipóteses:

H_0) Os resíduos são iid ;

H_1) Os resíduos não são iid .

O p-valor do teste foi de 0,2563. Dessa forma, para um nível de significância de 5% existem evidências suficientes para afirmar que os resíduos da série são independentes e identicamente distribuídos.

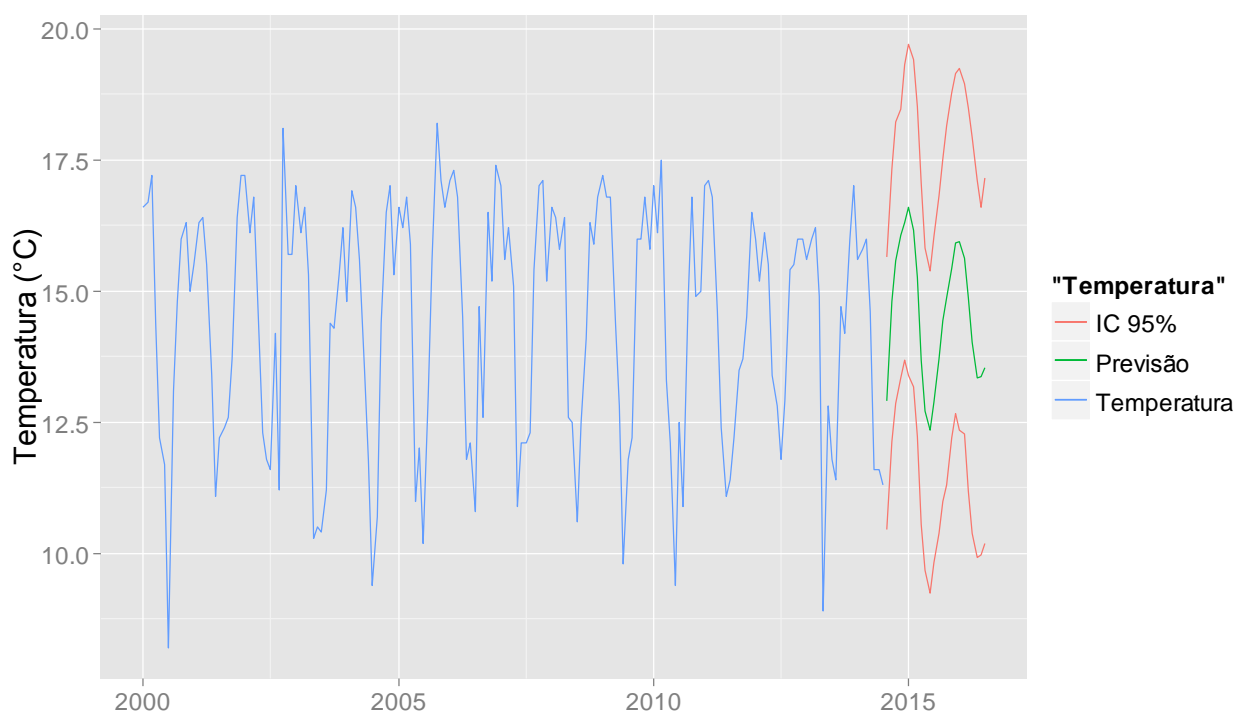
Assim como na etapa de estimação, a análise de diagnóstico do modelo foi realizada no software livre *R*, com o uso do pacote *TSA* sob o uso da função *Box.Test*.

5.1.5 Previsão

Como o modelo TAR(1,8,2) apresentou bons resultados em seus diagnósticos, é interessante verificar a capacidade preditiva desse modelo aos dados propostos.

Com o uso do da função *predict* no pacote *TSA*, foi possível plotar as previsões do modelo para os 24 meses subsequentes à última informação do banco de dados. Ou seja, foram preditas as temperaturas mínimas do Distrito Federal para o período compreendido entre os meses de agosto de 2014 a julho de 2016. O resultado foi o seguinte:

Figura 8: Previsões das temperaturas mínimas do DF para os meses de agosto de 2014 a julho de 2016 sob o modelo TAR(2,1,8) estimado



É possível perceber que as previsões do modelo estimado continuam agregando a sazonalidade vista no conjunto de dados. Entretanto, nota-se claramente que a curva das previsões e do intervalo de 95% de confiança são mais suaves em relação à curva do conjunto de dados. Isso se dá pois o modelo TAR estimado leva em consideração somente dois regimes distintos: um regime explica o aumento das temperaturas mínimas no DF e outro explica a diminuição das mesmas.

Para captar perturbações entre os picos de temperatura, é necessário estimar um modelo TAR com mais de dois regimes. Neste projeto não foi possível estimar um modelo que agregasse mais de dois regimes porque a função TAR do pacote TSA, não dá suporte para a estimação de modelos auto-regressivos threshold com quantidade de regimes superiores a 2.

Uma saída aos modelos TAR para se captar as perturbações entre os picos de temperatura de forma mais eficiente pode ser o ajuste por meio de um modelo ARIMA sazonal (SARIMA). A metodologia e aplicação dos modelos SARIMA não fazem parte do escopo deste texto e podem ser encontradas em Morettin e Tolo (2006, pag. 243).

É importante ressaltar que a realização de previsões de longo prazo não são propriedades dos modelos Threshold. Os modelos TAR apresentam melhores previsões para pequenos intervalos de tempo. Para exemplificar essas afirmações, note como seriam as previsões para as temperaturas mínimas do Distrito Federal para os 120 meses (10 anos) subsequentes à última informação do banco de dados:

Figura 9: Previsões das temperaturas mínimas do DF para os meses de agosto de 2014 a julho de 2024 sob o modelo TAR(2,1,8) estimado



