UNIVERSIDADE ABERTA

Variabilidade e quebras de estruturas em séries temporais. Comparação de métodos e aplicação a séries econômico-financeiras

Pedro Guilherme Frade Moro

Mestrado em Estatística, Matemática e Computação

Ramo Estatística Computacional

2025

UNIVERSIDADE ABERTA

Variabilidade e quebras de estruturas em séries temporais. Comparação de métodos e aplicação a séries econômico-financeiras

Pedro Guilherme Frade Moro

Mestrado em Estatística, Matemática e Computação

Ramo Estatística Computacional

Dissertação orientada pela Professora Doutora Maria do Rosário Olaia Duarte Ramos

2025

DECLARAÇÃO RELATIVA ÀS CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TECEIROS

Agradecimentos

Agradeço a todos os meus colegas de profissão e de estudos, pelo apoio, direto ou indireto na realização deste trabalho.

A meus professores, em especial à professora Maria do Rosário Ramos pela orientação neste trabalho de forma fluida e leve.

A Minha esposa Aline e a minhas filhas, Melissa e Giovanna, pelo entendimento e apoio no tempo que investi neste trabalho.

A meus pais, Lucia e Lincoln (in memoriam) que me guiaram e continuam guiando nesta jornada que é a vida.

DECLARAÇÃO DE INTEGRIDADE

STATEMENT OF INTEGRITY

Declaro ter atuado com integridade na elaboração da presente dissertação/tese. Confirmo que em todo o trabalho conducente à sua elaboração não recorri à prática de plágio ou a qualquer outra forma de falsificação de resultados.

Mais declaro que tomei conhecimento integral do Regulamento Disciplinar da Universidade Aberta, publicado no Diário da República, 2.ª série, n.º 215, de 6 de novembro de 2013.

I hereby declare having conducted my thesis with integrity. I confirm that I have not used plagiarism or any form of falsification of results in the process of the thesis elaboration.

I further declare that I have fully acknowledged Disciplinary Regulations of the Universidade Aberta (regulation published in the official journal Diário da República, 2.ª série, N.º 215, de 6 de novembro de 2013).

Universidade Aberta, 23 de Outubro de 2024

Nome completo/Full name: Pedro Guilherme Frade Moro

Assinatura/Signature:

Assinado por: Pedro Guilherme Frade Moro
Identificação: BI31427754
Data: 2024-10-23 às 15:29:43

manuscrita ou digital / handwritten or digital

Resumo

Nesta dissertação, exploramos a análise de séries temporais financeiras, focando em métodos clássicos e mais recentes. Investigámos a deteção de pontos de mudança e a previsão de volatilidade em séries como a taxa de juro SELIC, os preços do ouro, os ETFs (fundos negociados em bolsa) e as criptomoedas.

Realizámos uma análise de pontos de mudança, utilizando os métodos PELT (Pruned Exact Linear Time) e SONDE (Self Organizing Neural Network for Detecting Novelties). Foram detetadas diversas rupturas na estrutura da série, consistentes com eventos de mercado conhecidos como a crise financeira de 2008 e a pandemia de COVID-19. Utilizámos modelos como SARIMA, Filtro de Kalman e GARCH, diretamente sobre as séries e sobre as suas componentes obtidas pela decomposição por modo empírico (EMD).

Observámos que as séries de volatilidade financeira apresentam diversos desafios na aplicação de um único método a toda a série, entre outros motivos pela quantidade de pontos de mudança, no entanto uma abordagem em janela móvel pode gerar resultados satisfatórios. Chama-nos a atenção o mau desempenho do algoritmo GARCH em relação ao SARIMA e ao Filtro de Kalman na nossa abordagem.

Discutimos a aplicação destes resultados em séries temporais reais do mercado financeiro e as suas aplicações práticas neste mesmo contexto, tais como a gestão de carteiras e a gestão de relações com clientes. A investigação sugere que há muito a explorar nesta área dinâmica e desafiante.

Abstract

In this dissertation, we explore the analysis of financial time series, focusing on both classical and more recent methods. We investigate the detection of changepoints and volatility forecasting in series such as the SELIC interest rate, gold prices, ETFs (exchange-traded funds), and cryptocurrencies.

We performed a changepoint analysis using the PELT (Pruned Exact Linear Time) and SONDE (Self Organizing Neural Network for Detecting Novelties) methods. Several structural breaks in the series were detected, consistent with known market events such as the 2008 financial crisis and the COVID-19 pandemic.

We employed models such as SARIMA, Kalman filter, and GARCH, directly on the series and on their components obtained by empirical mode decomposition (EMD). We observed that financial volatility series present several challenges in applying a single method to the entire series, due in part to the number of changepoints. However, a rolling window approach can yield satisfactory results. We were particularly struck by the poor performance of the GARCH algorithm compared to SARIMA and the Kalman filter in our approach.

We discuss the application of these results to real-world financial time series and their practical applications in this context, such as portfolio management and customer relationship management. The research suggests that there is much to explore in this dynamic and challenging field.

Sumário

Índice de figuras

Índice de Siglas e Acrônimos

ARMA: *Autorregressive Moving Average Models* – modelos autorregressivos e de média móvel.

ARIMA: *Autorregressive Integrated Moving Average Models* – modelos integrados autorregressivos e de média móvel.

ARFIMA: *Autorregressive Fractionary Integrated Moving Average Models* – modelos de autorregressão e média móvel fracionário integrado.

SARIMA: Seasonal *Autorregressive Integrated Moving Average Models* – modelos sazonais de autorregressão e média móvel fracionário integrado.

SARIMAX: Seasonal *Autorregressive Integrated Moving Average Models* Exogenous– modelos sazonais de autorregressão e média móvel fracionário integrado com variáveis exógenas.

ARCH: Autorregressive conditional heteroskedasticity model – modelo autorregressivo de heterocedasticidade condicional

GARCH: Generalized Autorregressive conditional heteroskedasticity model – modelo autorregressivo de heterocedasticidade condicional generalizado

CAPM: *Capital Asset Pricing Model* – modelo de precificação de ativos de capital

VAR: Vector Autorregression – Autorregressão vetorial

AR: Autorregressão

MA: Média Móvel

LOESS: *Locally estimated scatterplot smoothing* - Suavização de dispersão estimada localmente

SIC: *Schwartz Information Criterion* – Critério de Informação de Schwartz - Também conhecido como Bayes Information Criterion - Critério de Informação de Bayes

AIC: *Akaike Information Criterion* – Critério de Informação de Akaike

EMD: *Empirical Mode Decomposition* – Decomposição por modo empírico

IMF: *Intrinsic Mode Functions* – Funções de modo intrínseco

PELT: *Pruned Exact Linear Time* – "Tempo Linear Exato Podado"

SONDE: *Self-Organizing Neural Network for Detecting Novelties* – Rede neural auto organizável para deteção de novidades

SOM: *Self-Organizing-Maps*

BMU: *Best Matching Unit*

Repo: Repurchase agreement – operação de venda de uma obrigação na qual a parte vendedora assume um compromisso de recompra em uma data futura.

SELIC: Serviço Especial de Liquidação e Custódia: câmara aonde são registradas e negociadas as obrigações de dívida soberana do Brasil, também chamado de "o SELIC"

Taxa SELIC: Taxa Média apurada nas operações compromissadas (Repo) de um dia registradas no Serviço Especial de Liquidação e Custódia, também abreviada para simplesmente "a SELIC"

COPOM: Comitê de política monetária do Banco do Brasil

ETF: *Exchange Traded Fund* – Fundo negociado em Bolsa

EWZ: Ishares MSCI Brazil ETF – ETF de ações Brasileiras contidas no índice MSCI Brazil

MSCI: Empresa provedora de soluções de dados para o mercado financeiro

B3: Brasil, Bolsa e Balcão – Bolsa de valores Brasileira

BTC: Abreviação de Criptomoeda bitcoin no padrão ISO 4217

USD: Abreviação de Dólar dos Estados Unidos no padrão ISO 4217

EUR: Abreviação de Euro da União Europeia no padrão ISO 4217

BRL: Abreviação de Real Brasileiro no padrão ISO 4217

MAE: Erro Absoluto Médio

COVID-19: Doença por Coronavírus 2019

CRM: *Customer Relationship Management* – Gerenciamento de relações com clientes

1 Introdução

Uma série temporal é uma sequência de dados observados ao longo do tempo, geralmente em intervalos regulares. Esses dados podem representar uma ampla variedade de fenômenos, desde medições climáticas diárias até preços de ações em intervalos de minutos. A característica fundamental de uma série temporal é a dependência temporal, onde os valores observados em um ponto no tempo são influenciados pelos valores anteriores. Essa dependência torna a análise de séries temporais uma ferramenta poderosa para entender e prever comportamentos futuros. (Teixeira, 2012) (Morettin, 2017)

A análise de séries temporais encontra aplicação em diversos campos do conhecimento. Na meteorologia, por exemplo, é utilizada para prever temperaturas diárias a um passo, eventos extremos, ou modelar padrões climáticos. Na medicina, pode ser aplicada para monitorar sinais vitais e prever surtos de doenças e construir modelos epidemiológicos, como dos dados da pandemia COVID-19. Na engenharia, é usada para a manutenção preditiva de máquinas e equipamentos. Além disso, em ciências sociais, a análise de séries temporais Pode ser ùtil para entender tendências populacionais e comportamentos eleitorais. Esses exemplos ilustram a versatilidade e a importância dessa técnica analítica. (Teixeira, 2012) (Morettin, 2017)

Uma diferença crucial entre a análise de séries temporais e a análise de regressão tradicional é a consideração da dimensão temporal. Na análise de regressão, os dados são geralmente tratados como independentes e identicamente distribuídos, sem levar em conta a ordem em que foram observados o que exige, por exemplo, um cuidado especial no momento de realizar inferência. Em contraste, a análise de séries temporais reconhece que os dados são sequenciais e que a ordem das observações importa. Isso implica que técnicas específicas, como modelos autorregressivos e de médias móveis, são necessárias para capturar a estrutura temporal dos dados. (Morettin, 2017) (Morettin & Toloi, 2018)

No campo da economia e finanças, a análise de séries temporais é amplamente utilizada para modelar e prever variáveis econômicas e financeiras. Por exemplo, economistas utilizam séries temporais para analisar o PIB, taxas de desemprego e inflação. No setor financeiro, a análise de séries temporais é essencial para prever preços de ações, taxas de câmbio e volatilidade do mercado. Essas previsões são fundamentais para a tomada de decisões estratégicas, gestão de riscos e desenvolvimento de políticas econômicas. A capacidade de modelar e prever com precisão os comportamentos futuros torna a análise de séries temporais uma ferramenta indispensável nesses campos. (Morettin, 2017) (Fava & Alves, 1998)

Um conceito importante na análise de séries temporais é o de pontos de mudança (changepoints), que representam quebras na estrutura da série. Esses pontos indicam momentos em que há uma mudança significativa no comportamento dos dados, como uma alteração na média ou na variância. Nas séries temporais financeiras, os pontos de mudança são particularmente comuns devido a eventos econômicos, políticos ou sociais que podem impactar drasticamente os mercados. A identificação e o tratamento adequado desses pontos são cruciais, pois ignorá-los pode levar a modelos imprecisos e previsões errôneas. A deteção de pontos de mudança permite ajustar os modelos para refletir melhor a realidade, melhorando a robustez das análises e a confiabilidade das previsões. (Teixeira, 2012)

Neste trabalho, serão apresentados os conceitos fundamentais da análise de séries temporais, com enfoque nas séries temporais financeiras, bem como as principais técnicas e modelos utilizados na área. Apresentaremos estudos de caso práticos envolvendo séries financeiras, demonstrando como a análise de séries temporais pode ser aplicada para entender e prever o comportamento de variáveis financeiras. (Morettin, 2017)

1.1 Motivação

Diariamente, milhares de pessoas monitoram os preços de diversos ativos, seja para adquirir insumos industriais, poupar para a aposentadoria ou simplesmente especular. No centro desses mercados estão as instituições financeiras, responsáveis pela distribuição dos instrumentos financeiros e pela gestão dos riscos que os investidores preferem evitar. Um dos conceitos mais importantes na gestão de risco é a volatilidade: seu valor atual e futuro determinam os níveis de risco que instituições e investidores podem suportar. A mensuração incorreta da volatilidade pode tornar os mercados vulneráveis a crises ou impedir a execução de projetos benéficos para a sociedade.

As séries temporais financeiras são notoriamente complexas, frequentemente violando muitas das premissas dos métodos tradicionais de análise de séries temporais. Além disso, o avanço da internet acelerou a velocidade com que as informações circulam, tornando os mercados financeiros globais cada vez mais interconectados. Isso significa que eventos em um mercado podem rapidamente afetar mercados em todo o mundo. Portanto, é necessário o uso de métodos algorítmicos para a correta mensuração do risco.

Uma característica marcante das séries temporais financeiras é a presença de regimes distintos em diferentes períodos: há momentos de clara tendência de alta, outros de baixa e períodos de relativa estabilidade. Isso gera interesse em técnicas que possam antecipar e mensurar os pontos de mudança (changepoints), ou seja, as quebras na estrutura da série. Identificar esses pontos é crucial para ajustar os modelos de forma a refletir melhor a realidade e melhorar a precisão das previsões.

Recentemente, surgiu uma nova categoria de ativo financeiro: as criptomoedas. Inicialmente vistas como uma curiosidade efêmera, hoje elas se consolidaram como uma classe de investimentos própria (ao lado de ações, obrigações e commodities) sendo procuradas por investidores que buscam proteção contra o risco associado às moedas soberanas.

As séries temporais financeiras são diversas e estudar todas em um único trabalho é inviável. No entanto, estudar apenas uma série pode deixar de fornecer informações importantes sobre a interação entre diferentes séries. Neste trabalho, utilizaremos um conjunto de quatro séries temporais económico-financeiras: as variâncias realizadas do preço do ouro em dólares americanos, do fundo de índice negociado em bolsa (ETF) correspondente ao índice Brasil (EWZ), da criptomoeda Bitcoin e a taxa de juros base da economia brasileira – SELIC. Todos esses dados são diários e disponíveis publicamente. A estimativa da variância futura (volatilidade) e, principalmente, de seus pontos de mudança, fornece uma visão do risco no mercado financeiro e subsidia decisões de compra ou venda de instrumentos de proteção contra esses riscos, como as opções.

1.2 Objetivos e Estrutura do Trabalho

O objetivo deste trabalho é aprofundar o estudo de alguns métodos clássicos e métodos mais recentes para identificação de padrões, pontos de *mudança (utilizaremos o termo changepoints a partir daqui)* em séries temporais e previsão, e realizar algumas aplicações sobre séries temporais financeiras., que são também do interesse pessoal, uma vez que se relacionam com a atividade profissional do autor. A componente prática será realizada com recursos computacionais. Utilizaremos também estudos de correlação e causalidade entre séries diferentes, a fim de avaliarmos a possibilidade de uma série poder ser utilizada como variável antecipadora das demais e, consequentemente, de seus changepoints.

Com esta meta, iniciaremos o trabalho com um enquadramento teórico dos conceitos e modelos que serão mais relevantes. Faremos, também, uma caracterização geral das séries temporais financeiras e alguns dos principais desafios relativos às mesmas.

Seguiremos com um capítulo dedicada à apresentação das principais técnicas de análise de séries temporais, esta seção se inicia com uma caracterização dos principais processos geradores de séries temporais. Na sequência abordaremos os principais modelos lineares univariados, tais como a família ARIMA. Como um dos objetivos principais deste trabalho é o estudo de volatilidade, é relevante termos uma subseção sobre modelos para variância, tais como o ARCH e GARCH.

Estamos particularmente interessados em quatro séries financeiras distintas, uma delas de taxa básica de juros, as demais de preços de ativos financeiros. O *Capital Asset Pricing Model* uma das bases da teoria financeira, afirma que o retorno esperado de cada ativo é uma função linear da taxa livre de risco e do prêmio de risco do mercado, ajustado pelo beta(coeficiente de correlação entre o preço do ativo e o retorno do mercado como um todo) de cada ativo, sugerindo uma relação entre as quatro variáveis. Por isso, a seção contendo as técnicas de séries temporais contém uma explicação dos principais métodos multivariados, como o VAR, comentaremos também sobre o conceito de causalidade segundo Granger e sobre os conceitos de correlação, cointegração e a diferença entre ambos.(Póvoa, 2012)

Encerramos a seção sobre análise de séries temporais com uma exposição de modelos em espaço de estado e o teorema de Takens e o filtro de Kalman, aplicáveis às séries temporais determinísticas. Como as séries temporais não possuem um único processo gerador definido, abordaremos também o conceito de decomposição por modo empírico que pode ser útil na decomposição das séries em múltiplas subséries independentes, potencialmente com processos geradores mais definidos. Nossa seção de revisão bibliográfica se encerra com uma seção dedicada a análise de changepoints, que se enquadra nos objetivos práticos deste trabalho.

A seguir, iniciaremos a parte prática do trabalho, para cada uma das quatro séries faremos uma análise exploratória da própria série e de sua variância realizada. Seguiremos com por uma análise de changepoints e uma modelagem univariada da variância. Feitas as análises univariadas, é explorada uma perspetiva multivariada com as quatro séries em estudo. Por fim, apresentaremos nossas conclusões e sugestões de próximos trabalhos.