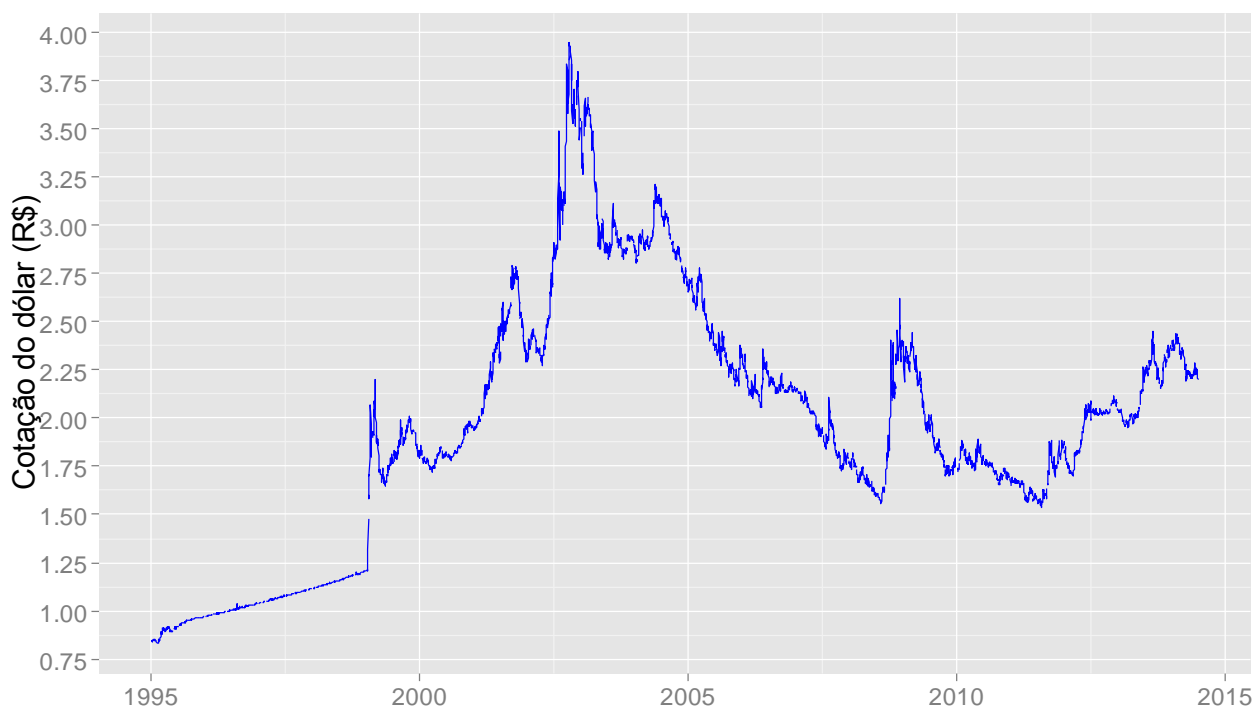
Os valores preditos expostos na figura 9 exemplificam o fato de que os modelos TAR não fornecem boas previsões de longo prazo. Percebe-se claramente no gráfico acima, que os valores preditos em um grande intervalo de tempo perdem amplitude e tendem a assumir o valor médio do processo. Para lidar com esse tipo de situação, recomenda-se a utilização da metodologia de sistemas dinâmicos (TONG, 1993).

5.2 Dados Financeiros

5.2.1 Análise Descritiva

Os dados financeiros analisados são referentes às cotações diárias do dólar comercial em relação ao real entre 02/01/1995 e 01/07/2014. Essas informações foram coletadas no site do Sistema de Reserva Federal do Estados Unidos, ou FED (*Federal Reserve System*). Os dados podem facilmente acessados no seguinte endereço eletrônico: <http://www.federalreserve.gov/datadownload/>.

Figura 10: Cotações diárias do dólar comercial entre janeiro de 1994 e julho de 2014



É possível perceber que o real passou por uma forte desvalorização entre os anos de 1995 e 2002. O final desse período foi marcado por desconfianças do mercado internacional em relação ao novo governo brasileiro. Associado a esse cenário o real atingiu a sua maior cotação ante o dólar (o valor de US\$ 1,00 equivalera a R\$ 3,945).

Entre 2003 e 2008 observou-se a valorização do real, visto que o valor da moeda em relação ao dólar sofreu uma queda expressiva no período. A melhoria de diversos indicadores socioeconômicos do país trouxe um cenário de otimismo para a economia brasileira.

O período de crise iniciado em 2008 motivou uma nova desvalorização da moeda brasileira: as taxas do dólar voltaram a subir. Tal desvalorização não foi tão acentuada como aquela observada no início da década, porém espera-se que as taxas cambiais do dólar sofram um processo de estabilização para os próximos meses.

Para conhecer o comportamento geral da série, algumas estatísticas descritivas podem ser observadas abaixo:

Tabela 4: Estatísticas descritivas da série analisada

Estatística	Valor
Mínimo	0,832
1º Quartil	1,66
Mediana	1,937
Média	1,957
3º Quartil	2,316
Máximo	3,945
Observações faltantes	185

Dentre as estatísticas observadas na tabela acima, percebe-se que a menor cotação do dólar no período foi de R\$ 0,832, a cotação média foi de R\$ 1,957 e a máxima foi de R\$ 3,945.

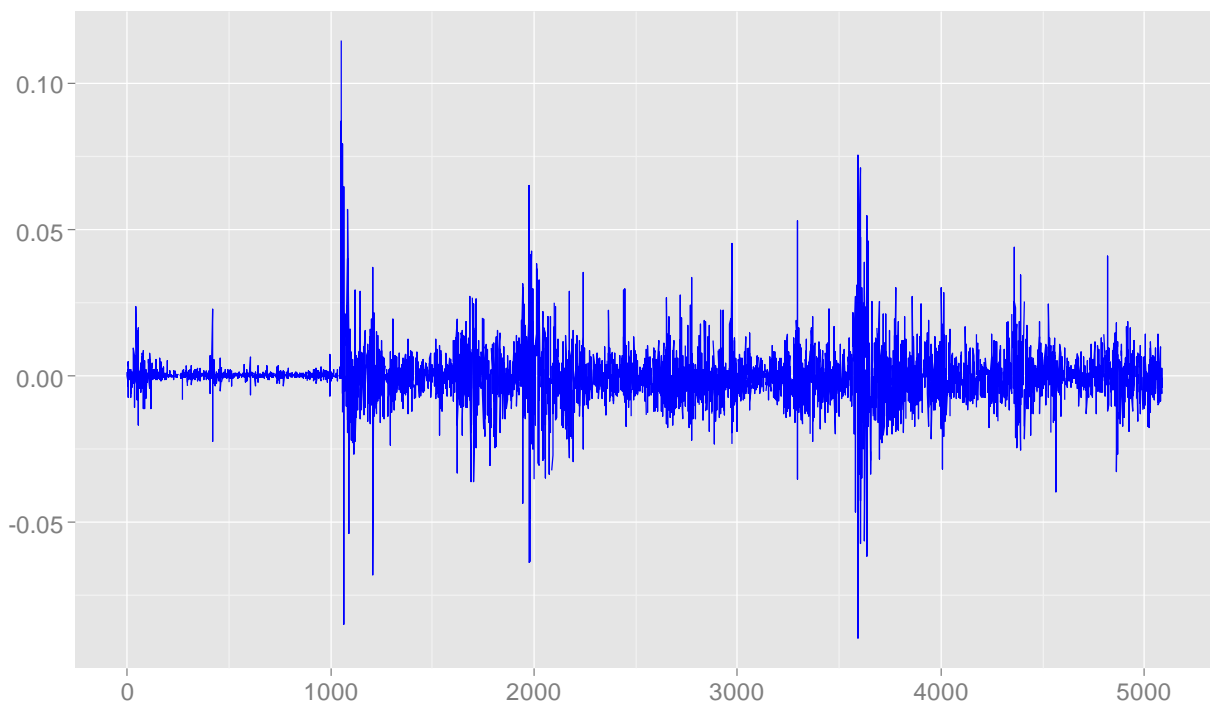
Para deixar a série com um caráter estacionário e facilitar a operacionalização na modelagem, foram utilizados os log-retornos das cotações diárias do dólar.

Seja P_t o preço de um ativo no instante t , Morettin e Tolo (2006) definem um log-retorno r_t , como:

$$r_t = \log\left(\frac{P_t}{P_{t-1}}\right), \quad (18)$$

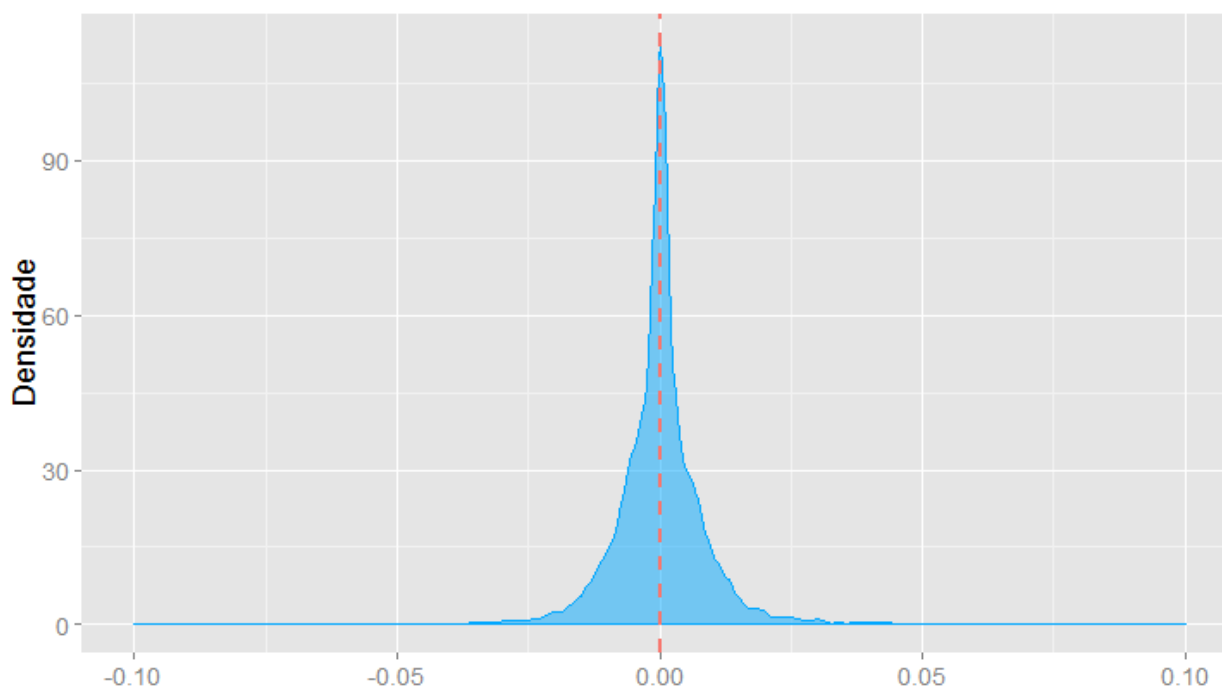
Tendo em vista a definição acima, a série dos log-retornos para os dados analisados é a seguinte:

Figura 11: Log-retornos da série de cotações diárias do dólar



Como esperado, percebe-se que os log-retornos da série concentram-se em torno de zero e apresentam picos de alta variabilidade ao longo do tempo. Para avaliar a distribuição dos log-retornos considere o gráfico abaixo:

Figura 12: Densidade dos log-retornos

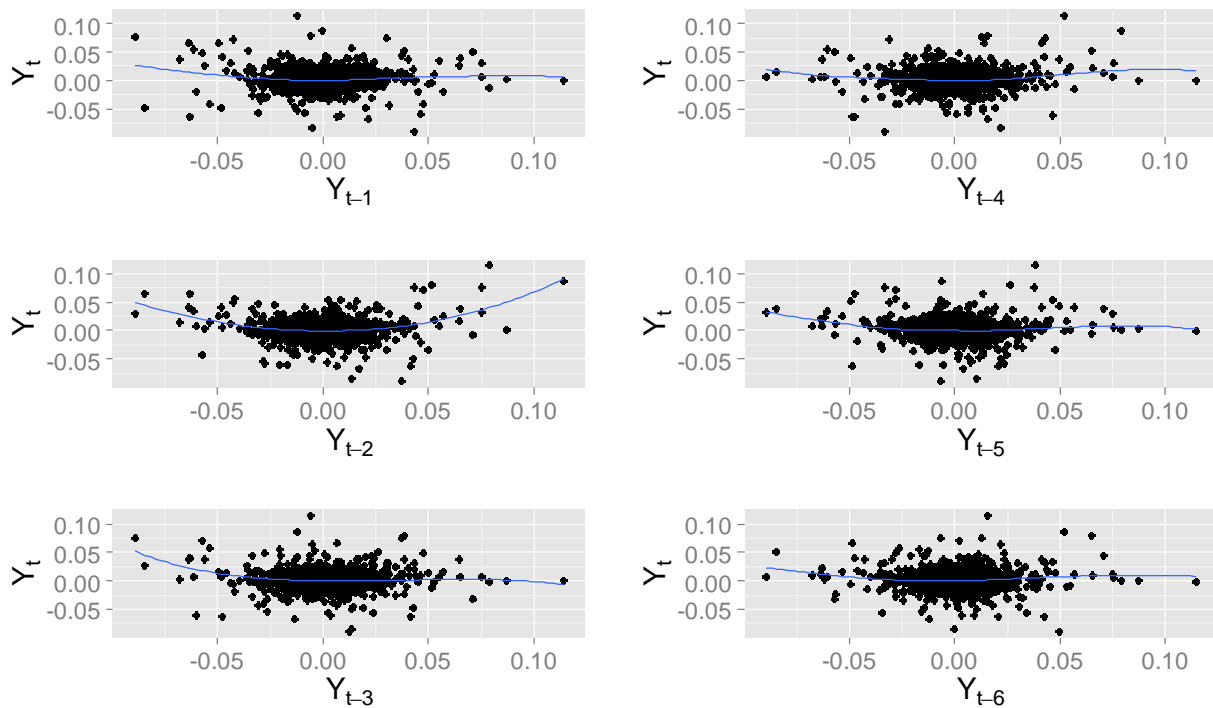


Percebe-se que os log-retornos possuem uma distribuição leptocúrtica, com cauda bastante pesada e com média em zero.

5.1.2 Verificação de não-linearidade

Assim como foi realizado na análise dos dados meteorológicos, é necessário avaliar a linearidade do processo antes de propor um modelo para a série dos log-retornos. Novamente, essa análise se dará por métodos gráficos e inferenciais. Para a análise gráfica considere os diagramas de dispersão:

Figura 13: Diagramas de dispersão dos lags 1 a 6 da série dos log-retornos



Em razão da grande quantidade de observações nesses diagramas, a identificação visual da linearidade do processo não é automática. Embora seja possível perceber uma não linearidade na curva de regressão no diagrama dos lags 2, 3 e 5, ainda não se pode afirmar que o processo seja não-linear. Para confirmar tal suspeita foi realizado o teste de Keenan. As hipóteses do teste de foram as seguintes:

H_0) A série é linear e segue algum processo auto – regressivo ;

H_1) A série é não – linear .

A estatística do teste foi 13,4296 e o seu p-valor associado foi de $2,5044 \times 10^{-5}$. Com esses resultados a hipótese nula é rejeitada para um nível de significância de 5%, ou seja, pode-se concluir que a série dos log-retornos segue um processo não-linear.

Já que existem evidências de que a série seja não-linear é necessário verificar se um modelo threshold poderia descrever os dados. Para isso, o teste da não-linearidade threshold foi aplicado sob as hipóteses:

H_0) A série segue algum processo $AR(p)$;

H_1) A série segue um modelo TAR com erros iid .

O teste threshold resultou numa estatística de 84,5502 e um p-valor de $8,88 \times 10^{-16}$. Para um nível de significância de 5%, a hipótese nula é rejeitada, portanto pode-se concluir que existem evidências de que a série dos log-retornos segue um modelo TAR.