2 Séries Temporais: Enquadramento Teórico

Uma série temporal, em sua definição mais básica, é um conjunto de observações ordenadas ao longo do tempo. As séries temporais podem ser classificadas como contínuas ou discretas. Uma série contínua é incontavelmente infinita, como, por exemplo, o valor da maré em uma determinada localização, que pode ser medido a qualquer momento. Já uma série discreta é contável, podendo ser finita ou infinita. Neste estudo, trabalharemos com séries temporais financeiras que são divulgadas diariamente e que serão consideradas, portanto, do tipo discreto. (Toloi & Morettin, 2018) (Rynne & Youngson, 2000)

2.1 Características das séries temporais

A seguir apresentaremos algumas características das séries temporais de uma forma geral e sua relevância para a definição de estratégias de modelagem. Neste trabalho estamos interessados em séries temporais financeiras, as quais, com frequência apresentam características distintas das séries temporais observadas nas ciências naturais. Abordaremos também estas especificidades.

2.1.1 Homocedasticidade

Homocedasticidade significa uma série temporal possuir variância finita e constante em toda sua extensão. Esta característica é muitas vezes avaliada na prática pelos resíduos de um modelo(diferença absoluta entre valor previsto e valor observado): espera-se que os resíduos possuem distribuição aleatória (R Core Team, 2021) (Morettin & Toloi, 2018)

Boa parte dos modelos clássicos de séries temporais, se baseia na premissa de homocedasticidade. Em nossa metodologia, aplicaremos os testes de Box-Pierce e Ljung-Box sobre os resíduos de um modelo para averiguar a aleatoriedade e o teste de Shapiro-WIlk para identificar se são normais, sendo ambas as condições verdadeiras, poderemos assumir homocedasticidade. Tais testes serão definidos formalmente na seção 4.4.5 (Morettin & Toloi, 2018)

Uma forma bastante comum de estabilizar a variância de um conjunto é efetuar uma transformação não linear na mesma. Um exemplo desta transformação é a de Yeo-Johnson, dada por:

$$y(\lambda) = \begin{cases} \dfrac{(y+1)^{\lambda}-1}{\lambda} & se\ \lambda \neq 0, y \geq 0 \\ \ln(y+1) & se\ \lambda = 0, y \geq 0 \\ \dfrac{-\left((-y+1)^{2-\lambda}-1\right)}{(2-\lambda)} & se\ y < 0, \lambda \neq 2 \\ -\ln(-y+1) & se\ y < 0, \lambda = 2 \end{cases}$$

Equação 1: Transformação de Yeo-Johnson

Onde y , são nossos dados e λ é um parâmetro a ser estimado. No entanto, a literatura de séries temporais afirma que as transformações não melhoram a qualidade das predições, ainda que eliminem a heterocedasticidade. Razão pela qual não serão utilizadas neste trabalho. (Morettin & Toloi, 2018)

A presença de heterocedasticidade, além de exigir abordagens robustas a este fenômeno, pode indicar quebras de estruturas na série que devam ser avaliadas, por exemplo por meio

de metodologias de análise de changepoints. Séries temporais financeiras são conhecidas por apresentarem regimes distintos, por isso, em nossa análise exploratória investigaremos a presença ou não da homocedasticidade. Além do uso dos testes de Box-Pierce e Ljung-Box sobre os modelos, inspeções visuais das séries, tais como gráficos de dispersão e de recorrência serão utilizados com este fim.

2.1.2 Autocorrelação

Uma das principais características de séries temporais é a presença de autocorrelação. Esta propriedade consiste na dependência intra série, isto é, nos valores passados de uma série influenciarem nos valores futuros e será detalhada aquando abordagem desta situação através da família de modelos ARIMA.

A presença de autocorrelação é um dos principais fatores que diferenciam séries temporais de outros problemas de regressão. Sua presença é investigada por meio dos gráficos de função de autocorrelação e autocorrelação parcial. Além disso, a maioria dos algoritmos de séries temporais contém componentes autorregressivas. Séries temporais financeiras, por sua vez, não raro apresentam a característica de autocorrelação e esta será utilizada em nossos estudos.

Por outro lado, autocorrelação pode se apresentar como desafio nos estudos de changepoints, uma vez que os métodos disponíveis para tal muitas vezes tem como premissa a independência das observações. Particularidades dos métodos de detecção de changepoints serão melhor abordadas na seção 2.5. (Teixeira, 2012)

2.1.3 Estacionaridade

Na análise de séries temporais, processos estocásticos podem ser classificados como estacionários ou não-estacionários. Intuitivamente, um processo estocástico $Z(t)$ é considerado estacionário quando suas características estatísticas não dependem da escolha da origem temporal, ou seja, as propriedades de $Z(t)$ são as mesmas de $Z(t+\tau)$ para qualquer deslocamento τ. (Toloi & Morettin, 2018)

Formalmente, a estacionariedade pode ser classificada em duas formas: estacionariedade forte e estacionariedade fraca. Um processo é dito fracamente estacionário (ou estacionário no sentido amplo) se satisfaz as seguintes condições: (Toloi & Morettin, 2018)

i. A média da série é constante para todo instante de tempo.
ii. A variância da série é constante para todo instante de tempo
iii. A covariância entre os valores da série em dois pontos no tempo depende apenas do espaçamento (*lag*) entre esses pontos.

A estacionariedade forte, por outro lado, exige uma condição mais rígida: além das três condições da estacionariedade fraca, requer que as distribuições conjuntas de quaisquer subconjuntos finitos da série temporal permaneçam invariantes ao longo do tempo. Esta definição de estacionariedade é mais aplicável a contextos teóricos e a processos estocásticos rigorosamente definidos (Toloi & Morettin, 2018)

Para muitos modelos de séries temporais, a suposição de estacionariedade, especialmente na forma fraca, é essencial, pois simplifica o tratamento analítico e

computacional. No entanto, em séries temporais reais, como as séries financeiras, mesmo a estacionariedade fraca é frequentemente violada. Existem métodos para avaliar a estacionariedade, como o teste aumentado de Dickey-Fuller, e técnicas para transformar séries em estacionárias, como a diferenciação. (Morettin, 2017)

2.1.4 Sazonalidade

Sazonalidade é um termo de difícil definição conceitual formal. De uma forma geral, consideramos que fenômenos que ocorrerem de forma cíclica em períodos longos são considerados sazonais. Quando este fenômeno está presente, há a necessidade de um ajustamento prévio, para modelagem (Toloi & Morettin, 2018)

De uma forma geral, um processo gerador de uma série temporal Z_t pode ser escrito como a soma de um processo sazonal S_t e outro T_t não sazonal:

$$Z_t = T_t + S_t$$

Equação 2: Processo gerador com sazonalidade

Desejamos obter uma estimativa do processo S_t , para assim retirá-lo da série total, desta forma podemos modelar a parte não sazonal do processo pelas ferramentas adequadas. (Toloi & Morettin, 2018)

A Sazonalidade pode ser do tipo determinística ou estocástica. Entendemos por sazonalidade determinística, aquela que pode ser prevista perfeitamente por um modelo de regressão, neste caso, o processo S_t, será dado por:

$$S_t = \sum_{j=1}^{p} \alpha_j d_{jt}$$

Equação 3: Sazonalidade determinística

Onde as variáveis d_{jt}são do tipo periódico, como senos, cossenos, dentre outros, p é o período que observamos (por exemplo, uma série mensal de periodicidade anual terá $p = 12$) (Toloi & Morettin, 2018)

Já a sazonalidade do tipo estocástica é aquela cuja componente sazonal varia com o tempo. Neste caso, podemos modelá-la por um processo de médias móveis. Tomando uma série, nosso processo S_t se torna uma estimativa $\widehat{S_t}$ dada por:

$$\widehat{S_t} = \overline{Y_t} - \overline{Y}$$

Equação 4: Sazonalidade estocástica

Onde $\overline{Y_j}$ e $\overline{Y}$ são os valores médios da parte estocástica no período j e no processo como um todo, respetivamente. Neste caso, $\widehat{S_t}$ é modelável por um processo de médias móveis MA(n).(Toloi & Morettin, 2018)

Para uma análise adequada sobre a necessidade de tratamento adicional em um modelo devido à sazonalidade, é útil aplicar o teste de Friedman. Este teste é particularmente eficaz quando os dados correlacionados podem ser estruturados em "blocos", que são conjuntos de observações que se espera serem homogêneos ou similares entre si, como diferentes anos em uma série temporal. (Toloi & Morettin, 2018)

O teste originalmente envolve a organização dos dados em "blocos" e a aplicação de diferentes "tratamentos" dentro de cada bloco. Este teste possui as seguintes hipóteses:

H_0: Todos os blocos possuem a mesma distribuição

H_1: Pelo menos um bloco possui distribuição diferente

A estatística do teste de Friedman é calculada pela fórmula:

$$T = \frac{12}{pk(k+1)} \sum_{j=1}^{k} R_j^2 - 3p(k+1)$$

Equação 5: Estatística de Friedman

Onde T é a estatística de Friedman, p é o número de blocos, k é o número de tratamentos, R_j é a soma das classificações para o j-ésimo tratamento.

A significância do teste é determinada pela comparação da estatística calculada com a distribuição exata ou, quando p e k são grandes, com uma distribuição aproximada ~~do tipo~~ χ2 com $k - 1$ graus de liberdade. Isso permite decidir pela rejeitação ou não da hipótese nula. Quando aplicado a séries temporais, é muito comum considerar o ano (período) como os *p* "blocos", e a subdivisão temporal por meses como os *k* "tratamentos". Isso facilita a identificação de padrões sazonais que podem estar afetando a série temporal de maneira significativa.

Conforme referido, as séries temporais financeiras não raramente apresentam efeitos sazonais ou cíclicos, como por exemplo as taxas de juros, que são notoriamente cíclicas. Assim, serão utilizadas ainda outras técnicas auxiliares como o periodograma para identificar estruturas cíclicas a fim de identificar a existência ou ausência de componentes cíclicas. (C. Hull, 2009) (P. Chan, 2021) (Morettin, 2017)

2.1.5 Ruído

Ruído pode ser definido como uma série temporal aleatória e que não é capturada pelas técnicas de modelagem de séries temporais, sendo necessário a aplicação de modelos probabilísticos. Em séries temporais é muito comum usarmos a definição de processo de ruído branco, o qual consiste em uma variável aleatória de média zero, independente e identicamente distribuída. A título de curiosidade: o nome “ruído branco” se deve a analogia com a luz branca, na qual todas as oscilações periódicas possíveis estão presentes com a mesma intensidade. O conceito de ruído e ruído branco também existe no contexto multivariado, com as assunções adicionais de independência e ausência de correlações entre séries. (Toloi & Morettin, 2018) (Shumway & Stoffer, 2016) (Toloi & Morettin, 2020)

Séries financeiras, de uma forma geral, possuem uma grande componente de ruído, isto é, ao decompormos a série, parte significativa dela é dada por uma série puramente aleatória. Este fato gera um grande desafio em qualquer tentativa de modelar séries temporais financeiras: o grande risco de *overfit* (sobreaprendizado), este maior risco se deve ao fato de modelos encontrarem no ruído padrões inexistentes e assumirem que os mesmos se repetirão no futuro. (P. Chan, 2021)

A presença de um domínio de componentes de ruído em nossas séries será analisada por meio do *Wald-Wolfowitz Runs test* (Teste de corridas, doravante usaremos a expressão *runs test*) o qual será melhor definido na seção de séries puramente aleatórias.

2.1.6 Dependência

Na seção 2.1.2 mencionamos a dependência intra série, a autocorrelação, Além desta, pode existir também o efeito de dependência extra série, na qual uma série pode interferir na outra: uma relação conhecida é a que ocorre com as taxas básicas de juros e os preços de ações e outros ativos de maior volatilidade, uma vez que o prêmio de risco, ao subir, diminui a demanda pelos mesmos, além de aumentar as taxas de desconto/custo de carregamento de papéis. (P. Chan, 2021) (Securato, 2008)

Além disso, queda nos preços podem levar ao fenômeno do *squeeze* aonde investidores são forçados a vender ativos a mercado, para levantar os recursos necessários à cobertura de prejuízos em posições perdedoras, desta forma, acaba por existir uma relação de dependência entre as duas séries, ainda que temporária. Um exemplo deste fato foi a crise de 2008, aonde prejuízos com ativos atrelados a financiamentos imobiliários forçaram investidores a vender ações para saldar dívidas. Embora as ações vendidas muitas vezes não tivessem qualquer relação com o mercado imobiliário ou de banca seus preços foram afetados por este movimento de venda. Dado estes fatos, incluiremos análises multivariadas, em especial a causalidade de Granger e estudos de cointegração, de forma a identificarmos os se nossas séries apresentam algum tipo de relação entre si (P. Chan, 2021)

2.2 Volatilidade

O conceito de volatilidade é bastante variado dentro da literatura financeira e consiste na mensuração da incerteza no preço de um determinado ativo. Esta incerteza possui várias causas, como por exemplo a disponibilização de novas informações que impactam os retornos dos ativos e os efeitos da própria negociação, como observado nas análises de Fama e French. Duas definições bastante comuns para volatilidade são a volatilidade implícita e volatilidade histórica. (C. Hull, 2009)

Para explicarmos a volatilidade implícita é necessária uma breve introdução do método de precificação de opções *Black and Scholes*, este método consiste em uma função que recebe como argumentos o preço da opção, o preço do ativo subjacente o tempo até o vencimento da opção a taxa livre de risco e a volatilidade e calcula o preço da opção. (C. Hull, 2009)

Todos estes argumentos, exceto a volatilidade são dados disponíveis publicamente, a volatilidade implícita consiste em inverter o método de *Black and Scholes* isto é: obtemos a volatilidade ao fornecer ao modelo o preço de mercado da opção, junto com os demais argumentos. (C. Hull, 2009)

Volatilidade histórica, por sua vez, consiste no desvio padrão rolante, de 21 dias, e anualizado dos retornos de um ativo, no decorrer de nosso trabalho, utilizaremos apenas esta definição. (C. Hull, 2009)

para obter a volatilidade, calcularemos inicialmente o preço relativo, isto é, seja S_t o preço no instante atual, S_{t-1} o preço no instante interior, obteremos:

$$\mu_t = ln\left(\frac{S_t}{S_{t-1}}\right)$$

Equação 6: Preço relativo (C. Hull, 2009)

e a volatilidade s será dada por:

$$s = \sqrt{\frac{1}{n-1}\Sigma_{t=1}^{n}(\mu_t - \bar{\mu})^2}$$

Equação 7: Definição de Volatilidade Histórica (C. Hull, 2009)

Volatilidade, tanto em sua forma implícita, como histórica são amplamente utilizadas na precificação de ativos financeiros, mensuração de riscos de mercado, otimização de portfólios, dentre outros, desta forma, sua correta estimação é de suma importância para a solidez dos mercados globais.

2.3 Análise de Séries Temporais

A Análise de séries temporais é focada nos objetivos a seguir:

a) Fazer Previsões de valores futuros de uma série temporal
b) Descrever o comportamento da série
c) Procurar periodicidades nos dados
d) Investigar o mecanismo gerador dos dados

No caso de séries temporais financeiras, o mecanismo gerador dos dados tende a ser dinâmico (isto é, tende a mudar com o tempo), de forma que não é possível identificá-lo como ocorre nas ciências naturais, razão pela qual o interesse maior está nos três primeiros objetivos (a, b e c). Para atender a estes objetivos precisamos entender o caráter da série temporal que estamos estudando. (Toloi & Morettin, 2018)

A Análise de séries temporais pode ser feita em dois domínios: A análise no domínio do tempo e a análise no domínio da frequência. A análise no domínio do tempo parte do pressuposto de que a correlação entre dois pontos adjacentes é mais bem explicada por meio de uma dependência dos valores atuais em relação aos anteriores. Desta forma podemos modelar a série como uma função dos valores atuais e passados. A Análise no domínio da frequência tem por objetivo estudar periodicidades ou sazonalidades presentes nas séries temporais, um exemplo desta análise é a análise espectral. (Toloi & Morettin, 2018) (Shumway & Stoffer, 2016)

Utilizam-se preferencialmente modelos paramétricos na análise no domínio do tempo e modelos não-paramétricos na análise no domínio da frequência. Entende-se por modelo paramétrico aquele em que o processo gerador é dado por um número finito de parâmetros os quais assumimos como premissas para a modelagem da série. Modelos não-paramétricos são aqueles nos quais não fazemos quaisquer assunções neste sentido. De uma forma geral, modelos paramétricos tendem a ter maior poder estatístico que os não paramétricos quando suas premissas não são violadas. (Toloi & Morettin, 2018) (Reis, Melo, Andrade, & Calapez, 1996)

Além dos tipos já citados, podemos também agrupar as séries temporais existentes em três grandes classes: as séries determinísticas, as estocásticas e as aleatórias. Discorreremos sobre o caráter de cada uma delas e como devemos analisar cada uma delas.

2.3.1 Decomposição da Série Temporal.

A decomposição de uma série temporal consiste em separar a série em componentes, as quais, quando combinadas, resultam na série temporal original. A ideia por trás da decomposição consiste em compreender padrões distintos da série. É comum aplicar a cada uma das componentes a abordagem matemática /estatística mais adequada para sua análise.

Nas abordagens clássicas, inicialmente precisamos entender se uma série Y_t é determinística ou estocástica e, após isto, se possui componentes de tendência(T_t), sazonalidade (S_t) e ruído(ϵ_t), que devem ser extraidas, resultando apenas a componente X_t.

Estas componentes podem possuir uma relação aditiva, na forma $Y_t = T_t + S_t + X_t + \epsilon_t$, aonde suas amplitudes são estáveis ao longo do tempo. Ou uma relação multiplicativa, isto é, na forma: $Y_t = T_t \times S_t \times X_t \times \epsilon_t$ aonde as amplitudes das componentes podem variar. Neste caso, uma abordagem comum é aplicar a transformação logarítimica para linearizar a relação entre as componentes.