5.1.3 Estimação do Modelo

Foi estimado inicialmente um modelo TAR(4,4,2) para os dados em questão. Note que o primeiro regime possui somente ordem 2. Isso se deu pois o software não computa os coeficientes que são irrelevantes para o modelo. Dessa forma, o modelo proposto foi um TAR(2,2,4). Além disso, o limiar (threshold) estimado para a série dos log-retornos das cotações do dólar é aproximadamente 0,01.

Os regimes inferior e superior apresentaram as seguintes estimativas:

Tabela 5: Estimativas dos coeficientes do regime inferior

Coeficientes	Estimativa	Erro padrão	Valor t	p-valor
ϕ_0	0,0002	0,0001	1,3768	0,1686
ϕ_1	0,0700	0,0177	3,9525	0,0001
ϕ_2	0,0662	0,0144	4,6006	0,0000

Tabela 6: Estimativas dos coeficientes do regime superior

Coeficientes	Estimativa	Erro padrão	Valor t	p-valor
ϕ_0	0,0058	0,0016	3,6509	0,0003
ϕ_1	-0,2812	0,0756	-3,7177	0,0002
ϕ_2	-0,0808	0,0573	-1,4094	0,1595
ϕ_3	0,0032	0,0501	0,0643	0,9488
ϕ_4	0,1550	0,0527	2,9424	0,0034

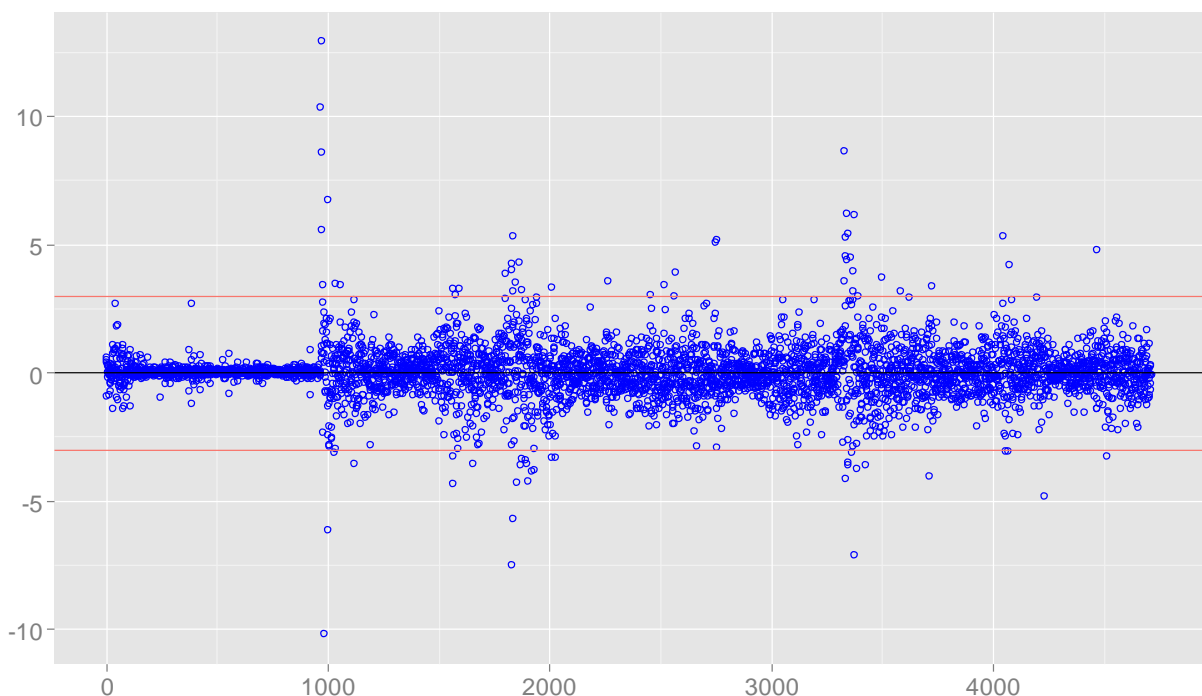
Tendo em vista os resultados das tabelas acima, para um nível de confiança de 95% para os coeficientes, o seguinte modelo é proposto:

$$\hat{y}_t = \begin{cases} 0,07y_{t-1} + 0,0662y_{t-2} & \text{se } y_{t-1} \leq 0,01 \\ 0,0058 - 0,2812y_{t-1} + 0,1550y_{t-4} & \text{se } y_{t-1} > 0,01. \end{cases} \quad (19)$$

5.1.4 Diagnóstico

A fim de verificar a viabilidade do modelo é necessário realizar uma análise de diagnóstico. Para isso considere o gráfico dos resíduos padronizados:

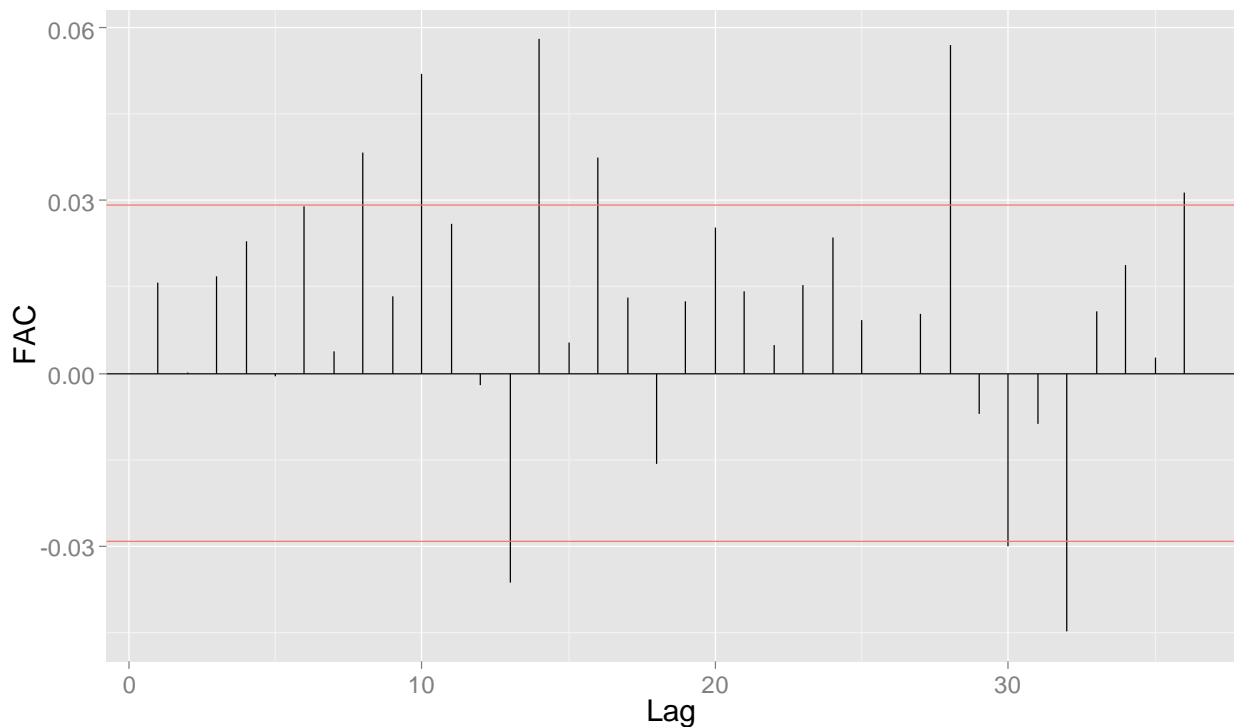
Figura 14: Resíduos padronizados do modelo TAR(2,2,4)



Por meio do gráfico de resíduos padronizados é possível identificar a presença de múltiplos regimes. Nota-se um primeiro regime até o tempo 1000 com baixa variabilidade dos dados. Entre os tempos 1000 e 2000 é possível perceber a presença de um segundo regime. Além disso, percebe-se também um terceiro regime entre os tempos 2000 e meados do tempo 3000. Um quarto regime pode ser notado a partir de meados do tempo 3000 em diante.

Para atestar se os resíduos são ruído branco, a função de autocorrelação amostral dos mesmos foi analisada. Considere o gráfico que se segue:

Figura 15: Função de autocorrelação (FAC) para os resíduos do processo analisado



Segundo o gráfico acima, os resíduos do modelo TAR ajustado não são ruído branco. Isso pode ser afirmado pois em diversos lags a função de autocorrelação se mostra fora das bandas de confiança. Para confirmar tal suspeita foi realizado o teste de Box-Pierce-Ljung sob as seguintes hipóteses:

H_0) Os resíduos são iid ;

H_1) Os resíduos não são iid .

O p-valor do teste foi de aproximadamente 0,0010. Dessa forma, para um nível de significância de 5% existem evidências suficientes para afirmar que os resíduos da série não são independentes e identicamente distribuídos.

Tendo em vista os resultados aqui expostos, o modelo proposto não se ajusta de maneira adequada aos dados.

6. Conclusão

A aplicação prática da metodologia dos modelos auto-regressivos Threshold contemplou dois casos distintos. Dadas as ressalvas explicitadas no texto, foi possível modelar os dados meteorológicos de forma satisfatória. Por outro lado, a utilização de um modelo TAR com dois regimes para o conjunto de dados financeiros se mostrou inadequada.

Os modelos auto-regressivos Threshold propostos por Tong (1983) constituem uma classe de modelos bastante versátil. A possibilidade de se utilizar múltiplos regimes para explicar um determinado processo faz com que essa metodologia seja bastante aplicada em diversas áreas do conhecimento.

É importante ressaltar que toda a metodologia dos modelos Threshold é recente e possui um caráter experimental, não sendo exaustivamente estudada desde a sua formulação.

Além disso, tal metodologia não se restringe para o caso puramente auto-regressivo. A modelagem pode ser generalizada juntamente com a utilização de médias móveis, os modelos TARMA (*Threshold Autoregressive Moving Average*), e também para modelos de volatilidade ARCH e GARCH, gerando os modelos TARARCH e T-GARCH.

7. Referências Bibliográficas

AMARAL, L. F. M. (2003), Modelos lineares e não-lineares na modelagem do preço do spot de energia elétrica no Brasil. Tese (mestrado) – Pontifícia Universidade Católica do Rio de Janeiro.

BROCKWELL, P. J., DAVIS, R. A. (2002), Introduction to Time Series Analysis and Forecasting. New York: Springer.

CRYER, J. D.; CHAN, K. (2008), Time Series Analysis With Applications in R. New York: Springer.

DIJK, D., TERASVIRTA, T., FRANCES, P. H, (2000). Smooth Transition Autoregressive Models – A Survey of Recent Developments, *Econometric Review*, v. 21, n. 1, p. 1-47.

FRANCES, P. H.; DIJK, D. (2000). Non-linear time series models in empirical finance. New York: Cambridge University Press

KEENAN, D. M. (1985). A Tuckey nonadditivity-type test for time series nonlinearity. *Biometrika*, v.72, n. 1, p.39-44.

MORETTIN, P. A., TOLOI, C. M. C. (2006). Análise de séries temporais. Rio de Janeiro: Blucher.

TONG, H. (1983). Threshold Models in Non-linear Times Series Analysis. Lecture Notes in Statistics, 21, Heidelberg: Springer.

TONG, H. (1993). *Non-linear Time Series: A Dynamical System Approach*. Oxford: Oxford University Press.

TONG, H.; LIM, K. S. (1980). Threshold autoregression, limit cycles and cyclical data. *Journal of Statistics Society. Series B (Methodological)*, p. 245-292.

TSAY, R. S. (2010). *Analysis of financial time series. Lecture Notes in Statistics*. Hoboken: John Wiley & Sons.

A – Programação: simulações

```
#####  
#####Simulando um ARIMA (1,1,0) com phi1 = 0,7 e sigma2=4  
#####  
  
ts.sim <- arima.sim(list(order = c(1,1,0), ar = 0.7), n =  
200, sd=sqrt(4))  
ts.plot(ts.sim)  
  
x = seq(from=1, to=201)  
  
ts.sim = as.numeric(ts.sim)  
  
# 1) Verificando a linearidade da série graficamente  
lagplot(ts.sim)  
  
# Fazendo no ggplot2  
library(ggplot2)  
  
#Criando um banco  
ts.sim_lag1 = ts.sim[-1]  
ts.sim_lag2 = ts.sim_lag1[-1]  
ts.sim_lag3 = ts.sim_lag2[-1]  
ts.sim_lag4 = ts.sim_lag3[-1]  
ts.sim_lag5 = ts.sim_lag4[-1]  
ts.sim_lag6 = ts.sim_lag5[-1]  
  
lags = cbind(ts.sim, ts.sim_lag1, ts.sim_lag2, ts.sim_lag3,  
            ts.sim_lag4, ts.sim_lag5, ts.sim_lag6)  
  
lags = as.data.frame(lags)  
  
#Gráficos  
lag1 = ggplot(lags, aes(x=ts.sim_lag1, y=ts.sim)) +  
  geom_point() +      # Use hollow circles  
  geom_smooth(method=loess,se=FALSE) +  
  theme_grey(base_size = 18)+  
  xlab(expression(Y[t-1])) + ylab(expression(Y[t]))  
  
lag2 = ggplot(lags, aes(x=ts.sim_lag2, y=ts.sim)) +  
  geom_point() +      # Use hollow circles  
  geom_smooth(method=loess,se=FALSE) +  
  theme_grey(base_size = 18)+  
  xlab(expression(Y[t-2])) + ylab(expression(Y[t]))  
  
lag3 = ggplot(lags, aes(x=ts.sim_lag3, y=ts.sim)) +  
  geom_point() +      # Use hollow circles
```

```

geom_smooth(method=loess,se=FALSE) +
theme_grey(base_size = 18)+
xlab(expression(Y[t-3])) + ylab(expression(Y[t]))

lag4 = ggplot(lags, aes(x=ts.sim_lag4, y=ts.sim)) +
geom_point() +      # Use hollow circles
geom_smooth(method=loess,se=FALSE) +
theme_grey(base_size = 18)+
xlab(expression(Y[t-4])) + ylab(expression(Y[t]))

lag5 = ggplot(lags, aes(x=ts.sim_lag5, y=ts.sim)) +
geom_point() +      # Use hollow circles
geom_smooth(method=loess,se=FALSE) +
theme_grey(base_size = 18)+
xlab(expression(Y[t-5])) + ylab(expression(Y[t]))

lag6 = ggplot(lags, aes(x=ts.sim_lag6, y=ts.sim)) +
geom_point() +      # Use hollow circles
geom_smooth(method=loess,se=FALSE) +
theme_grey(base_size = 18)+
xlab(expression(Y[t-6])) + ylab(expression(Y[t]))

# Função para juntar os gráficos numa página só
multiplot <- function(..., plotlist=NULL, file, cols=1,
layout=NULL) {
  require(grid)

  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)

  numPlots = length(plots)

  # If layout is NULL, then use 'cols' to determine layout
  if (is.null(layout)) {
    # Make the panel
    # ncol: Number of columns of plots
    # nrow: Number of rows needed, calculated from # of cols
    layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                      ncol = cols, nrow =
ceiling(numPlots/cols))
  }

  if (numPlots==1) {
    print(plots[[1]])
  } else {
    # Set up the page
    grid.newpage()
    pushViewport(viewport(layout =
grid.layout(nrow(layout), ncol(layout))))