O tratamento do ruído em séries temporais é muitas vezes realizado de maneira a minimizar sua influência, considerando-o como um componente não modelável de forma direta pelos algoritmos tradicionais de séries temporais. Em contraste, a sazonalidade é frequentemente modelada de forma distinta e cuidadosa, como discutido anteriormente. (Toloi & Morettin, 2018)

Quanto à tendência, ela é comumente tratada através da aplicação de diferenciação. Este processo envolve diferenciar a série temporal até que ela se torne estacionária, ou seja, suas propriedades estatísticas como média e variância sejam constantes ao longo do tempo. Uma vez que a série estacionária é modelada, ela é então integrada para que se possam gerar previsões ou obter valores que se apliquem à série temporal original. (Toloi & Morettin, 2018)

Uma vez removidas as partes de sazonalidade e tendência, necessitamos escolher uma abordagem para modelar a série, para isso é importante conhecer o tipo de seu processo gerador. Diferentes técnicas são aplicáveis a cada processo gerador, logo, conhecer o processo gerador nos permite identificar quais as abordagens mais adequadas para cada uma das séries. Ou, como no caso de séries temporais aleatórias, identificar que não será possível efetuar uma modelagem de séries temporais. (Ishii, Rios, & Mello, 2011)

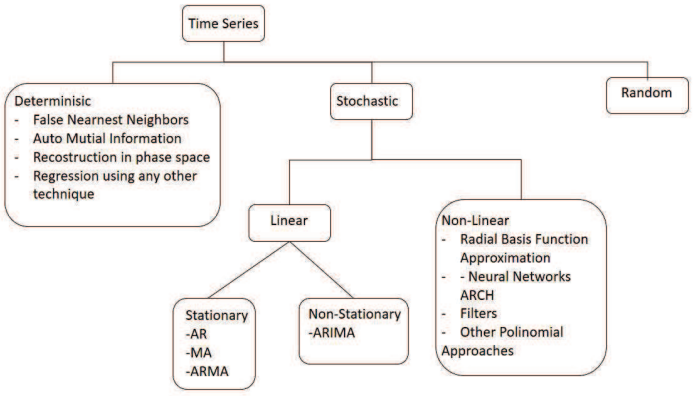

Figura 1:Dendrograma contendo classificação dos processos geradores de séries temporais e sugestão de ferramentas de modelagem (Ishii, Rios, & Mello, 2011)

)

Torna-se, portanto, necessário efetuarmos análises no sentido de identificarmos se a série é determinística ou estocástica e, neste último caso, se trata de uma série linear, não linear ou aleatória.

Decomposição de séries em componentes de tendência ou sazonalidade pode ser feita pelo algoritmo LOESS. Em situações nas quais uma série apresente componentes de múltiplas naturezas, pode-se usar a decomposição por modo empírico (EMD), ambos serão melhor detalhados posteriormente

4.4.2.1 Métodos de Decomposição de séries temporais

Séries temporais financeiras podem ser constituídas de componentes de múltiplas naturezas, para as quais pode ser adequado aplicar uma técnica diferente de modelagem. Neste caso, se for possível, seria interessante separar as séries em suas componentes, modelá-las individualmente com as melhores técnicas possíveis, e efetuar uma predição final utilizando a soma das componentes. (Flandrin, Rilling, & Gonçalvés, 2004) (Rilling & Flandrin, 2008)

Existem diversos métodos para decomposição, neste trabalho abordaremos dois métodos não paramétricos, que são a LOESS e a EMD, explicaremos cada uma delas na sequência.

4.4.2.1.1 Decomposição por suavização de dispersão estimada localmente (LOESS)

Suavização de dispersão estimada localmente (LOESS na sigla em inglês: *Locally Estimatated Scatterplot Smoothing*) é uma técnica que objetiva extrair de uma série temporal suas componentes de tendência e sazonalidade. Ela propõe o ajuste de polinômios de forma local na série e usa os mesmos como filtros para extração das componentes (R. B. Cleveland, 1990)

LOESS é uma técnica não paramétrica e bastante robusta para seus objetivos, no entanto em nosso estudo, a série pode apresentar componentes estocásticas ou

determinísticas, sendo que estas últimas não se encaixam na proposta da LOESS, razão pela qual abordaremos a *Empirical Mode Decomposition* (EMD) (R. B. Cleveland, 1990) (Flandrin, Rilling, & Gonçalvés, 2004)

4.4.2.1.2 Decomposição por modo empírico (EMD)

Empirical Mode Decomposition ou EMD é uma técnica de decomposição adaptativa que permite analisar séries temporais de forma não linear e não estacionária. O método divide a série em componentes intrínsecas chamadas de funções de modo Intrínseco (IMF).

Essas funções são obtidas por meio do chamado *sifting process* (processo de peneiramento), que consiste em um algoritmo iterativo envolvendo as seguintes etapas: (Flandrin, Rilling, & Gonçalvés, 2004) (Rilling & Flandrin, 2008) (Santiago Velasco-Forero, 2022)

1. Localizamos todos os máximos e mínimos locais de uma série $f(x)$
2. Interpolamos conjuntamente todos os máximos locais, obtendo uma função $\hat{f}(x)$ (envelope superior) e todos os mínimos locais, formando uma função $\bar{f}(x)$ (envelope inferior)
3. A IMF daconsiste na a média de ambas as funções: $IMF(x) = \frac{1}{2}(\hat{f}(x) + \bar{f}(x))$
4. Iterar este processo no resíduo, $r(x) = f(x) - IMF(x)$

Desfa forma, a série $f(x)$ pode ser escrita como $f(x) = \sum_{i=1}^{k} IMF_i(x) + r(x)$, o processo iterativo acima continua até que um critério de parada seja atingido, por exemplo quando $r(x)$ se tornar menor do que um limite pré definido. (Flandrin, Rilling, & Gonçalvés, 2004) (Rilling & Flandrin, 2008) (N.E. Huang, 2006)

É possível efetuar uma análise de frequência-intensidade-tempo das componentes EMD por meio do gráfico do espectro de Hilbert, o qual se baseia na aplicação da transformada de mesmo nome e que será mencionada na seção 4.4.5.2.3, Análise Espectral. Em nossos estudos, utilizarem as dependências Rlibeemd e gsignal para a aplicação da EMD e da construção do espectro de Hilbert. (Flandrin, Rilling, & Gonçalvés, 2004) (Rilling & Flandrin, 2008) (Helske J, 2021) (Van Boxtel, 2021)

2.3.2 Séries Determinísticas e Séries Estocásticas

Séries determinísticas são aquelas em que o comportamento futuro pode ser totalmente determinado pelos valores passados da mesma série ou (em uma abordagem multivariada), de outra série. Este tipo de série possui auto correlação/correlação forte em pelo menos um dos lags. De uma forma geral, pode ser tratada no domínio univariado como um modelo $AR(p)$ ou no domínio multivariado por análises de causalidade de Granger. Já uma série estocástica é definida como uma família de funções na qual a cada instante, o valor da série é dado por uma variável aleatória. (Toloi & Morettin, 2018) (Shumway & Stoffer, 2016) (Morettin, 2017)

Uma forma de avaliarmos se uma série é determinística ou estocástica é por meio de seu Gráfico de recorrência (*recurrence plot)*, aliado ao gráfico de auto correlação. O *recurrence plot* é um gráfico gerado a partir da matriz de recorrência. Esta, por sua vez é uma matriz $n \ x \ n$ (Onde n é determinado pelo tamanho da série temporal, pela dimensão de *embedding (*embutimento) e pelo número de lags considerados), na qual se os pontos i e j da série

estiverem a uma distância menor que ϵ eles são considerados “Recorrentes” e o elemento ij recebe o valor 1, caso contrário recebe o valor zero. Ou de outra forma: $R_{ij} = H(\epsilon_i - ||\overrightarrow{x_i} - \overrightarrow{x_j}||), \overrightarrow{x_i} \in \mathbb{R}^m, i, j \in [1,2,\dots,n]$, onde $H(.)$ é a função de Heaviside. Séries determinísticas tendem a possuir padrões no recurrence plot, como linhas diagonais paralelas a diagonal principal (que sempre tem valor igual a 1), enquanto séries estocásticas tendem a ter padrões menos visíveis. (Ishii, Rios, & Mello, 2011) (ECKMAN, OLIFFSON KAMPHORST, & Ruelle, 1987)

A figura abaixo mostra o gráfico de recorrência do atractor de Lorenz nota-se a presença de diversas áreas homogêneas internamente, mas separadas por áreas sem qualquer recorrência, indicando a presença de parâmetros determinísticos, mas de variação lenta

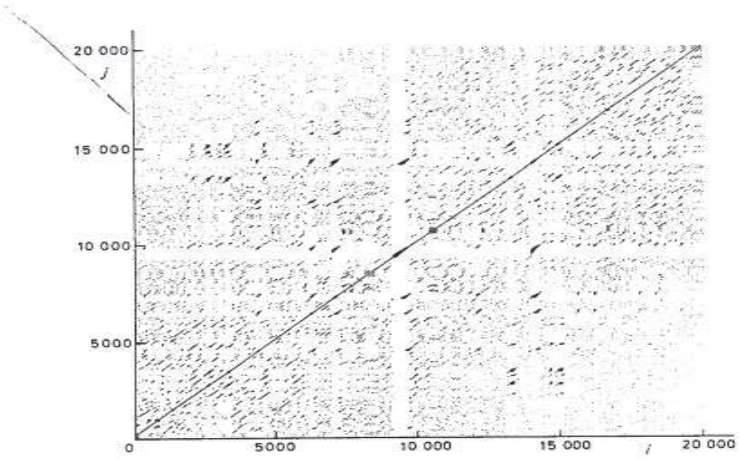

Figura 2: Exemplo de Recurrence plot do atractor de Lorenz com Drift (ECKMAN, OLIFFSON KAMPHORST, & Ruelle, 1987)

A interpretação de recurrence plots pode ser um pouco desafiadora, mas existem padrões conhecidos que podem nos apoiar, abaixo temos uma tabela com alguns exemplos de padrões observáveis e suas interpretações:

Observation	Interpretation
Homogeneity	the process is obviously stationary
Fading to the upper left and lower right corners	nonstationarity; the process contains a trend or drift
Disruptions (white bands) occur	nonstationarity; some states are rare or far from the normal; transitions may have occurred
Periodic/ quasi-periodic patterns	cyclicities in the process; the time distance between periodic patterns (e.g. lines) corresponds to the period; long diagonal lines with different distances to each other reveal a quasi-periodic process
Single isolated points	heavy fluctuation in the process; if only single isolated points occur, the process may be an uncorrelated random or even anti-correlated process
Diagonal lines (parallel to the LOI)	the evolution of states is similar at different times; the process could be deterministic; if these diagonal lines occur beside single isolated points, the process could be chaotic (if these diagonal lines are periodic, unstable periodic orbits can be retrieved)
Diagonal lines (orthogonal to the LOI)	the evolution of states is similar at different times but with reverse time; sometimes this is a sign for an insufficient embedding
Vertical and horizontal lines/clusters	some states do not change or change slowly for some time; indication for laminar states
Long bowed line structures	the evolution of states is similar at different epochs but with different velocity; the dynamics of the system could be changing (but note: this is not fully valid for short bowed line structures)

Tabela 1:Padrões de recorrências e suas interpretações (Potsdam Institute for Climate Impact Ressearch , 2023) (N. Marwan, 2007)

2.3.3 Séries Puramente Aleatórias

Dentro das séries estocásticas, encontramos as séries puramente aleatórias. Definimos estas séries como séries estocásticas que são independentes e identicamente distribuídas, isto é seu valor não depende de nenhuma outra variável, inclusive de valores passados dela própria. Um exemplo deste tipo de série é o ruído branco, já abordado. Séries estocásticas e não aleatórias tendem a possuir valores de auto correlação significativos em mais de um lag e são comumente modeladas por meio de processos autorregressivos. (Shumway & Stoffer, 2016) (Toloi & Morettin, 2018)

Algoritmos de modelagem de séries temporais não são adequados para realizar previsões em séries puramente aleatórias. Existem diversos testes capazes de identificar aleatoriedade, em nosso trabalho utilizaremos o *runs test* (teste de "corridas"): uma "corrida" é definida neste caso como uma sequência de eventos similares precedidos e sucedidos por eventos diferentes, um exemplo é uma sequência de "caras" ou "coroas" no lançamento de uma moeda. Este teste possui como hipóteses nula e alternativa: (Xiannong, 2002)

H_0: O número de corridas é consistente com o esperado em uma variável aleatória

H_1: O número de corridas não é consistente com o esperado em uma variável aleatória.

O número de "corridas" esperado e sua variância é dado por:

$$\mu_a = \frac{2N - 1}{3}$$

Equação 8: Valor esperado do número de runs

$$\sigma = \frac{16N - 29}{90}$$

Equação 9: Variância do número de runs

Onde a é o número de corridas, e N o tamanho da sequência. Para $N > 20$, podemos aproximar a distribuição do número de "runs" por uma distribuição normal padronizada, isto é, podemos obter:

$$Z_0 = \frac{a - \mu_a}{\sigma}$$

Equação 10: Normal Padronizada

E rejeitaremos a hipótese nula se Z_0 estiver fora do nível de significância α determinado. (Xiannong, 2002)

Para o caso de séries temporais consideraremos que uma run será uma de sequência valores maiores ou iguais ou uma sequência de valores menores.

2.3.4 Modelos Lineares Univariados

4.4.5.1 Modelos da família ARIMA

O Uso de modelos ARIMA (Autorregressivos Integrados de Média Movel) é em geral feito com base na abordagem de Box e Jenkins. Esta abordagem consiste em efetuar o ajustamento de modelos autorregressivos integrados (ARIMA) sobre um conjunto de dados. (Toloi & Morettin, 2018)

Entende-se como Autorregressivo de ordem p $(AR(p))$, um modelo cujo valor no tempo t seja dado pelo Ruído Branco ~~mais, mais~~ uma combinação linear dos valores dos p-instantes anteriores, ponderados, sendo os pesos parâmetros do modelo. Ou seja: (Toloi & Morettin, 2018):

$$x_t = \sum_{j=0}^{p} \phi_j x_{t-j} \; + \; \epsilon$$

Equação 11: Modelo Autorregressivo de ordem p

Onde o x dentro do somatório é o valor da série no instante $t - j$ e ϵ é o termo de ruído.

Entende-se como Média móvel de ordem q $(MA(q))$ um modelo cujo valor no tempo t seja dado pela média da série, mais a ponderação linear dos erros verificados até o período q, ou seja: (Toloi & Morettin, 2018):

$$x_t = \mu + \sum_{i=1}^{q} \quad \theta_i \epsilon_{t-1} + \epsilon_t$$

Equação 12: Modelo de média móvel de ordem q

Os parâmetros p e q de um modelo ARMA, podem ser estimados por meio das funções de auto correlação e auto correlação parcial. Definimos como função de auto correlação, a correlação de Pearson entre a série e uma versão com n-lags dela mesma. Esta função pode ser plotada, junto com seu domínio, no chamado gráfico de autocorrelação. Neste gráfico, escolhemos q como o ponto aonde a correlação deixa de ser considerada significativa (isto é, seu valor cai abaixo do nivel signifcância.(Morettin, 2017) (Brownlee, 2020)

A estimativa de p, pode usar um gráfico de auto correlação parcial, este gráfico, como o de auto correlação, se baseia na correlação de Pearson entre a série e seu lag n, mas, ao contrário do gráfico de correlação, subtraímos de cada *lag n* os efeitos de todos os *lags* intermediários entre 1 e n. Quando o valor aonde essa função cai abaixo da significância nos retorna um candidato a p (Toloi & Morettin, 2018) (Brownlee, 2020)

Ambos os termos combinados, geram um modelo do tipo $ARMA(p, q)$. Os parâmetros (coeficientes) deste modelo podem ser estimados por meio de uma função de máxima verossimilhança, aonde minimizamos o erro quadrático entre as estimativas feitas pelo ARMA e a série real.

Este modelo, assim, como o $AR(p)$ possui como premissa que a série seja estacionária séries financeiras, em geral, não são estacionárias, mas tornam-se estacionárias após serem diferenciadas. Após diferenciada até a estacionariedade, modelamos a série por um processo

$ARMA(p, q)$, na sequência, integra-se a série para obter o valor da predição final. (Toloi & Morettin, 2018)

Uma das formas que podemos usar para o número de vezes d que uma série deve ser diferenciada é por meio do teste de Dickey-Fulley. este teste, possui as seguintes hipóteses:

H_0: A Série tem raiz unitária (Não Estacionária)

H_1: A Série não tem raiz unitária (Estacionária)

O processo consiste em estimar um modelo autorregressivo de forma: $\Delta Y_t = \alpha + \gamma Y_{t-1} + \epsilon_t$ e testar a significância do coeficiente γ. A estatística do teste é dada por:

$$\tau = \frac{\hat{\gamma}}{SE(\hat{\gamma})}$$

Equação 13: Estatística do teste de Dick-Fulley

Essa estatística é comparada com valores críticos da Tabela de Dickey-Fuller, se a hipótese nula for rejeitada, conclui-se que a série é estacionária (ou seja $d = 0$), caso contrário, aplica-se uma diferenciação e refaz-se o teste. Este procedimento é repetido até que a estacionarieade tenha sido obtida. O parâmetro d séra dado pelo número de diferenciações necessárias para se obter uma série estacionária.

Uma limitação do teste de Dickey-Fuller é que assume que não existe autocorrelação dos resíduos. Para lidar com esta questão, foi desenvolvida a versão Aumentada do teste de Dickey-Fuller, nesta versão o modelo ajustado é dado por $\Delta Y_t = \alpha + \gamma Y_{t-1} + \sum_{i=1}^{p} \delta_i \Delta Y_{t-1} + \epsilon_t$. A inclusão dessas defasagens reduz o risco de correlação serial nos resíduos tornando o teste mais robusto. (Toloi & Morettin, 2018)

É importante notar que modelos ARMA apresentam algumas premissas para sua correta modelagem. Dentre elas, podemos citar a estacionariedade, já mencionada, como também a homocedasticidade dos resíduos e sazonalidade moderada ou ausente.