```

```

# Make each plot, in the correct location
  for (i in 1:numPlots) {
    # Get the i,j matrix positions of the regions that
    contain this subplot
    matchidx <- as.data.frame(which(layout == i, arr.ind
= TRUE))

    print(plots[[i]], vp = viewport(layout.pos.row =
matchidx$row,
                                  layout.pos.col =
matchidx$col))
  }
}

multiplot(lag1, lag2, lag3, lag4, lag5, lag6, cols=2)

#####
#####Simulando um TAR(2,1,1) com phi1,1= 0,5 e phi2,1= -1,8
#####

set.seed(1234579)
tar_sim=tar.sim(n=100,Phi1=c(0,0.5),
               Phi2=c(0,-1.8),p=1,d=1,
               sigma1=1,thd=-1,sigma2=2)$y
plot(y=tar_sim,x=1:100,type='b',xlab="t",ylab=expression(Y[
t]))

x = seq(from=1, to=201)

tar_sim = as.numeric(tar_sim)

# 1) Verificando a linearidade da série graficamente
lagplot(tar_sim)
# Fazendo no ggplot2
library(ggplot2)

#Criando um banco
tar_sim_lag1 = tar_sim[-1]
tar_sim_lag2 = tar_sim_lag1[-1]
tar_sim_lag3 = tar_sim_lag2[-1]
tar_sim_lag4 = tar_sim_lag3[-1]
tar_sim_lag5 = tar_sim_lag4[-1]
tar_sim_lag6 = tar_sim_lag5[-1]

lags = cbind(tar_sim, tar_sim_lag1, tar_sim_lag2,
tar_sim_lag3,
             tar_sim_lag4, tar_sim_lag5, tar_sim_lag6)

lags = as.data.frame(lags)

```

```

#Gráficos
lag1 = ggplot(lags, aes(x=tar_sim_lag1, y=tar_sim)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-1])) + ylab(expression(Y[t]))

lag2 = ggplot(lags, aes(x=tar_sim_lag2, y=tar_sim)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-2])) + ylab(expression(Y[t]))

lag3 = ggplot(lags, aes(x=tar_sim_lag3, y=tar_sim)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-3])) + ylab(expression(Y[t]))

lag4 = ggplot(lags, aes(x=tar_sim_lag4, y=tar_sim)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-4])) + ylab(expression(Y[t]))

lag5 = ggplot(lags, aes(x=tar_sim_lag5, y=tar_sim)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-5])) + ylab(expression(Y[t]))

lag6 = ggplot(lags, aes(x=tar_sim_lag6, y=tar_sim)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-6])) + ylab(expression(Y[t]))

# Função para juntar os gráficos numa página só
multiplot <- function(..., plotlist=NULL, file, cols=1,
layout=NULL) {
  require(grid)

  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)

  numPlots = length(plots)

  # If layout is NULL, then use 'cols' to determine layout
  if (is.null(layout)) {
    # Make the panel
    # ncol: Number of columns of plots

```

```

    # nrow: Number of rows needed, calculated from # of cols
    layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                    ncol = cols, nrow =
ceiling(numPlots/cols))
  }

  if (numPlots==1) {
    print(plots[[1]])

  } else {
    # Set up the page
    grid.newpage()
    pushViewport(viewport(layout =
grid.layout(nrow(layout), ncol(layout))))

    # Make each plot, in the correct location
    for (i in 1:numPlots) {
      # Get the i,j matrix positions of the regions that
      contain this subplot
      matchidx <- as.data.frame(which(layout == i, arr.ind
= TRUE))

      print(plots[[i]], vp = viewport(layout.pos.row =
matchidx$row,
                                       layout.pos.col =
matchidx$col))
    }
  }
}

multiplot(lag1, lag2, lag3, lag4, lag5, lag6, cols=2)

```

B – Programação: temperaturas

```
library(plyr)
library(gdata)
library(ggplot2)
library(reshape)
library(TSA)
library(tsDyn)

#Importando os dados
setwd("C:\\Users\\Leonardo\\Desktop\\Backup\\UnB\\Monografia\\Dados")

dados = read.csv2(file="tempmin_df4.txt", head=T,
stringsAsFactors=F)

#Transformando dados e mudando suas classes
data=dados$Data
temp=dados$TempMinima

temp=as.numeric(temp)
data=as.Date(dados$Data, "%d/%m/%Y")

class(temp)
class(data)

# Obtendo o máximo, média e mínimo das temperaturas
yyyymm = paste(format(as.POSIXlt(data), format="%Y-%m"),
"01", sep="-")
tempmedia = tapply(temp, yyyymm, mean)
tempmax = tapply(temp, yyyymm, max)
tempmin = tapply(temp, yyyymm, min)

data2 = seq(from=as.Date("2000-01-01"), to=as.Date("2014-
07-01"), by="month")

dados2 = data.frame(data2, tempmedia, tempmax, tempmin)

#####
##### Análise Descritiva #####
#####

#Resumo
summary(tempmin)

#Densidade da série "pura"
ggplot(dados2, aes(x=tempmin)) +
  geom_density(col="#00a7ff", fill="#00a7ff", alpha=0.5) +
```

```

geom_vline(aes(xintercept=14.45, colour="red"), #Média
           linetype="dashed", size=1) +
theme_grey(base_size = 18)+
xlab("Temperatura (°C)") + ylab("Densidade")

#Gráfico da série "pura"
ggplot(dados2, aes(x=data2, y=tempmin)) +
  geom_line(col="blue") +
  theme_grey(base_size = 18)+
  xlab("") + ylab("Temperatura (°C)")

#Componente cíclica não-linear. Ciclos assimétricos

#####
##### LINEARIDADE #####
#####

#Verificando a linearidade da série graficamente
lagplot(tempmin)

# Fazendo no ggplot2

#Criando um banco
tempmin_lag1 = tempmin[-1]
tempmin_lag2 = tempmin_lag1[-1]
tempmin_lag3 = tempmin_lag2[-1]
tempmin_lag4 = tempmin_lag3[-1]
tempmin_lag5 = tempmin_lag4[-1]
tempmin_lag6 = tempmin_lag5[-1]

lags = cbind(tempmin, tempmin_lag1, tempmin_lag2,
tempmin_lag3,
tempmin_lag4, tempmin_lag5, tempmin_lag6)

lags = as.data.frame(lags)

#Gráficos
lag1 = ggplot(lags, aes(x=tempmin_lag1, y=tempmin)) +
  geom_point() + # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-1])) + ylab(expression(Y[t]))

lag2 = ggplot(lags, aes(x=tempmin_lag2, y=tempmin)) +
  geom_point() + # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-2])) + ylab(expression(Y[t]))

```

```

lag3 = ggplot(lags, aes(x=tempmin_lag3, y=tempmin)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-3])) + ylab(expression(Y[t]))

lag4 = ggplot(lags, aes(x=tempmin_lag4, y=tempmin)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-4])) + ylab(expression(Y[t]))

lag5 = ggplot(lags, aes(x=tempmin_lag5, y=tempmin)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-5])) + ylab(expression(Y[t]))

lag6 = ggplot(lags, aes(x=tempmin_lag6, y=tempmin)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess,se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-6])) + ylab(expression(Y[t]))

# Função para juntar os gráficos numa página só
multiplot <- function(..., plotlist=NULL, file, cols=1,
layout=NULL) {
  require(grid)

  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)

  numPlots = length(plots)

  # If layout is NULL, then use 'cols' to determine layout
  if (is.null(layout)) {
    # Make the panel
    # ncol: Number of columns of plots
    # nrow: Number of rows needed, calculated from # of cols
    layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                      ncol = cols, nrow =
ceiling(numPlots/cols))
  }

  if (numPlots==1) {
    print(plots[[1]])
  } else {
    # Set up the page
    grid.newpage()

```



```

    pushViewport(viewport(layout
grid.layout(nrow(layout), ncol(layout))))

    # Make each plot, in the correct location
    for (i in 1:numPlots) {
        # Get the i,j matrix positions of the regions that
        contain this subplot
        matchidx <- as.data.frame(which(layout == i, arr.ind
= TRUE))

        print(plots[[i]], vp = viewport(layout.pos.row =
matchidx$row,
layout.pos.col =
matchidx$col))
    }
}

multiplot(lag1, lag2, lag3, lag4, lag5, lag6, cols=2)

#Teste de Keenan
Keenan.test(tempmin)

#Teste Threshold
tlrt(tempmin, 8, d=6, a=.05, b=.95)

#####
#####  MODELAGEM  #####
#####

#Estimando um modelo TAR
tar_temp = tar(tempmin, p1=8, p2=4, d=6,
a=.05, b=.95, method="CLS", print=T)

setar_temp = setar(tstemp, m=2, steps=1)

#Diagnóstico
tsdiag.TAR(tar_temp)

Box.test(tar_temp$residuals, lag=20, type="Ljung-Box")

#Gráfico dos resíduos
res = tar_temp$residuals
std_res = tar_temp$std.res
seqres = seq(from=1, to=167)

```

```

dados_res = data.frame (std_res,seqres)

ggplot(dados_res, aes(x=seqres, y=std_res)) +
  geom_point(col='blue', shape=1) +
  geom_hline(aes(yintercept=0)) +
  geom_hline(aes(yintercept=3, colour="red")) +
  geom_hline(aes(yintercept=-3, colour="red")) +
  theme_grey(base_size = 18)+
  xlab("") + ylab("")

# Gráfico da FAC dos resíduos
fac = acf(res)

bacf <- acf(res, plot = FALSE)
bacfdf <- with(bacf, data.frame(lag, acf))

q <- ggplot(data = bacfdf, mapping = aes(x = lag, y = acf))
+
  geom_hline(aes(yintercept = 0)) +
  geom_segment(mapping = aes(xend = lag, yend = 0)) +
  geom_hline(aes(yintercept=(2/sqrt(167)), colour="red")) +
  geom_hline(aes(yintercept=-(2/sqrt(167)), colour="red"))+
  theme_grey(base_size = 18)+
  xlab("Lags") + ylab("FAC")

q

##### Previsão
n_dados = 175
n_prev = 24

tar_prev = predict(tar_temp, n.ahead=n_prev, n.sim = 1000)

#Manipulando os dados da previsão
prev1 = t(tar_prev$pred.interval)
prev2 = tar_prev$fit
prev3 = cbind(prev1,prev2)

prev = as.data.frame(prev3)

colnames(prev)[1] = "prevmin"
colnames(prev)[2] = "prevmax"
colnames(prev)[3] = "previsao"

#prev
#row.names$prev = NULL
#prev

```

```

# Criando datas para as previsões
start = as.Date("2014-08-01")
end = as.Date("2016-07-01")
dataprev = seq(from=start, by="month", to=end)

#Criando um vetor com a data dos valores passados e das
previsões
data3 = append(dados2$data2, dataprev)

#Criando um novo banco com os valores passados e valores
preditos
temp2 = rep(NA,n_prev)
prevmin = rep(NA,n_dados)
prevmax = rep(NA,n_dados)
previsao = rep(NA,n_dados)

temp3 = append(tempmin,temp2)

previsoes.na = cbind(prevmin, prevmax, previsao)

previsoes = rbind(previsoes.na,prev)

previsoes2 = cbind(previsoes, temp3, data3)

#gráfico

ggplot(previsoes2, aes(data3)) +
  geom_line(aes(y = temp3, colour = "Temperatura")) +
  geom_line(aes(y = previsao, colour = "Previsão"))+
  geom_line(aes(y = prevmin, colour = "IC 95%"))+
  geom_line(aes(y = prevmax, colour = "IC 95%"))+
  geom_ribbon(aes(ymin = prevmin, ymax = prevmax),fill =
"grey", alpha = 0.4)+
  theme_grey(base_size = 18)+
  xlab("") + ylab("Temperatura (°C)")

##### Previsão longa
n_dados = 175
n_prev = 120

tar_prev = predict(tar_temp, n.ahead=n_prev, n.sim = 1000)

#Manipulando os dados da previsão
prev1 = t(tar_prev$pred.interval)
prev2 = tar_prev$fit
prev3 = cbind(prev1,prev2)

```

```

prev = as.data.frame(prev3)

colnames(prev)[1] = "prevmin"
colnames(prev)[2] = "prevmax"
colnames(prev)[3] = "previsao"

#prev
#row.names$prev = NULL
#prev

# Criando datas para as previsões
start = as.Date("2014-08-01")
end = as.Date("2024-07-01")
dataprev = seq(from=start, by="month", to=end)

#Criando um vetor com a data dos valores passados e das
previsões
data3 = append(dados2$data2, dataprev)

#Criando um novo banco com os valores passados e valores
preditos
temp2 = rep(NA, n_prev)
prevmin = rep(NA, n_dados)
prevmax = rep(NA, n_dados)
previsao = rep(NA, n_dados)

temp3 = append(tempmin, temp2)

previsoes.na = cbind(prevmin, prevmax, previsao)

previsoes = rbind(previsoes.na, prev)

previsoes2 = cbind(previsoes, temp3, data3)

#gráfico

ggplot(previsoes2, aes(data3)) +
  geom_line(aes(y = temp3, colour = "Temperatura")) +
  geom_line(aes(y = previsao, colour = "Previsão"))+
  geom_line(aes(y = prevmin, colour = "IC 95%"))+
  geom_line(aes(y = prevmax, colour = "IC 95%"))+
  geom_ribbon(aes(ymin = prevmin, ymax = prevmax), fill =
"grey", alpha = 0.4)+
  theme_grey(base_size = 18)+
  xlab("") + ylab("Temperatura (°C)")