Podemos avaliar a homocedasticidade por meio dos testes de Ljung-box/Box-Pierce e Shapiro-WIlk sobre os resíduos, os primeiros servem para identificar se os resíduos são independentes ou se possuem algum tipo de autocorrelação. Já o último avalia se os mesmos seguem ou não uma distribuição normal. (R Core Team, 2021) (Toloi & Morettin, 2018)

Os testes de Ljung Box e Box-Pierce possuem as seguintes hipóteses nula e alternativa:

H_0: As autocorrelações observadas nos dados são nulas, ou de outra forma, os dados são independentemente distribuídos.

H_1: Os dados apresentam autocorrelação e, portanto, não são independentes.

A estatística de teste é dada por:

$$Q = n(n+2) \sum_{k=1}^{h} \frac{\rho_k^2}{n-k}$$

Equação 14: Estatística do teste Ljung box e Box-Pierce

Onde n é o tamanho da amostra, k é o número de lags e o termo ρ é o coeficiente de correlação amostral.

Já o teste Shapiro-Wilk: trata-se de um teste paramétrico, que possui como hipótese nula que uma população é normalmente distribuída e como alternativa a de que não segue esta distribuição. (Mohd Razali & Wah, 2011)

O teste de Shapiro-Wilk é dado pela estatística:

$$W = \frac{(\sum_{i=1}^{n} \quad a_i y_i)^2}{\sum_{i=1}^{n} \quad (y_i - \hat{y})^2}$$

Equação 15: Estatística do teste de Shapiro-Wilk

Onde os valores y_i são os valores da série analisada, $\hat{y}$ é o valor da média amostral, e os vetores a_i são dados por:

$$a_i = (a_1, \dots, a_n) = \frac{m^T V^{-1}}{\sqrt{(m^T V^{-1} V^{-1} m)}}$$

Equação 16: Vetores a do teste de Shapiro-Wilk

Neste cálculo, os valores m são os valores de ordem das variáveis aleatórias i.i.d. (independentes e identicamente distribuídas) e V a matriz de covariância destas estatísticas de ordem normal. A linguagem de programação R possui este teste já incorporado por meio de função específica e o mesmo será utilizado em nossos estudos. (Mohd Razali & Wah, 2011) (R Core Team, 2021)

A Sazonalidade pode ser avaliada visualmente ou por uma adaptação do teste de Friedman, conforme será abordado posteriormente (Toloi & Morettin, 2018)

4.4.5.2 Modelos Sazonais

2.3.4.2.1 Modelos SARIMA

Modelos SARIMA são utilizados quando, após remoção da componente sazonal determinística, ainda existir auto correlação significativa. Este modelo consiste em dois modelos ARIMA acoplados: um para as ordens baixas e outro ajustado ao termo sazonal. (Shumway & Stoffer, 2016) (Toloi & Morettin, 2018)

O termo sazonal consiste em diferenciação em intervalos iguais ao número de estações (por exemplo: para dados anuais no número de anos). Sendo assim, um modelo ARIMA pode ser definido por sete hiperparâmetros, agrupados em duas tuplas, dado por $SARIMA = ARIMA(p, d, q)(P, D, Q, m)$, onde p, d, q são os termos de autorregressão, diferenciação e média móvel, respetivamente, relativos às ordens não sazonais, $P, D\ e\ Q$ correspondem aos mesmos termos nas ordens sazonais e m é o número de passos temporais (*time steps*) constantes em uma estação (por exemplo, para uma série mensal, com período anual, $m = 12$) (Toloi & Morettin, 2018) (Shumway & Stoffer, 2016)

Mais formalmente, o modelo pode ser definido como:

$$\Phi(B^S)\phi(B)\Delta^d\Delta_s^D(x_t) = \Theta(B^S)\theta(B)\epsilon_t$$

Equação 17: Definição de modelo SARIMA

Onde $\Phi(B^S)$, $\Theta(B^S)$ e Δ_s^D, são os termos sazonais (de autorregressão, média móvel e diferenciação respectivamente e $\phi(B)$, $\theta(B)$ e Δ^d são os termos não-sazonais (igualmente

também de autorregressão, média móvel e diferenciação, respectivamente) . (Toloi & Morettin, 2018) (Shumway & Stoffer, 2016)

4.4.5.2.2 Análise de Fourier

A Análise de Fourier é uma ferramenta que objetiva aproximar uma série temporal por uma combinação linear de funções senoidais. Com base nisso, conseguimos encontrar frequências dentro do processo gerador de séries temporais, bem como suas respetivas amplitudes e fases. (Toloi & Morettin, 2018)

A transformada de Fourier, na qual a análise de mesmo nome se baseia, pode ser descrita como o produto interno da função do processo gerador da série temporal e componentes trigonométricas. Isto é, seja um processo gerador de série temporal $f(t)$, a transformada para a frequência λ, ela será dada por: (Narkowich & Boggess, 2009)

$$\hat{f}(\lambda) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{-i\lambda t} dt$$

Equação 18: Transformada de Fourier na forma contínua

Em aplicações computacionais, de uma forma geral não temos o processo gerador $f(t)$, mas sim vetores, na forma $(x_1, x_2, \dots, x_n)$que armazenam o valor da série temporal de forma discreta. Neste caso, usamos a transformada de Fourier em sua forma discreta:

$$\widehat{x_k} = \sum_{j=0}^{n-1} x_j e^{\frac{2\pi i k}{n}}$$

Equação 19: Transformada de Fourier na forma discreta

Esta transformada de Fourier tem complexidade computacional $O(n^2)$, por meio de um dos algoritmos de computação rápida (*Fast Fourier Transform* - FFT) é possível reduzi-la para $O(nlog\ (n))$. (Narkowich & Boggess, 2009)

Feita a aplicação do algoritmo que gera a transformada de Fourier discreta, obtemos um vetor contendo as frequências de cada um dos ciclos senoidais observados, ao plotarmos estes valores em um histograma, obtemos uma um gráfico que consistem em uma aproximação do periodograma (frequência x intensidade) desta série temporal, revelando componentes periódicas. O gráfico do periodograma será melhor detalhado na seção de análise espectral. (Toloi & Morettin, 2018) (Narkowich & Boggess, 2009)

4.4.5.2.3 Análise Espectral

A análise espectral é uma metodologia cujo objetivo é a procura de periodicidades nos dados. Neste trabalho, abordaremos duas possíveis ferramentas: o periodograma, utilizando a transformada de Fourier e o espectro de Hilbert. (Toloi & Morettin, 2018).

Para a criação do periodograma, definiremos como Função densidade espectral de um processo gerador de série temporal, a transformada de Fourier de sua função de autocovariância. Ou seja (Toloi & Morettin, 2018):

$$f(\lambda) = \frac{1}{2\pi} \sum_{\tau=-\infty}^{\infty} \gamma(\tau) e^{-i\gamma\lambda}$$

Equação 20: Transformada de Fourier da função de autocovariância

Esta função pode ser interpretada como uma decomposição da variância do processo. Um pico em $f(\lambda)$ para um valor λ, indica uma contribuição importante para a variância do processo nesta frequência. A este gráfico, denominamos gráfico do espectro. (Toloi & Morettin, 2018)

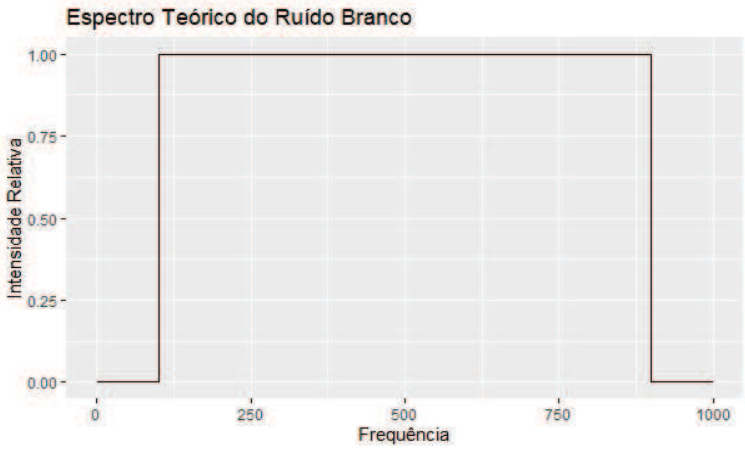

Figura 3:Gráfico do espectro de um ruído branco, nota-se que o valor da intensidade é constante em todo o espectro.

Podemos também construir um periodograma, que consiste em um gráfico da transformada de Fourier discreta do sinal versus a frequência em questão. Este gráfico, por ser instável leva a ser comum efetuarmos suavizações do mesmo.

Tendo o valor das principais frequências a partir de um periodograma, podemos aplicar filtro no sinal. A ideia de um filtro é separarmos do sinal componentes de diferentes naturezas, como por exemplo separar as que possam ser modeladas de forma determinística daquelas estocásticas. De uma forma geral, o filtro é uma transformação L, sobre processos f e g, que deve obedecer às seguintes propriedades: (Toloi & Morettin, 2018) (Narkowich & Boggess, 2009)

Aditividade:

$$L[f + g] = L[f] + L[g]$$

Equação 21: Propriedade da aditividade

Homogeneidade:

$$L[dg] = cL[g], c \in C$$

Equação 22: Propriedade da Homogeneidade

Invariância Temporal:

$$L[f_a(t)] = L[f(t+a)]\ \forall a\ \in\ R$$

Equação 23: Propriedade da invariância temporal

Onde f_a consiste no final f atrasado em a instantes. Existem diferentes tipos de filtros que podem ser aplicados, além daqueles que filtram uma única frequência, como o Passa-Alto (deixa passar as frequências, maiores que uma especifica), o Passa-Baixo (Deixa passar apenas as frequências abaixo), o convolucional (usado em redes neurais para processamento de som e imagem), etc. (Toloi & Morettin, 2018) (Narkowich & Boggess, 2009).

O periodograma pode também ser utilizado na estimativa dos parâmetros de sazonalidade de um SARIMA: por meio dele podemos observar qual(is) as periodicidades existentes nos dados, a fim de aplicar os testes de Friedman, já mencionados. (Shumway & Stoffer, 2016) (Toloi & Morettin, 2018)

O periodograma, baseado na transformada de Fourier, todavia, assume que a série é estacionária e linear. O Espectro de Hilbert, por outro lado, não faz esta assunção. Este tipo de análise espectral se baseia na transformada de Hilbert $H()$ de um sinal $\mu(t)$, dada por: $H\big(\mu(t)\big) = \frac{1}{\pi} P \int_{-\infty}^{\infty} \frac{\mu(t')}{t-t'} dt'$ Onde P é o valor principal de Cauchy.

O Espectro de Hilbert fornece os valores de frequência instantânea e suas intensidades para cada instante de tempo, para cada componente IMF (Intrinsic Mode Function) extraída por uma decomposição de modo empírico (EMD). (Toloi & Morettin, 2018) (Narkowich & Boggess, 2009). (N.E. Huang, 2006)

2.3.5 Modelos para Variância

4.4.6.1 Modelos ARCH e GARCH

Embora os modelos ARIMA sejam uma família poderosa, apresentam como uma de suas premissas a homocedasticidade, que frequentemente é violada nas séries financeiras. Em especial na modelagem da variância condicional dos retornos – volatilidade – são utilizados os modelos ARCH e GARCH. (Morettin & Toloi, 2018)(Morettin, 2017)

Heterocedasticidade pode ser estimada por meio de um teste dos multiplicadores de Lagrange de Engle, este teste envolve em efetuar uma regressão linear do quadrado da série, ou seja:

$$\mu_t^2 = \alpha_0 + \alpha_1 \mu_{t-1}^2 + \cdots + \alpha_r \mu_{t-r}^2 + \epsilon_t$$

Equação 24: Definição dos multiplicadores de Lagrange

E testar as seguintes hipóteses:

$$H_0 : \alpha_i = 0\ \forall i\ \in [1.r]$$

$$H_1\ \exists \alpha_i \neq 0$$

Equação 25: Hipóteses do teste de Lagrange

O que pode ser feito por um teste χ^2 com r graus de liberdade.

Modelos ARCH, ou autorregressivos com heterocedasticidade condicional partem do princípio que a volatilidade depende dos valores passados, por meio de função quadrática, ou seja:

$$\mu_t = \sqrt{h_t}\epsilon_t$$

Equação 26: Modelos ARCH

Onde μ_t é o retorno, ja definido na seção 4.3, ϵ_t é um termo de ruído branco e h_t é dado por:

$$h_t = \alpha_0 + \alpha_1\mu_{t-1}^2 + \cdots + \alpha_r\mu_{t-r}^2$$

Equação 27: Coeficientes h

Daí se nota que um modelo ARCH tem um hiperparâmetro r, do tipo autorregressivo. Para construirmos este tipo de modelo, iniciamos com o teste dos multiplicadores de Lagrange de Engle descrito acima, caso identifiquemos heterocedasticidade, podemos calcular a função de auto correlação parcial de X_t^2 e estimarmos por meio dela a ordem r de um modelo ARCH da mesma forma que faríamos com a ordem p de um modelo ARIMA. (Toloi & Morettin, 2020)(Morettin, 2017)

O modelo GARCH, ou ARCH Generalizado, consiste em um modelo ARCH que considera também os parâmetros h_t passados de forma linear, isto é, um modelo GARCH também apresenta:

$$\mu_t = \sqrt{h_t}\,\epsilon_t$$

Equação 28: Modelo GARCH

Mas h_t é dado por:

$$h_t = \alpha_0 + \sum_{t=1}^{r} \alpha_i X_{t-i}^2 + \sum_{j=1}^{s} \beta_j h_{t-j}$$

Equação 29: definição do coeficiente h_t de um modelo GARCH

Desta forma, enquanto ARCH possui um hiperparâmetro (r) , GARCH possui dois, r e s A estimação do novo parâmetro é em geral considerada difícil, uma forma possível é iniciarmos de forma iterativa, iniciando-se com valores baixos como $(1,1)$, $(1,2)$, etc. e subirmos os valores comparando o Critério de informação de Akaike (*Akaike Information Criterion* - AIC) dos mesmos: O AIC, para modelos GARCH, possui como fórmula: (Toloi & Morettin, 2020)(Diez, Çetinkaya-Russel, & D Barr, 2019) (Nesbitt, 2016)

$$AIC = 2(r + s + 1) + ln\,(L)$$

Equação 30: Akaike Information Criterion (AIC)

Onde r e s são os hiperparâmetros já mencionados e L é a função de verossimilhança. Nota-se que o AIC é um critério que considera tanto a performance do modelo em prever corretamente o comportamento do processo gerador da série temporal, como também se o aumento de complexidade (representado por r e s) se traduz em um melhor ganho. Desejamos modelos com o menor AIC possível. (Morettin, 2017)(Diez, Çetinkaya-Russel, & D Barr, 2019) (Nesbitt, 2016)

2.3.6 Modelos lineares multivariados

Nas seções anteriores tratamos de modelos univariados, aonde tratamos cada série individualmente, em modelos multivariados, dado um conjunto de séries temporais, estamos interessados também nas relações entre as mesmas e como podemos utilizá-la para prever o comportamento umas das outras. (Morettin, 2017)

4.4.7.1 Causalidade de Granger

Clive Granger em seu paper de 1969 define que uma variável causa a outra se a primeira apresenta algum efeito de antecipação ou previsibilidade da última, isto é, sejam duas variáveis X e Y, se o valor atual de Y for melhor previsto usando valores passados de X, dizemos que X causa (ou "Granger-causa") Y. (Morettin, 2017)

Em outras palavras, sejam duas séries X_t e Y_t , A_t o conjunto de toda a informação relevante, com $X_t \in A_t$, $P_t(Y_t|A_t)$um preditor de máxima verossimilhança para Y_t usando A_t e $\sigma^2(Y_t|A_t)$ seu erro quadrático médio. Dizemos que X_t causa Y_t se

$$\sigma^2(A_t) < \sigma^2(Y_t|A_t - X_t)$$

Equação 31: Definição de Causalidade de Granger (*Morettin*, 2017)

Existem diversas formas de operacionalizarmos o conceito acima. Sendo possível fazê-lo por ajustes de modelos VAR ou ARIMA. No R, isso é feito por meio de um teste de Wald, comparando o modelo restrito (isto é, Y_t é explicado apenas por $Y_j, 0 \leq j < t$) com um modelo não-restrito (Y_t explicado também por $X_j, 0 \leq j < t$). Havendo significância, podemos afirmar que existe um efeito de antecipação ou "causalidade". (Morettin, 2017)(Zeileis A, 2002)

2.3.6.2 Cointegração e Correlação

Entendemos por correlação qualquer relacionamento entre duas variáveis x e y, seja ele causal ou não, sendo que a métrica de correlação mais comum é a de Pearson, que mensura o relacionamento linear entre duas variáveis, varia entre -1 e 1 e é dada pela seguinte expressão:

$$\rho = \frac{\sigma_{xy}}{\sigma_x \sigma_y}$$

Equação 32: Correlação de Pearson (*Morettin*, 2017)

Onde ρ é o coeficiente de correlação de Pearson, σ_{xy} é a covariância entre x e y, σ_x e σ_y são os desvios padrões de x e y, respetivamente.

Embora existam séries que possuem relacionamento de longo prazo, é sabido que há processos, principalmente os não estacionários, que podem ser fortemente correlacionados, mas apresentarem relações completamente espúrias. Podemos as séries que realmente

possuem relacionamento de longo prazo por meio da análise de cointegração. (Morettin, 2017)(Toloi & Morettin, 2018)

Definimos duas ou mais séries como cointegradas de ordem (d, b) se ambas forem integradas de ordem d (ou, de outra forma, se obtivermos séries estacionárias ao diferenciarmos d vezes) e se existir uma combinação linear das séries que seja integrada de ordem menor que d. Ao vetor $\beta = (\beta_1, \dots, \beta_n)$ dos coeficientes de combinação lineares dá-se o nome de vetor de cointegração. (Morettin, 2017)

Podemos testar se duas ou mais séries são cointegradas por meio do procedimento de Johansen, este método possui como hipótese nula a ausência de cointegração e como hipótese alternativa a presença de cointegração e envolve estimarmos os vetores de cointegração por meio de um processo iterativo de máxima verossimilhança, estes vetores tem sua significância estimada pelo teste do máximo autovalor e a partir dessa significância toma-se a decisão de rejeitar ou não a hipótese nula (Morettin, 2017)

4.4.7.3 Modelos VAR

Modelos VAR ou autorregressivos vetoriais, consistem em uma extensão do conceito de autorregressão (AR) para modelos multivariados. Neste modelo, uma série temporal pode ser descrita em função dos lags passados próprios e também das demais variáveis. Se tivermos n séries temporais, um modelo VAR de ordem p pode ser descrito como n modelos (um para cada série) autorregressivos, aonde cada modelo tem como variável alvo uma das séries e como variáveis explicativos os lags da própria variável alvo, mais os lags das demais. (Morettin, 2017)

$$X_t = \sum_{i=0}^{n} \sum_{j=0}^{p} \phi_{i,t-j} x_{i,t-j} + \epsilon_t$$

Equação 33: Definição de Modelo VAR

A construção de um modelo VAR parte do pressuposto de que existe relação entre as variáveis estudadas, portanto, é importante a aplicação de um teste de causalidade de Granger e/ou cointegração a fim de confirmarmos se esse pressuposto é válido, do contrário, quaisquer relações que obtivemos com um modelo VAR podem não ser válidas.

De uma forma geral, modelos VAR são ajustados por meio de método de máxima verossimilhança, usando critérios como o AIC. (Morettin, 2017)

2.3.7 Modelos em Espaço de Estados

4.4.8.1 Representação em Espaço de Estados

Modelos em espaço de estado, ou modelos lineares dinâmicos, são uma classe bastante geral de modelos de representação de séries temporais. Estes modelos representam um sistema multivariado por meio das seguintes equações:

$$Z_t = A_t X_t + v_t$$

Equação 34: Equação das Observações

$$X_t = G_t X_{t-1} + \omega_t$$

Equação 35: Equação dos Estados

Onde A_t é a matriz do sistema ou matriz de mensurações, ω_t e v_t são vetores ruído, X_t é a matriz de estados, Z_t sé o vetor de observações e G_t é a matriz de transição (Toloi & Morettin, 2020)

Obtendo uma estimativa da matriz de estados, podemos usar as equações acima para efetuar estimativas das observações futuras. Para isso, podemos usar diversos métodos, tais como o filtro de Kalman e o Teorema de Takens que serão melhor elaborados abaixo. (Toloi & Morettin, 2020) (Takens, 1981)

2.3.7.2 Filtro de Kalman

O Filtro de Kalman é um algoritmo recursivo de estimativa, utilizado para estimar o estado de um sistema dinâmico a partir de uma série de medições sujeitas a ruído. O Nome "filtro" se origina do fato de que a matriz de estado é dada por uma combinação linear do vetor de observações: Por meio da equação do estado, podemos usar as equações de representação do espaço de estados, descritas acima, para fornecer predições de estados desconhecidos. (Toloi & Morettin, 2020)

Este algoritmo opera em duas etapas principais: a etapa de previsão (ou predição) e a etapa de atualização (ou correção). Na etapa de previsão, o filtro estima o próximo estado do sistema com base no modelo dinâmico do sistema, na estimativa anterior do estado e na covariância do estado. Na etapa de atualização, o filtro corrige a estimativa do estado com base na nova medição disponível e na estimativa previamente calculada. O Filtro de Kalman é capaz de fornecer estimativas mais precisas, especialmente quando há incerteza nas medições e no modelo do sistema. (Toloi & Morettin, 2020)

Matematicamente, o Filtro de Kalman pode ser explicado da seguinte forma:

Etapa de predição: Inicialmente se estima o estado atual, usando o estado inicial e inicializando-se as covariâncias (por exemplo com uma diagonal principal 1), para os demais instantes, a predição do estado segue a equação:

$$\hat{X}_{t|t-1} = G_t \hat{X}_{t-1}$$

Equação 36: Equação de predição do Estado

Onde $X_{t|t-1}$ é o vetor de estados no instante t, dada as observações até $t-1$, G_té a matriz de transição de estados.

Em seguida, efetuamos a predição da covariância dos estados:

$$P_{t|t-1} = G_t P_{t-1} G_t^T + Q_t$$

Equação 37: Predição da covariância do processo

Onde $P_{t|t-1}$ é a matriz de covariâncias no instante t, dada as informações até t-1 G_té a matriz de transição de estados e Q_t é matriz de ruídos do processo.

Na sequência passamos à etapa de atualização. Esta etapa envolve a estimação do ganho de Kalman, dado por:

$$K_t = P_{t|t-1}A_t^T(A_tP_{t|t-1}A_t^T + R_k)^{-1}$$

Equação 38: Ganho de Kalman

Onde K_t é o ganho de Kalman , R_t é a matriz de ruídos das observações e $P_{t|t-1}$ A_t são as matrizes definidas no parágrafo e item anterior, respetivamente. Com base no ganho de Kalman, atualizamos os termos de estado e covariância de acordo com as seguintes equações:

$$\hat{X}_t = \hat{X}_{t|t-1} + K_t(Z_t - A_t\hat{X}_{t|t-1})$$

Equação 39: Atualização dos estados

$$P_{t|t-1} = (I - K_tA_t)P_{t|t-1}$$

Equação 40: Atualização da Covariância

O processo se repete indefinidamente. Na existência de variáveis exógenas que podem ser usadas como preditoras, a equação de predição de estado é alterada para:

$$\hat{X}_{t|t-1} = G_t\hat{G}_{t-1} + B_tu_t$$

Equação 41: Equação de predição dos espaços, dada uma variável exógena de controle

Aplicações do filtro de Kalman contextos onde se deseja fazer previsões em tempo real com base em dados observados, sendo frequentemente aplicado em problemas de controle, localização, rastreamento e previsão de séries temporais, bem como predição de volatilidade de portfólios financeiros (Toloi & Morettin, 2020)(Bedendo & Hodges, 2009)

2.3.7.3 Teorema de Takens

O teorema de Takens consiste em uma série de condições que nos permitem reconstruir o espaço de fase de um sistema dinâmico a partir de uma única série temporal observada. Floris Takens propôs este teorema no contexto de detetar “atractores estranhos” (estados para os quais o sistema converge e aparentam ser do tipo fractal) em fluidos no estado de turbulência. (Takens, 1981) (Toloi & Morettin, 2020)

A construção do espaço fase a partir das observações terá qualidade tão melhor quanto mais determinística for a série. Para determinarmos o espaço fase a partir das observações, precisamos definir as dimensões de incorporação/embutimento do mesmo e o número de Atrasos temporais (time-lags) que são relevantes para a predição. (Takens, 1981) (E. Meyer, 2022) (A. Garcia, 2022)

O número de *time-lags* consiste na quantidade de pontos anteriores que devemos utilizar para a predição, já o número de dimensões de *embedding* consiste no número de cópias atrasadas da série que serão usadas para reconstruir o sistema na sua forma de estado de espaço. (A. Garcia, 2022)(Toloi & Morettin, 2020) (E. Meyer, 2022)

O número de time-lags pode ser definido por meio da informação mutua: esta é uma mensuração do relacionamento entre duas variáveis, basicamente calculamos a informação mútua entre a série e cada um de seus time-lags e escolhemos como valor do time-lag o primeiro mínimo local, em uma heurística similar à usada na definição dos parâmetros de um ARIMA. (A. Garcia, 2022) (E. Meyer, 2022)

As dimensões podem ser definidas pelo método de falsos vizinhos próximos (*False Nearnest Neighbors*) após definirmos o número de time lags. Um falso vizinho é um ponto que está na vizinhança de um ponto, mas que deixa de estar quando o espaço é "desdobrado" em outra dimensão. A Heurística neste caso, consiste em desdobrar o espaço em múltiplas dimensões e calcular a fração de falsos vizinhos (número de falsos vizinhos sobre o número de vizinhos totais para um mesmo *épsilon* de raio de distância) para cada uma delas, escolhemos a menor dimensão na qual está fração seja inferior a 20%. (A. Garcia, 2022) (E. Meyer, 2022) (Di Narzo, 2019)

Em seguida, construímos a matriz de estados, que será do tipo:

$$\begin{matrix} f(t) & f(t+d) & \dots & f(t+(m-1)d) \\ f(t+1) & f(t+d+1) & \dots & f(t+(m-1)d+1) \\ f(t+2) & f(t+d+2) & \dots & f(t+(m-1)d)+2) \\ \dots & \dots & \dots & \\ f(t+(m-1)) & f(t+d+(m-1)) & \dots & f(t+(m-1)d+(m-1)) \end{matrix}$$

Figura 4:Exemplo Matriz de Estados obtida com o uso do teorema de Takens (Toledo, 2022)

Na Figura 4, acima, f(t) significa a observação para o período t, m e d são a dimensão de *embedding* e o número de *time delays*, respetivamente.

Uma vez construída, a matriz de estados não possui mais dependências temporais, desta forma, podemos fazer predições para o estado do próximo momento usando algoritmos de Aprendizado de máquina da teoria de Aprendizado Estatístico e obter as observações esperadas por meio da equação das observações. (A. Garcia, 2022) (Toloi & Morettin, 2020)(Hastie, Tibshirani, & Friedman, 2008)

2.4 Análise de Changepoints

2.4.1 Definição de Changepoint

Um *changepoint* ("ponto de mudança") consiste em um ponto no tempo aonde os parâmetros da distribuição da série temporal ou os parâmetros do modelo utilizado para descrever a série repentinamente se alteram. A ocorrência de changepoints torna modelos efetuados na série original descalibrados para predição futura, exigindo uma nova modelagem para que o mesmo continue relevante. (Teixeira, 2012)

A análise de changepoints consiste em detetarmos se ocorreu alguma mudança na série observada e estimar o número de mudanças e suas localizações no tempo. Isto é, seja uma série de n elementos aonde temos funções de distribuição F_i para cada elemento, faremos uma inferência estatística com a seguinte hipótese nula:

$$H_0: F_1 = F_2 = \cdots = F_i = \cdots = F_n \; \forall \; n \; \in [1, n]$$

Equação 42: Hipótese nula de teste de changepoint

E a seguinte hipótese alternativa:

$$H_1: \exists \; i, j \mid F_j \neq \; F_i \; , i, j \; \in [1, n]$$

Equação 43: Hipótese alternativa de teste de changepoint

Onde, na hipótese alternativa i e j não necessariamente são únicos e denotam as posições aonde ocorreram o changepoint (Teixeira, 2012)

A Análise de changepoints pode envolver pontos de mudança conhecidos ou desconhecidos. No caso de pontos de mudança conhecidos, como por exemplo uma alteração de política monetária, desejamos estudar os impactos do mesmo na série. Já na análise de changepoints desconhecidos, objetivamos identificar os pontos nos quais ocorreram mudanças sem conhecimento prévio. Em nosso trabalho, possuímos conhecimento de alguns eventos que podem influenciar as séries (como as crises de 2008, a pandemia de Covid-19 o Plano Real no Brasil, dentre outros), mas focaremos em algoritmos de deteção de changepoints desconhecidos a fim de verificarmos se os mesmos capturam eventos tanto conhecidos como aqueles que podem não o ser. (Teixeira, 2012) (P. Chan, 2021)

Existem numerosas formas de efetuarmos deteção de changepoints. Exemplos são os métodos de somas cumulativas e somas cumulativas iterativas. Em nosso trabalho daremos ênfase a dois métodos: o algoritmo PELT com penalização pelo critério de informação de Schwartz e o Algoritmo SONDE que se baseia em redes neuronais artificiais (Albertini & Mello, 2007) (Killick R, 2022)

Changepoints são um conhecido fator gerador de "débito técnico" em sistemas com algoritmos de *machine learning*, pois fazem com que um modelo treinado no passado rapidamente perca sua capacidade de generalização sobre a série, Séries temporais financeiras são caracterizadas por numerosas mudanças de regime, portanto a capacidade de deteta-los permite uma maior confiabilidade dos mesmos. (Teixeira, 2012) (Google, Inc, 2015)

2.4.2 Changepoint na média

Para um sistema com variâncias constantes, o changepoint na média consiste em efetuarmos um teste de hipótese aonde nossa hipótese nula consiste nas médias de nossa série serem iguais para todos os pontos a um valor único, (conhecido ou não) e a hipótese alternativa em essa igualdade não ser válida para pelo menos um ponto, ou seja: (Teixeira, 2012)

$$H_0: \mu_1 = \mu_2 = \mu_3 = \cdots = \; \mu_n = \mu$$

Equação 44: Hipótese nula de um changepoint para média

$$H_1: \exists k \mid \mu_k \neq \mu_{k+1}$$

Equação 45: Hipótese alternativa de um changepoint para média

Onde k é o ponto aonde ocorreu o changepoint.

Changepoints na média foram inicialmente estudados por meio de métodos de somas cumulativas. Estes métodos consistem em calcular a média das observações iniciais, calcular a diferença entre cada observação e a média calculada a soma cumulativa até o momento. Todas as observações cuja soma cumulativa exceder um determinado valor são consideradas changepoints. (Teixeira, 2012)

2.4.3 Changepoint na variância

Em nosso trabalho estamos interessados especificamente no changepoint sobre as variâncias, as quais são relacionadas a volatilidade. Uma mudança na variância se apresenta em um gráfico como uma descontinuidade aonde a série passa a variar com maior (ou menor) intensidade: (Teixeira, 2012)

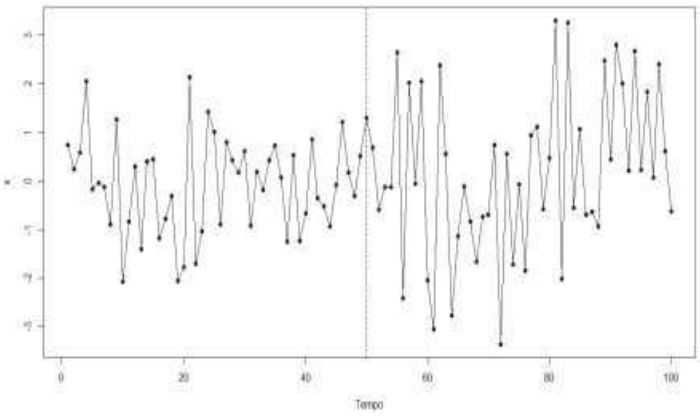

Figura 5:Exemplo de changepoint na variância em uma série independente (Teixeira, 2012)

Para estudarmos changepoints na variância, efetuamos uma abordagem similar ao estudo de changepoints na média, já comentado: efetuarmos um teste de hipótese aonde, neste caso, nossa hipótese nula consiste nas variâncias de nossa série serem iguais para todos os pontos a um valor único, (da mesma forma, conhecido ou não) e a hipótese alternativa em essa igualdade não ser válida para pelo menos um ponto, ou seja: (Teixeira, 2012)

$$H_0: \sigma_1^2 = \sigma_2^2 = \sigma_3^2 = \cdots = \sigma_n^2 = \sigma^2$$

Equação 46: Hipótese nula de um teste de hipótese para variância

$$H_1: \exists k \mid \sigma_k^2 \neq \sigma_k^2$$

Equação 47: Hipótese alternativa de um teste de hipótese para variância

Enquanto para a média uma abordagem clássica envolve a soma cumulativa, para variância podemos utilizar a soma cumulativa dos quadrados iterativa (ICSS). Neste algoritmo a série é ajustada a um modelo estatístico (por exemplo o GARCH) e calculam-se os resíduos, na sequência, remove-se um ponto qualquer e calcula-se a mudança na soma dos quadrados dos resíduos. O ponto que gerar a maior mudança será o primeiro changepoint identificado. Em seguida a série é dividida em segmentos antes e depois do changepoint e o processo é repetido de forma separada em cada um dos segmentos até que não seja mais possível identificar changepoints. Na sequência falaremos sobre demais métodos de deteção de changepoints que serão usados neste trabalho. (Teixeira, 2012) (Killick R, 2022)

2.4.4 O Método PELT com Penalização SIC

Um possível método de identificação de changepoints é o uso da abordagem PELT com penalização pelo critério de informação de Schwartz (SIC). PELT significa "Pruned Exact Linear Time". E consiste nas seguintes etapas: Ajusta-se um modelo (por exemplo, GARCH) utilizando-se como função de custo a soma dos quadrados dos resíduos, mais a penalização SIC. Divide-se a série em dois segmentos e ajustam-se dois modelos para a série. Caso na divisão acima haja diminuição significativa da função de custo, identifica-se um changepoint. O processo é repetido para as demais possíveis quebras até que seja atingido um critério de parada. (Teixeira, 2012) (Killick R, 2022)

O SIC consiste na função de máxima verossimilhança, penalizada pelo logaritmo natural do número de parâmetros, segundo a expressão (Teixeira, 2012):

$$SIC_j = -2\ ln\left(\theta_j\right)\ + p_j ln\ (n)$$

Equação 48: Schwartz information Criterion (SIC)