```

C – Programação: câmbio

```
setwd("C:\\Users\\Leonardo\\Desktop\\Backup\\UnB\\Monografia\\Dados")

dados = read.csv(file="dados.csv", head=T,
stringsAsFactors=F)

library(ggplot2)
library(TSA)
library(tsDyn)

#Mudando a classe das variáveis
dados$tempo = as.Date(dados$tempo)
dados$valor = as.numeric(dados$valor)

class(dados$tempo)
class(dados$valor)

tsvalor = ts(dados$valor)

#Criando os log-retornos
logret_valor = diff(log(tsvalor))
logret_valor2 = diff(log(dados$valor))

#Banco de dados com logret_valor2 e uma sequencia de 5086
numeros
seq = seq(from=1, to=5086)

dados2 = data.frame(seq, logret_valor2)

#####
##### Análise Descritiva #####
#####

#Resumo
summary(dados$valor)
summary(dados2$logret_valor2)

#Densidade da série "pura"
ggplot(dados, aes(x=valor)) +
  geom_density(col="#00a7ff", fill="#00a7ff", alpha=0.5) +
  scale_x_continuous(breaks=seq(0.5,4,0.5)) +
  geom_vline(aes(xintercept=1.957, colour="red"),
            linetype="dashed", size=1) +
  theme_grey(base_size = 18)+
  xlab("Cotação do Real (R$)") + ylab("Densidade")
```

```

#Gráfico da série "pura"
ggplot(dados, aes(x=tempo, y=valor)) +
  geom_line(col="blue") +
  scale_y_continuous(breaks=seq(0.5,4,0.25)) +
  theme_grey(base_size = 18)+
  xlab("") + ylab("Cotação do dólar (R$)")

#Densidade dos log-retornos
ggplot(dados2, aes(x=logret_valor2)) +
  geom_density(col="#00a7ff", fill="#00a7ff", alpha=0.5) +
  geom_vline(aes(xintercept=0.0001, colour="red"),
             linetype="dashed", size=1) +
  theme_grey(base_size = 18)+
  scale_x_continuous(limits = c(-0.10, 0.10))+
  xlab("") + ylab("Densidade")

#Gráfico dos log-retornos
ggplot(data=dados2, aes(x=seq, y=logret_valor2)) +
  geom_line(col="blue") +
  theme_grey(base_size = 18)+
  xlab("") + ylab("")

# Em caso de erro
#dev.off()

plot(logret_valor, type='l', col='blue', ylab='Log-
retornos', xlab='Tempo')

#####
#Lidando com NAs
logret=na.exclude(logret_valor)

#####
##### LINEARIDADE #####
#####

# 1) Verificando a linearidade da série graficamente
lagplot(logret)

# Fazendo no ggplot2

#Criando um banco
logret_lag1 = logret[-1]
logret_lag2 = logret_lag1[-1]
logret_lag3 = logret_lag2[-1]
logret_lag4 = logret_lag3[-1]

```

```

logret_lag5 = logret_lag4[-1]
logret_lag6 = logret_lag5[-1]

lags = cbind(logret, logret_lag1, logret_lag2, logret_lag3,
             logret_lag4, logret_lag5, logret_lag6)

lags = as.data.frame(lags)

#Gráficos
lag1 = ggplot(lags, aes(x=logret_lag1, y=logret)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess, se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-1])) + ylab(expression(Y[t]))

lag2 = ggplot(lags, aes(x=logret_lag2, y=logret)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess, se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-2])) + ylab(expression(Y[t]))

lag3 = ggplot(lags, aes(x=logret_lag3, y=logret)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess, se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-3])) + ylab(expression(Y[t]))

lag4 = ggplot(lags, aes(x=logret_lag4, y=logret)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess, se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-4])) + ylab(expression(Y[t]))

lag5 = ggplot(lags, aes(x=logret_lag5, y=logret)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess, se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-5])) + ylab(expression(Y[t]))

lag6 = ggplot(lags, aes(x=logret_lag6, y=logret)) +
  geom_point() +      # Use hollow circles
  geom_smooth(method=loess, se=FALSE) +
  theme_grey(base_size = 18)+
  xlab(expression(Y[t-6])) + ylab(expression(Y[t]))

# Função para juntar os gráficos numa página só
multiplot <- function(..., plotlist=NULL, file, cols=1,
layout=NULL) {
  require(grid)

  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)

```

```

numPlots = length(plots)

# If layout is NULL, then use 'cols' to determine layout
if (is.null(layout)) {
  # Make the panel
  # ncol: Number of columns of plots
  # nrow: Number of rows needed, calculated from # of cols
  layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                    ncol = cols, nrow =
ceiling(numPlots/cols))
}

if (numPlots==1) {
  print(plots[[1]])
} else {
  # Set up the page
  grid.newpage()
  pushViewport(viewport(layout =
grid.layout(nrow(layout), ncol(layout))))

  # Make each plot, in the correct location
  for (i in 1:numPlots) {
    # Get the i,j matrix positions of the regions that
    contain this subplot
    matchidx <- as.data.frame(which(layout == i, arr.ind
= TRUE))

    print(plots[[i]], vp = viewport(layout.pos.row =
matchidx$row,
                                layout.pos.col =
matchidx$col))
  }
}

multiplot(lag1, lag2, lag3, lag4, lag5, lag6, cols=2)

# 2) Teste de Keenan
Keenan.test(logret)

# 3) Teste Threshold
tlrt(logret, 2, d=1, a=.05, b=.95)

```



```

#####
#####  MODELAGEM  #####
#####

# Estimando um modelo TAR
tar_logret = tar(logret, p1=4, p2=4, d=1,
                 a=.05, b=.95, method="MAIC", print=T)

setar_logret = setar(logret, m=3, steps=1)

##### Diagnóstico
tsdiag(tar_logret)

Box.test(tar_logret$residuals, lag=12, type="Ljung-Box")

#Gráfico dos resíduos
res = tar_logret$residuals
std_res = tar_logret$std.res
seqres = seq(from=1, to=4714)
dados_res = data.frame (std_res,seqres)

ggplot(dados_res, aes(x=seqres, y=std_res)) +
  geom_point(col='blue', shape=1) +
  geom_hline(aes(yintercept=0)) +
  geom_hline(aes(yintercept=3, colour="red")) +
  geom_hline(aes(yintercept=-3, colour="red")) +
  theme_grey(base_size = 18)+
  xlab("") + ylab("")

# Gráfico da FAC dos resíduos
fac = acf(res)

bacf <- acf(res, plot = FALSE)
bacfdf <- with(bacf, data.frame(lag, acf))

q <- ggplot(data = bacfdf, mapping = aes(x = lag, y = acf))
+
  geom_hline(aes(yintercept = 0)) +
  geom_segment(mapping = aes(xend = lag, yend = 0)) +
  geom_hline(aes(yintercept=(2/sqrt(4714)), colour="red"))
+
  geom_hline(aes(yintercept=- (2/sqrt(4714)),
colour="red"))+
  theme_grey(base_size = 18)+
  xlab("Lag") + ylab("FAC")
q

```