Onde θ_j é a função de máxima verossimilhança para o modelo j, p_j é o número de parâmetros que tem que ser estimados e n é o número de observações.(Teixeira, 2012)

Tanto o SIC como o critério de informação de Akaike (AIC), tem por objetivo penalizar modelos com maior número de parâmetros, diminuindo-se a chance de um possível *overfit* do modelo. Porém o fato do SIC utilizar $ln\ (n)$ como termo de penalidade (ao contrário do critério de Akaike que utiliza a constante 2), significa que para $n > 7$, teremos $SIC > AIC$ (pois $\ln(8) > 2$). Isto é, a SIC penalizará modelos mais complexos com maior intensidade que AIC (Teixeira, 2012) (Killick R, 2022)

Para o caso de variância e média de série temporal, temos como estimadores de máxima verossimilhança na hipótese nula, respetivamente:

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(X_i - \overline{\mathrm{X}})^2$$

Equação 49: Estimador de máxima verossimilhança da variância

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Equação 50: Estimador de máxima verossimilhança da média

A função de máxima verossimilhança fica:

$$\theta_0(\hat{\mu}, \hat{\sigma}^2) = \prod_{i=1}^{n} \quad \frac{1}{\sqrt{2\pi\sigma^2}} exp\ (\frac{-(X_i - \hat{\mu})^2}{2\hat{\sigma}^2}$$

Equação 51: Função de máxima verossimilhança

E, portanto, o SIC, na hipótese nula, após simplificações resulta em:

$$SIC(n) = n\ ln\ ln\ (2\pi)\ + n\ ln\ (\sum_{i=1}^{n}\ \ \left(X_i - \underline{X}\right)^2) + n + (2-n)ln\ (n)$$

Equação 52: Hipótese nula do SIC

Para a hipótese alternativa, devemos estimar duas variâncias e duas médias, antes e depois do candidato a changepoint, nessa hipótese a função de máxima verossimilhança se torna:

$$\theta_1\left(\widehat{\mu_I}, \hat{\mu}_{II}, {\hat{\sigma}_I}^2, {\hat{\sigma}_{II}}^2\right) = \left(\prod_{i=1}^{n}\ \ \frac{1}{\sqrt{2\pi}{\hat{\sigma}_I}^2} exp\ \left(-\frac{(X_i - \widehat{\mu_I})^2}{2{\widehat{\sigma_I}}^2}\right)\right)\left(\prod_{i=1}^{n}\ \ \frac{1}{\sqrt{2\pi}{\hat{\sigma}_{II}}^2} exp\ \left(-\frac{(X_i - \widehat{\mu_{II}})^2}{2{\widehat{\sigma_{II}}}^2}\right)\right)$$

Equação 53: Função de máxima verossimilhança com os estimadores

Resultando em um SIC:

$$SIC(k) = -2\ ln\ ln\ \left(\theta_1\left(\widehat{\mu_I}, \hat{\mu}_{II}, {\hat{\sigma}_I}^2, {\hat{\sigma}_{II}}^2\right)\right)\ + 3ln\ (n)$$

Equação 54: SIC dado em função do estimador

Rejeitamos a hipótese nula quando temos:

$$min_{2\leq k<n}(SIC(k) = c_a) < SIC(n)$$

Equação 55: Critério do teste

Aonde o coeficiente c_a é tabelado de acordo com o nível de significância desejado. O software R possui o pacote "changepoints" que tem o critério de Schwartz embarcado. (Teixeira, 2012) (Killick R, 2022)

2.4.5 O método SONDE

Outro método de deteção de changepoints é conhecido como SONDE (de "Self-Organizing Neural Network for Detecting Novelties"). Este método, ao contrário do Schwartz, não se baseia na teoria de aprendizado estatístico diretamente e sim no conceito de redes neurais artificiais não supervisionadas. (Albertini & Mello, 2007)

Uma Rede neural, consiste em várias camadas nas quais ocorrem operações matriciais: o vetor de dados de entrada é multiplicado por uma matriz com número de colunas determinado pelas dimensões deste vetor e um número de linhas variável, ao qual chamamos de "neurônios". O resultado destas multiplicações lineares posteriormente é usado como input de uma função não-linear, conhecida como função de ativação. (Géron, 2019)

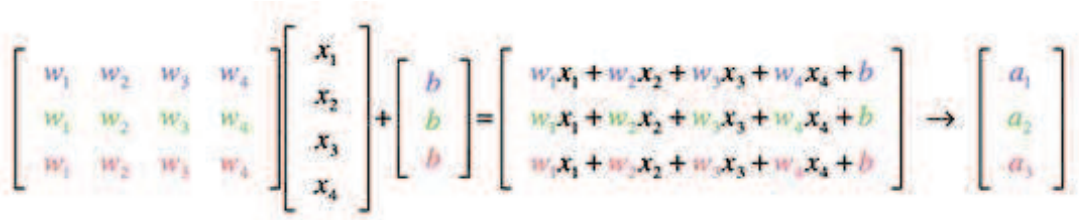

Figura 6:Representação matricial de uma camada de rede neural artificial, a matriz 2 contém os pesos, x é o vetor de entrada e a é o vetor resultante após a aplicação da função de ativação (J Msigwa, 2022)

Na Representação acima, os pesos (W) são definidos pelo método de retro propagação ("Backpropagation"), que basicamente consiste em inicializar a matriz com valores definidos por alguma heurística, fornecer ao algoritmo um conjunto de dados com valores conhecidos de nossa variável de interesse e mensurar o erro entre os valores previstos e os reais. Em sequência mensura-se a contribuição de cada neurônio para o erro, resultando nos gradientes de erros. Ajustam-se os pesos de cada neurônio por este gradiente e o processo é repetido até que um critério de parada seja atingido. (Géron, 2019)

O Treinamento de uma rede neural por meio da retropropagação pode também ser realizado para dados nos quais não tenhamos uma marcação por meio de um mapa de variáveis independentes auto organizável (*Self-Organizing-Maps)*: A rede neural é treinada de forma a gerar como output uma representação bidimensional dos dados. Quando um novo dado é alimentado, a SOM avalia qual neurônio transpõe este dado para a representação bidimensional mais similar. (Kohonen & Honkela, 2007)

A SONDE, consiste em utilizar uma série temporal como dado de entrada e define como changepoint quando nenhum neurônio pré-existente consegue classificar o dado inserido – de acordo com uma métrica de distância – exigindo a criação de um novo neurônio para este mapa. (Albertini & Mello, 2007)

O método SONDE, envolve os seguintes passos: Inicializar uma série de neurônios (centroides) com valores padrão ou aleatórios e calcular a distância entre o ponto e os neurônios existentes e com base nesta distância uma função de ativação dada por:

$$act = exp\left(\frac{||I_t - \overrightarrow{w_c}||}{2\sigma^2}\right)$$

Equação 56: função de Ativação utilizada na SONDE, o numerador da equação consiste na distância ao quadrado entre os centroides e o vetor de entrada e o numerador contém um parâmetro de entrada da função

O Neurônio que possuir a maior ativação (*Best Matching Unit* ou BMU) tem sua ativação comparada com um limiar (*Threshold*), caso a ativação seja inferior ao limiar, o neurônio BMU não foi ativado. Neste caso, considera-se que houve novidade e necessidade de criação de um novo neurônio (centroide) o qual será inserido junto aos demais. (Albertini & Mello, 2007)

Por outro lado, se o BMU for ativado, ele terá as coordenadas de seu centroide alteradas para refletir as características atuais. O processo é repetido até que todos os dados tenham sido analisados pelo algoritmo. (Albertini & Mello, 2007)

O método (ou algoritmo) SONDE necessita receber alguns parâmetros de entrada, são eles: α (taxa de aprendizado ou fator de movimentação), σ (abertura ou variância da função de ativação), ϵ (peso de novo evento) δ (parâmetro de controle) e o threshold (limitar de ativação), desta forma pode ser bastante sensível a essas inicializações. (Albertini & Mello, 2007)

3 Aplicação à Séries Temporais Financeiras

Nas seções anteriores apresentaram-se os principais métodos a serem utilizados na parte prática. Neste capítulo é realizada a componente prática dedicada ao estudo de quatro séries financeiras e algumas das suas relações. Na secção a seguir descreve-se a metodologia e como serão aplicados os métodos aos dados.

3.1 Metodologia

Para cada uma faz séries, faremos uma análise exploratória, de changepoints e uma tentativa de modelagem das mesmas. As análises serão feitas com recurso ao software R, em sua versão 4.1.0 (2021-05-18) por não apresentar conflitos com nenhum dos pacotes utilizados.

Os pacotes utilizados neste trabalho são: *knitr*, para criação de arquivos no formato *markdown*, *ggplot2*, para plotagem de gráficos, *dplyr*, para operações com *dataframes*, *tidyr*, para possibilitar manipulação de números em formato longo, *tseries* e *nonlinearTseries*, que contém métodos para manipulação de séries temporais lineares e não lineares, *stats* para os testes de aleatoriedade, *forecast*, para uso de auto arima, *rugarch* para GARCH, zoo, para criar janelas rolantes, *urca* para obter métodos de cointegração, Rlibeemd que possibilita usar funções para decomposição por modo empírico e, finalmente, *Metrics* para obtermos métricas de avaliação.

A etapa de análise exploratória será realizada individualmente para cada uma das séries originais, aonde apresentaremos visualizações dos dados e algumas medidas descritivas. A seguir, aplicamos estas mesmas técnicas univariadas considerando a série da volatilidade dos dados, que é nosso objetivo principal.

Na sequência, faremos uma análise de *changepoints* das séries, com o objetivo de identificar eventuais pontos de mudança no comportamento das séries, averiguar se estes se relacionam com eventos conhecidos e também sua quantidade no período considerado.

Por fim, utilizaremos as informações obtidas nas análises mencionadas para aplicarmos uma seleção de modelos preditivos adequados às referidas séries. A performance dos modelos é avaliada através de medidas conhecidas como o Erro Absoluto Médio (MAE) e o R^2.

O desenvolvimento da análise das séries será complementado com uma abordagem multivariada, uma vez que estamos interessados em estudar a relação entre algumas séries na relação entre as volatilidades das séries. Nesta etapa faremos estudos para avaliarmos se há alguma relação entre as mesmas que justifique esta abordagem e, caso exista, modelaremos as séries em conjunto. Cada uma destas macro etapas será melhor detalhada a seguir.

3.1.1 Análise Exploratória dos Dados

Nesta etapa, o objectivo é caracterizar as séries temporais em estudo para avaliar o comportamento dos retornos e da respetiva volatilidade, que é a variável de interesse. Primeiramente, procura-se identificar o tipo de processo gerador das séries (determinístico ou estocástico) e se este é linear, não linear ou completamente aleatório, de modo a definir as abordagens mais adequadas. A análise inclui a verificação de padrões repetitivos, recorrendo aos gráficos de autocorrelação e autocorrelação parcial, bem como à construção do recurrence plot. Serão adoptados valores padrão para esta última análise, baseados em estudos anteriores, para facilitar a interpretação.

Além disso, métodos como o teste aumentado de Dickey-Fuller são aplicados para identificar a presença de tendências e, caso necessário, estacionarizar as séries. A análise espectral será utilizada para verificar componentes sazonais ou cíclicas, com o periodograma a servir para explorar as frequências presentes na série. A aleatoriedade será testada com o runs test, e, se for observada, indicará a necessidade de uma abordagem probabilística, que está fora do escopo deste trabalho. Por fim, a análise multivariada incluirá correlação bivariada, cointegração e o teste de causalidade de Granger, sendo todas estas análises aplicadas tanto às séries de preços como às de volatilidade.

3.1.2 Análise de Changepoints

Séries temporais financeiras são conhecidas por apresentarem momentos de mudança abrupta em seu comportamento, desta forma, faremos um estudo de *changepoints* nas séries estudadas. Os estudos envolverão a aplicação dos métodos PELT (com penalidade pelo critério de Schwartz) e SONDE descritos na seção 4.5. Os changepoints encontrados serão exibidos em gráficos de dispersão, juntamente com a série correspondente, para avaliação e, caso possível, serão identificados com algum evento conhecido.

De uma forma geral, percebemos que para a maioria das séries temporais os changepoints fornecidos pelo método SONDE são bastante flutuantes de acordo com a escolha dos parâmetros, sendo que encontrou diversos changepoints para a taxa SELIC (de acordo com a escolha de parâmetros) e nenhum para as demais séries. Dado isso, discorreremos sobre este método em comparação com o método PELT com penalidade SIC apenas para a taxa SELIC e para as demais séries aplicaremos somente o método PELT

3.1.3 Aplicação de Modelagem de séries temporais

Nesta seção aplicaremos as técnicas de modelagem explicadas anteriormente à volatilidade de cada série (daqui por diante, nesta seção, denominada apenas como série).

Iniciaremos nossos estudos efetuando um ajuste de um único modelo ARIMA a cada uma das séries em janela fixa, utilizaremos este ajuste para avaliarmos se os resíduos se

comportam de forma normal e aleatória, utilizaremos para isso os testes de Shapiro-Wilk para normalidade e de ljung-box e box-Pierce para aleatoriedade. Avaliaremos também a ordem do modelo ARIMA do modelo de máxima verossimilhança.

Na sequência, faremos ajustes de vários modelos ARIMA, Garch, além do filtro de Kalman, utilizando predições em janelas móveis, isto é janela de treinamento se deslocará adiante e um novo modelo será ajustado para a predição do próximo valor e assim sucessivamente até predições serem obtidas para todo o dataset.

Na próxima etapa, aplicaremos empirical mode decomposition a cada uma das séries, e o mesmo procedimento de predições em janelas rolantes móveis será aplicado para cada uma das IMF's de forma individual, selecionaremos as predições feitas pelo algoritmo de melhor performance para cada IMF e somaremos estas predições individuais para obtermos uma predição da série completa.

Por fim, serão explorados algoritmos multivariados, considerando o relacionamento entre as séries identificado anteriormente. As predições serão feitas também de forma rolante e os resultados comparados aos anteriores.

Uma vez que desejamos comparar múltiplos resultados de séries temporais, faremos uso da métrica do R^2 ou coeficiente de determinação, dada por:

$$R^2 = 1 - \frac{\sum_{i=1}^{k}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{k}(y_i - \bar{y}_i)^2}$$

Equação 57: R^2 ou coeficiente de determinação

O R^2 mede o percentual da variação explicada pelo modelo e varia de zero a um, um modelo perfeito explicaria totalmente a variância enquanto um modelo totalmente não relacionado não explicaria nada (zero) da variância da série original.

No entanto, há um ponto importante a notar sobre o R^2: trata-se de uma métrica extremamente sensível a *outliers*, desta forma, utilizaremos também a métrica de erro médio absoluto (*Mean Absolute Error* - MAE*),* esta métrica é dada pela média da soma dos módulos dos erros entre o valor predito e o valor observado, ou seja:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}|$$

Equação 58: Erro Absoluto Médio

Desta forma podemos avaliar além da qualidade do modelo se o mesmo apresenta valores outliers em sua predição ou não, de forma quantitativa além de visual.

3.1.3.1 Análise Univariada com ajuste em Janela Fixa

O propósito desta etapa é avaliarmos aplicabilidade de uma abordagem tradicional, utilizando um único modelo ARIMA ajustado à série completa. Para cada série efetuaremos o ajuste do modelo por meio da função auto.arima do software R.

Na sequência, obteremos os resíduos do modelo e o submeteremos a um teste Shapiro-Wilk, este teste tem como finalidade avaliar se os resíduos são normais e é um dos requisitos para haver homocedasticidade do modelo. Caso a hipótese nula seja rejeitada, teremos resíduos não-normais.

A seguir, aplicaremos os testes de ljung-box e box-Pierce para avaliar se há independência dos resíduos, caso a hipótese nula seja rejeitada, teremos pelo menos uma autocorrelação nos resíduos, indicando que os mesmos não são independentes, não havendo homocedasticidade.

3.1.3.2 Análise Univariada com ajuste em Janela Móvel

Nesta etapa é realizado um ajuste com os modelos estocásticos GARCH, SARIMA e o Filtro de Kalman (que possui interpretação determinística e estocástica)) nos datasets das volatilidades e discutiremos as mesmas.

Para o modelo SARIMA, em cada iteração será feito o ajuste de um modelo SARIMA, cujos parâmetros serão definidos por meio do AIC (Akaike Information Criterion): O modelo que apresentar o menor AIC no treino será o utilizado para predição futura.

Um Raciocínio Análogo será feito para o modelo GARCH, mas otimizaremos com base no BIC (Bayesian Information Criterion), de forma análoga ao AIC, a combinação de coeficientes que apresentar o menor BIC no treino será o utilizado para predição futura.

Já no caso do filtro de Kalman, seguiremos com o algoritmo recursivo iterativo definido anteriormente.

5.1.3.3 Análise com EMD com ajuste em Janela Móvel

Nesta seção aplicaremos a análise de Decomposição por Modo Empírico (Empirical Mode Decomposition - EMD) a cada uma das séries estudas, nossa metodologia envolverá a decomposição das séries em suas respetivas Funções de Modo Intrínseco – (*Intrinsic Mode Functions* - IMF's), a seguir um modelo SARIMA será ajustado de forma rolante a cada uma das IMF's, de forma análoga ao que foi feito para a abordagem univariada.

Na sequência as predições obtidas serão somadas e este valor será assumido como a predição utilizando EMD. Apresentaremos os resultados para cada uma das séries.

3.2 Análise das Séries Financeiras

Nesta seção, são apresentados os resultados obtidos com a aplicação da metodologia exposta anteriormente, para cada uma das séries e para a análise multivariada

3.2.1 Taxa SELIC

A Taxa SELIC é a taxa básica de juros da economia brasileira. Ela consiste na taxa de juros média praticada nas operações compromissadas (Repo-Repurchase Agreement) com títulos públicos federais com prazo de um dia útil. O Banco Central do Brasil realiza operações no mercado de títulos públicos para que a taxa Selic efetiva esteja em linha com a meta da taxa Selic, que é definida pelo Comitê de Política Monetária (Copom). (Banco Central do Brasil, 2024)

Para fins de referência, na figura 7, abaixo, apresentamos uma tabela contendo valores mensais para algumas taxas de juro soberanas, inclusive SELIC:

Country	Last	Previous	Reference	Unit
Japan	-0.1	-0.1	Jan/24	%
Switzerland	1.75	1.75	Dec/23	%
China	3.45	3.45	Feb/24	%
South Korea	3.5	3.5	Jan/24	%
Singapore	3.64	3.64	Feb/24	%
Australia	4.35	4.35	Feb/24	%
Euro Area	4.5	4.5	Jan/24	%
Canada	5	5	Jan/24	%
United Kingdom	5.25	5.25	Feb/24	%
United States	5.5	5.5	Jan/24	%
Indonesia	6	6	Jan/24	%
Saudi Arabia	6	6	Dec/23	%
India	6.5	6.5	Feb/24	%
South Africa	8.25	8.25	Jan/24	%
Brazil	11.25	11.75	Jan/24	%
Mexico	11.25	11.25	Jan/24	%
Russia	16	16	Jan/24	%
Turkey	45	42.5	Jan/24	%
Argentina	100	100	Dec/23	%

Figura 7:Últimos valores disponíveis de Taxas básicas de juros de diversas economias, explicitadas em percentuais ao ano. A SELIC se encontra na quinta linha, de baixo para cima ("Brazil") (Trading Economics, 2024

A seguir, apresentaremos algumas visualizações da taxa Selic, abrangendo o período de 1986 até os 2023, utilizando gráficos de dispersão e histogramas. A Figura 8 evidencia valores nulos a partir da metade da década de 1990. Isso ocorre devido aos valores significativamente mais altos (ordens de grandeza diferentes) no período anterior a 1994. Além disso, observa-se uma descontinuidade na série nesse mesmo ano. Ambos os fenômenos são consequência do “Plano Real”, que envolveu uma mudança de moeda e a implementação de um novo arcabouço macroeconômico, com o objetivo - bem-sucedido - de estabilizar a inflação massiva que prevalecia até então.

Além disso, observamos no histograma, figura 9 abaixo, valores concentrados em zero (consistentes com o esperado para um valor dado em pontos percentuais, mas também uma cauda a direita com valores na ordem de 1e4 por cento, também fora do que vemos nos exemplos da tabela da figura 7.

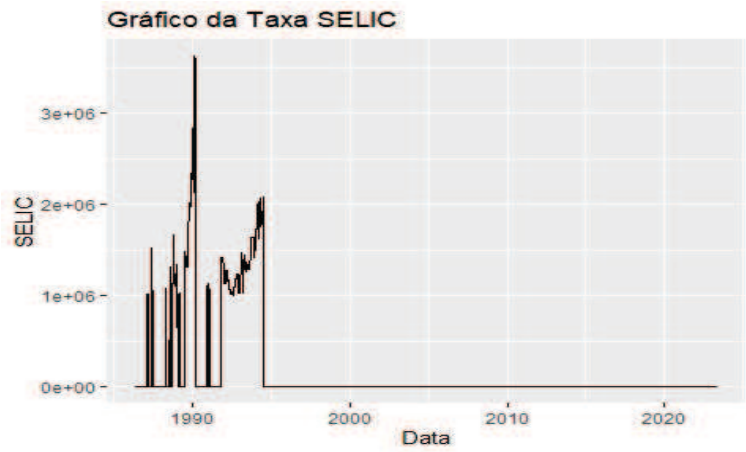

Figura 8:Visualização da taxa SELIC em pontos percentuais ao longo do tempo, aonde se observam a posição dos valores altos e descontinuidades na mesma

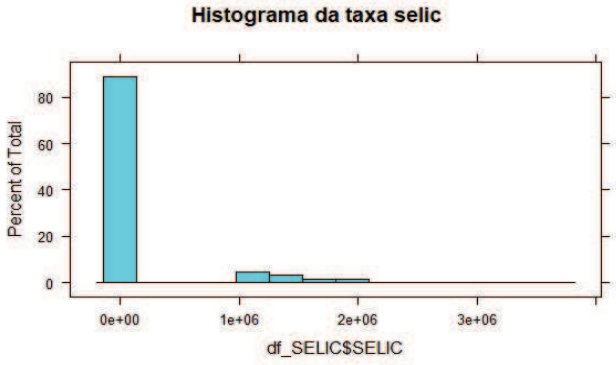

Figura 9:Histograma da Taxa SELIC

Devido a estas distorções, e porque a realidade passou a ser efetivamente outra, passa-se a considerar apenas os dados pós- primeiro de julho de 1994 que foi a data de implantação da nova moeda brasileira que estabilizou a inflação. Nestes gráficos podemos observar valores mais próximos aos esperados para uma taxa de juros:

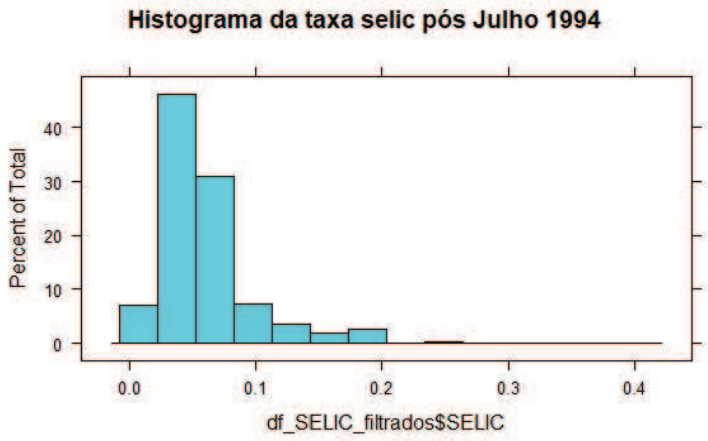

Figura 10:Histograma da taxa Selic no período pós Julho de 1994

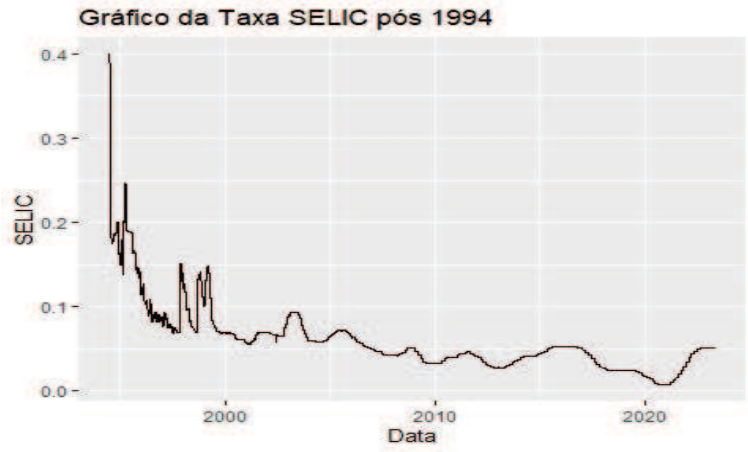

Figura 11:visualização da taxa SELIC em função do tempo, aonde observamos que a mesma apresenta uma tendência decrescente.

No histograma 10, observamos valores concentrados entre 0 e 20%, valores dentro do esperado, com outilers nos 40% (próximos aos valores das taxas de juros de Turquia e Argentina na Tabela 10, as quais passavam por processo inflacionário quando da obtenção da tabela). No gráfico 11, notamos um comportamento consistente com o de uma economia de alta inflação com trajetória de estabilização: partimos de taxas altas, como 40%, mas que foram sendo paulatinamente reduzidas até chegarem a valores entre 0 e 10% nos anos 2010 e 2020. Visualmente, a presença de grandes flutuações iniciais, em contraste com o comportamento menos extremo posterior sugere uma não estacionariedade, a qual será avaliada pelo teste aumentado de Dickey-Fuller.

Devido aos efeitos do plano real, que introduzem mudanças de ordem de grandeza nos valores da SELIC, as próximas análises desta nesta seção consideram apenas o período pós-1994.

Na sequência, faremos um gráfico de recorrência, com o objetivo de apoiar o diagnóstico do caráter da série temporal da Selic, no período em que estamos interessados.

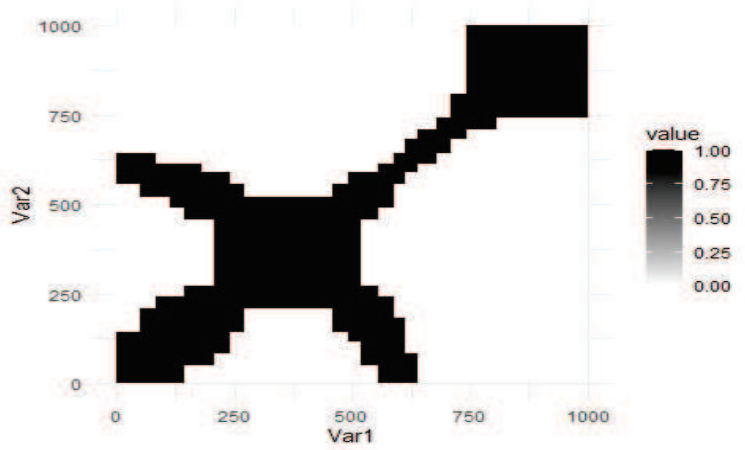

Figura 12:Gráfico de recorrência da taxa SELIC, aonde se observam regiões homogêneas ao centro e no canto superior direito

O Gráfico 12, de recorrência, mostra uma linha ortogonal à diagonal principal, bem como regiões homogêneas ao centro e no canto superior direito, o que sugere que os estados evoluem de forma similar em períodos distintos, podendo ser determinístico, mas com clusters homogêneos ao centro e ao canto superior direito, indicando que estes estados são estacionários ou mudam de forma lenta, o que pode sugerir algum tipo de comportamento cíclico ou sazonal.

É bastante conhecido que séries de taxas de juros são cíclicas, seguiremos então com uma análise espectral para identificar ciclos:

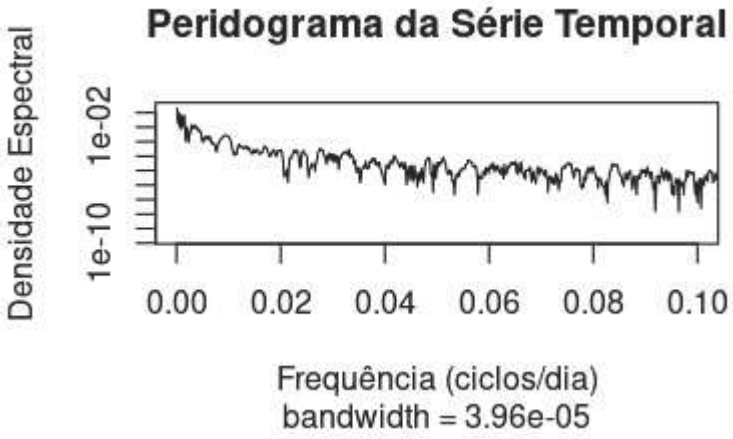

Figura 13:Periodograma da série temporal da SELIC, em ciclos por dia

Observamos que a densidade espectral é maior nas frequências mais baixas. Consistente com periodicidades mais longas (maiores do que 500 dias), características de ciclos econômicos, aos quais a taxa de juros está intimamente relacionada. No entanto, a exibição de frequência causa alguma dificuldade de interpretação, mostraremos aqui os períodos de maior densidade espectral em formato de tabela:

periodo_dias	densidade_espectral
7290.00000	0.2110075733
2430.00000	0.1040713899
1041.42857	0.0658482156
662.72727	0.0652836848
1458.00000	0.0643197639
810.00000	0.0536685097
911.25000	0.0271826387
3645.00000	0.0190617176
347.14286	0.0138132148
270.00000	0.0132107318
316.95652	0.0129052382
1822.50000	0.0117664770
280.38462	0.0116329563
729.00000	0.0106039852
260.35714	0.0101676549

Tabela 2:Periodo, em dias do ciclo e sua densidade espectral da taxa SELIC, os ciclos estão ordenados de forma decrescente, por espectro

Observamos que os ciclos mais intensos possuem períodos longos, da ordem de até 20 anos (7290 dias), mas passando por ciclos de 6 anos (2430 dias). O Teste de Friedman, para os períodos de 346 dias, 1041 dias e 2430 dias, resultaram em p-valores de $1.028 * 10^{-7}$, $2.2 * 10^{-16}$ e $2.2 * 10^{-16}$. Ambas as informações juntas indicam a existência de ciclos, sendo que os de mais longo prazo são mais pronunciados.

Densidades espectrais maiores em ciclos mais longos também podem sugerir componentes de tendência. A presença dos mesmos pode ser avaliada por meio do teste de Dickey-Fuller: (Toloi & Morettin, 2018) (Mankiw, 2015)

```
##
##  Augmented Dickey-Fuller Test
##
## data:  df_SELIC_filtrados$SELIC
## Dickey-Fuller = -4.8863, Lag order = 19, p-value = 0.01
## alternative hypothesis: stationary
```

Observamos que o teste de Dickey-Fuller rejeitou a hipótese nula de presença de raiz unitária. O que contrasta com o observado na figura 11. Provavelmente temos uma série que, embora apresente grandes flutuações iniciais, tende a estacionariedade. Desta forma, consideraremos a série da taxa SELIC como estacionária dentro do período estudado. Seguiremos com a elaboração dos gráficos de autocorrelação e autocorrelação parcial, para suportar a identificação do processo gerador.

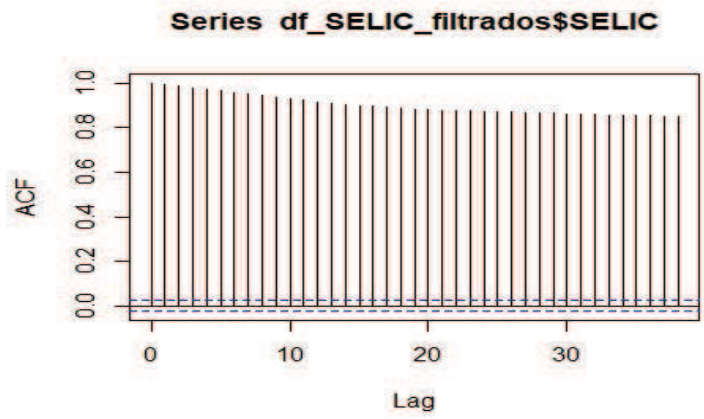

Figura 14:Gráfico de Autocorrelação da taxa SELIC, aonde observamos autocorrelações significativas para diversos lags

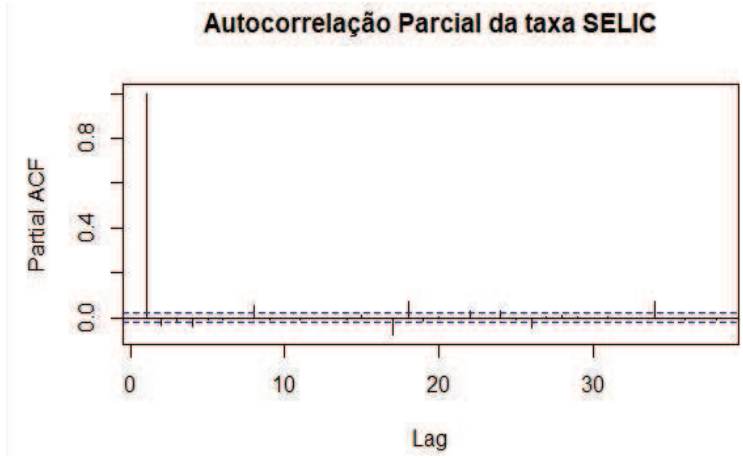

Figura 15:Autocorrelação parcial da taxa SELIC, aonde observamos lag 1 como o único relevante

O gráfico da autocorrelação indica autocorrelação significativa para todos os lags mostrados. No entanto, o gráfico de autocorrelação parcial, mostra apenas o primeiro lag como relevante, sendo os demais residuais. Isso sugere uma série AR(1), potencialmente determinística ou mesmo completamente aleatória.

Para avaliar se temos uma série aleatória, exibiremos os resultados do Runs tests abaixo:

```
        Runs Test

data:  Binarize_Factorize(df_SELIC_filtrados$SELIC)
Standard Normal = -22.016, p-value < 2.2e-16
alternative hypothesis: two.sided
```

O Runs Test rejeitou a hipótese nula de aleatoriedade, em conjunto com as demais conclusões, observamos uma série com um forte componente determinístico e existência de comportamento cíclico de longo prazo. Estes fatos sugerem que, embora a SELIC seja modelável, provavelmente exigirá uma abordagem determinística e/ou aplicação de técnicas de dessazonalização. Além disso, a grande descontinuidade

observada em 1994 pode esconder outros changepoints os quais poderiam ser analisados.

5.2.1.2 Volatilidade SELIC

Seguiremos na sequência com o cálculo da volatilidade da taxa SELIC e visualizaremos apenas os dados pós outubro de 1994, a fim de eliminar os efeitos de hiperinflação. Como será visto, há um pico significativo no início do gráfico 16, referente a troca de moeda, o gráfico 17, contém dados apenas do século XXI, aonde este efeito não aparece:

Figura 16:Gráfico da volatilidade da taxa SELIC pós julho de 1994, o pico a esquerda se deve a queda abrupta no valor da mesma quando do início do plano real

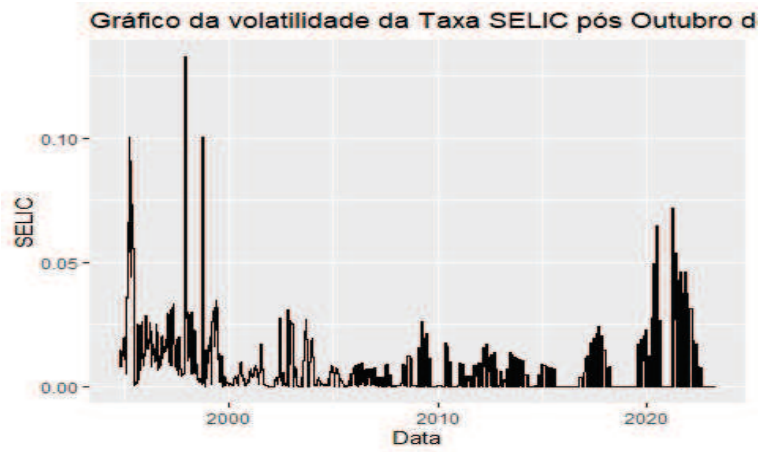

Figura 17:Volatilidade da taxa SELIC pós outubro 1994 (desconsiderando o efeito do plano real) aonde se observam os valores posteriores em uma escala menor

Podemos observar que a volatilidade da taxa SELIC se apresenta bastante elevada nos anos finais da década de 90, tornando-se menor nas duas primeiras décadas do século XXI e voltando a subir nos anos finais da década de 2010 e inicias de 2020 (mas com picos menores do que os dos anos 90). Em Ambos os gráficos, visualmente há indícios de estacionariedade. Iniciaremos nossa análise com um gráfico de autocorrelação e autocorrelação parcial:

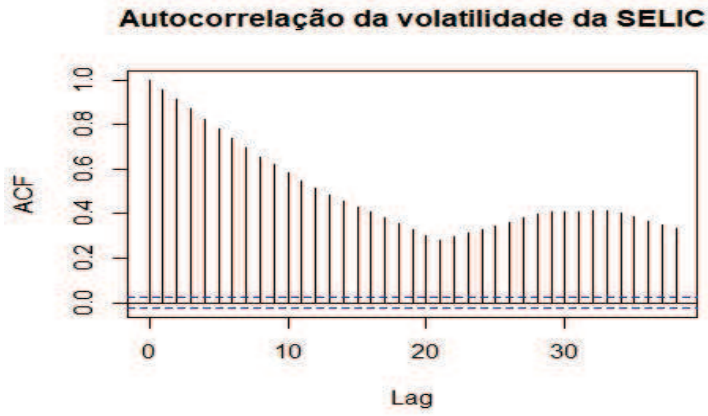

Figura 18 da volatilidade da taxa SELIC

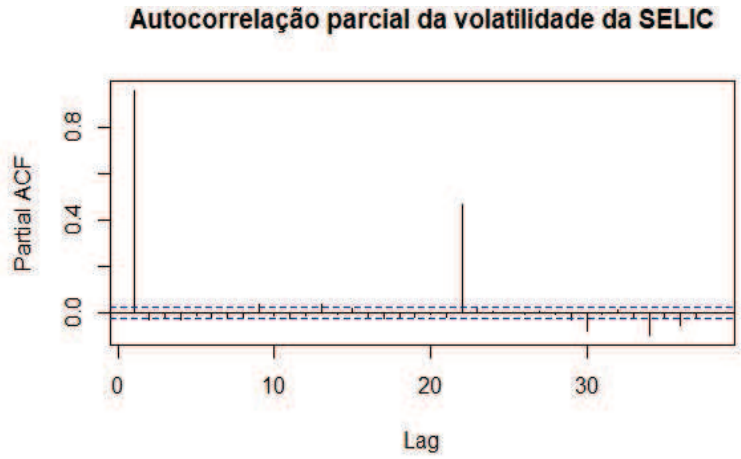

Figura 19 parcial da volatilidade da SELIC

Observamos no teste de autocorrelação que existem lags relevantes por toda a extensão, havendo um decréscimo da relevância do lag 1 ao 20, com posterior crescimento do 20 ao 30, o Gráfico de autocorrelação parcial, por sua vez, apontou o lag 22 (correspondente ao período de cálculo da volatilidade) além do lag 1 como relevante. Desconsideraremos o Lag 22, pois sua relevância é devido a forma de construção da volatilidade (que utiliza uma janela rolante de 21 dias úteis) e a consideraremos como uma série de autocorrelação parcial igual a 1. A combinação das duas informações sugere uma série autorregressiva de ordem 1. Seguiremos com uma análise dos gráficos de recorrência:

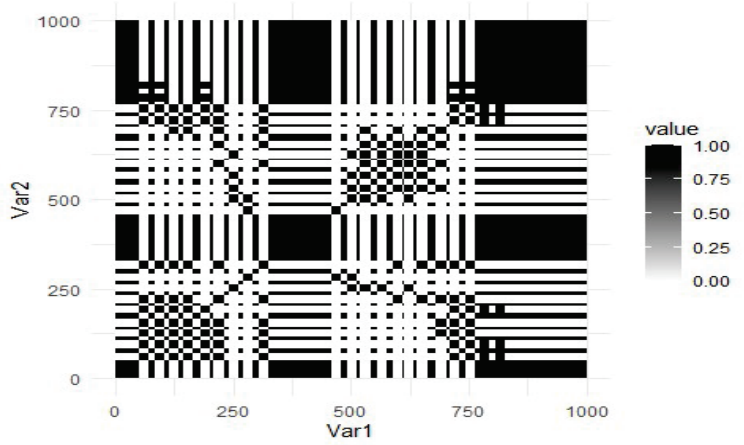

Figura 20:Gráfico de Recorrência da Taxa SELIC

O gráfico de recorrência mostra a presença de linhas e padrões ocorrendo de forma periódica. O que pode ser indicativo de comportamento cíclico, o qual sabemos existir na série original. O teste *Runs* para aleatoriedade indica que a série não possui caráter aleatório, como se deduzia expectável

```
        Runs Test

data:  Binarize_Factorize(na.omit(df_SELIC_filtrados)$volatilidade)
Standard Normal = -25.837, p-value < 2.2e-16
alternative hypothesis: two.sided
```

Em conjunto com os observados no recurrence plot e na análise de autocorrelação, sugere-se a presença de uma componente cíclica, como ocorre na própria taxa SELIC ou em alternativa a ocorrência de changepoints que possam alterar o processo gerador da série.

Faremos agora um periodograma da volatilidade:

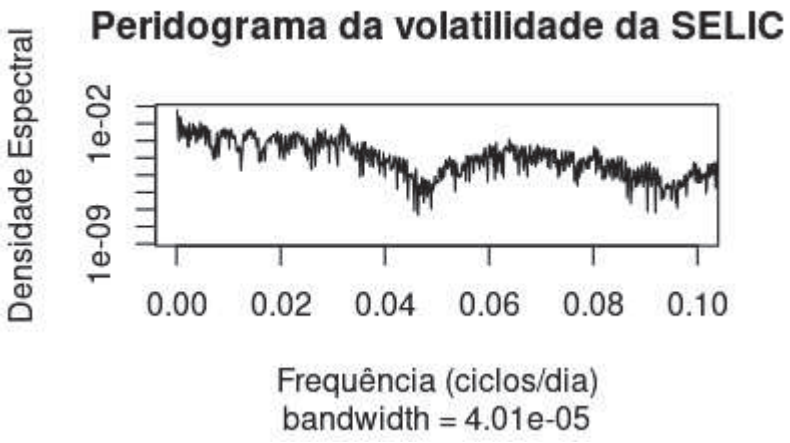

Figura 21:Periodograma da Volatilidade da SELIC

Notam-se no periodograma a presença de ciclos de baixa frequência, indicando a necessidade de tratamento para remoção da periodicidade. Da mesma forma que fizemos para a taxa Selic, plotaremos uma tabela com os períodos e suas densidades espectrais:

periodo_dias	densidade_espectral
7200.00000	0.061802078
1440.00000	0.023966297
1200.00000	0.019110751
1800.00000	0.016294586
3600.00000	0.013296591
218.18182	0.011384954
400.00000	0.010498211
313.04348	0.009586474
31.44105	0.008151266
720.00000	0.007883261
189.47368	0.007687376
900.00000	0.007533857
211.76471	0.007161355
31.57895	0.006882466
184.61538	0.006858538

Tabela 3:Periodo, em dias do ciclo e sua densidade espectral da volatilidade da taxa SELIC, os ciclos estão ordenados de forma decrescente, por espectro

De forma parecida ao que observamos na própria taxa, observamos a presença de ciclos longos de 10, 15 ou 20 anos como sendo os mais relevantes. O teste de Friedman para alguns dos períodos mencionados na tabela:400 dias, 1800 dias e 3300 dias. Nestes testes observamos p-valores de 0.22, praticamente zero e 0.00289, respetivamente. Tais fatos indicam a existência de componentes cíclicas de longo prazo, mas provavelmente não de curto prazo.

Concluímos então que a série da volatilidade SELIC parece ser uma série modelável, embora ainda seja necessário avaliar qual a melhor forma futuramente.

5.2.1.3 Changepoints na variância da taxa SELIC

Apresentamos abaixo os resultados de changepoints na variância da taxa SELIC segundo o método PELT-SIC:

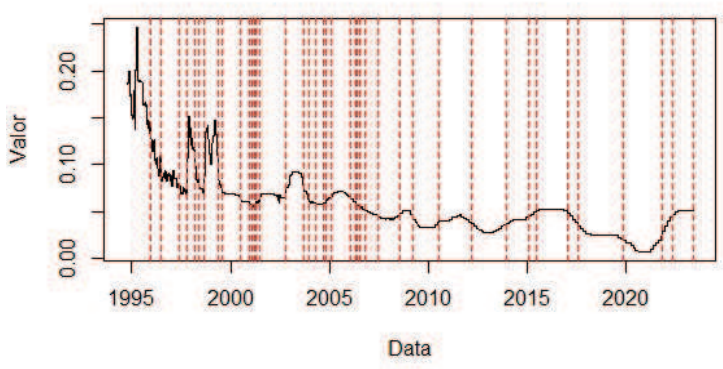

Figura 22:Changepoints na variância da SELIC calculados por meio do método PELT com penalidade SIC, plotados contra a própria taxa

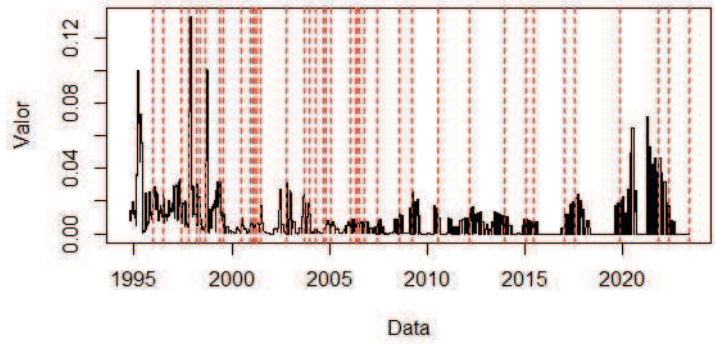

Figura 23:Changepoints na taca SELIC, calculados por meio do método PELT com penalidade SIC, plotados contra a variância

Observamos que o Método PELT-SIC apresentou grande quantidade de changepoints, há uma concentração em alguns eventos conhecidos (crise da Ásia ao final de 1997, eleição presidencial brasileira de 2002, crise do subprime em 2008), mas há também diversos changepoints apontados em momentos de relativa tranquilidade.

A seguir, mostraremos os resultados de changepoints plotados pelo método SONDE, os parâmetros utilizados foram: α (fator de movimentação do cluster) = 2.01, σ (abertura) =0.9, ϵ (peso dado para um novo evento na cadeia markov)=0.9, δ (parâmetro de controle interno)=1e-13 e threshold (limiar de ativação)=1e-5.

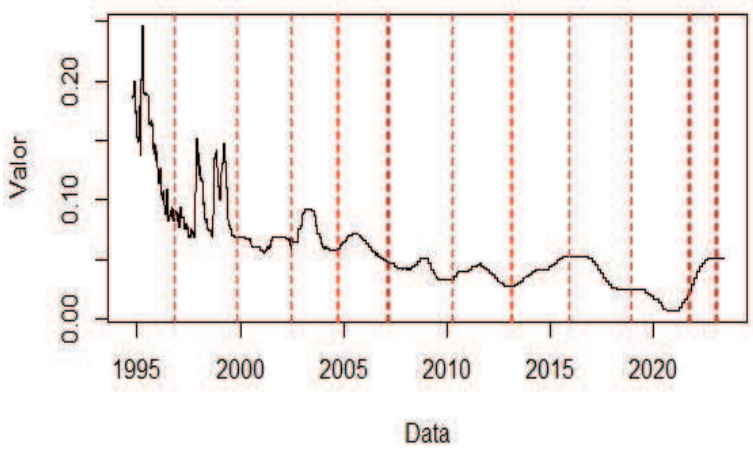

Figura 24:Changepoints na taca SELIC, calculados por meio do método SONDE, plotados contra a própria taxa

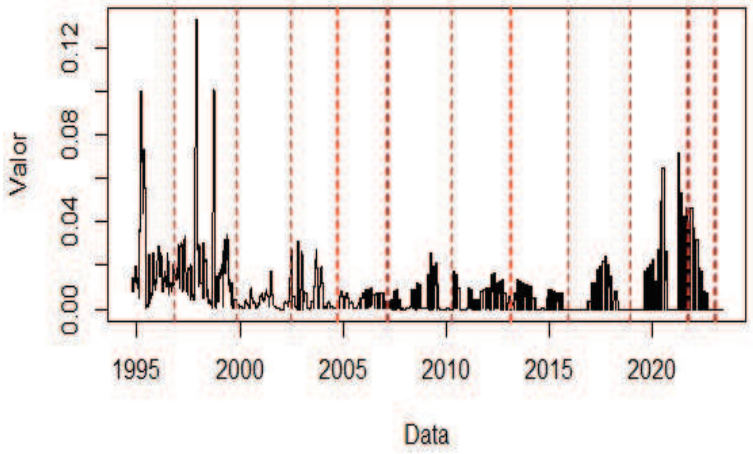

Figura 25:Changepoints na taca SELIC, calculados por meio do método SONDE, plotados contra a variância

Com os parâmetros utilizados neste momento, observamos que o método SONDE detetou changepoints em intervalos espaçados de forma aproximadamente igual, parte deles são consistentes com períodos conhecidos de crises, como o final de 1996 (próximo a crise da Ásia). 2002 (Bolha Ponto Com), 2007 (próximo a crise do subprime), 2013-2015 (período de instabilidade política no Brasil, com o Impeachment da então presidente Dilma Rousseff) e 2021-22 (final do COVID-19) e em menor quantidade que o PELT-SIC

Para este dataset em especifico, SONDE parece ter um desempenho interessante, sendo capaz de identificar changepoints possivelmente correspondentes a ciclos macroeconômicos.

3.2.1.3 Modelagem Univariada da volatilidade da Taxa SELIC

5.2.1.4.1 Ajuste em Janela Fixa

O ajuste de um modelo ARIMA na volatilidade da SELIC resultou em um modelo ARIMA de ordem (0,1,0), isto é um processo de passeio aleatório. Apresentaremos abaixo os resultados dos testes de Shapiro-Wilk, Ljung-Box e Box-Pierce:

```
##
##  Shapiro-Wilk normality test
##
## data:  amostra
## W = 0.16858, p-value < 2.2e-16

##
##  Box-Ljung test
##
## data:  residuos
## X-squared = 27.995, df = 10, p-value = 0.001809

print(lb_test)

##
##  Box-Pierce test
##
## data:  residuos
## X-squared = 27.957, df = 10, p-value = 0.001834
```

Observamos aqui que todos os testes rejeitaram suas hipóteses nulas, isto é, os resíduos são não-normais e não-aleatórios. Violando a premissa de homocedasticidade do modelo ARIMA. Isso, aliado ao fato de o auto-arima ter apontado o passeio aleatório como a ordem de máxima verossimilhança indica que um único modelo ARIMA não é capaz de se ajustar de forma satisfatória à série. Seguiremos agora com as demais séries de volatilidade.

5.2.1.4.2 Ajuste em Janela móvel

Abaixo apresentamos os resultados que tivemos com as predições de volatilidade SELIC:

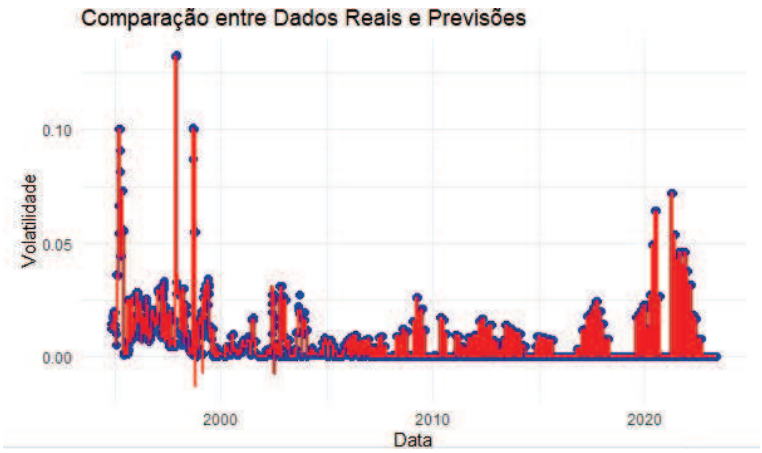

Figura 26:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade da taxa SELIC, utilizando um modelo SARIMA

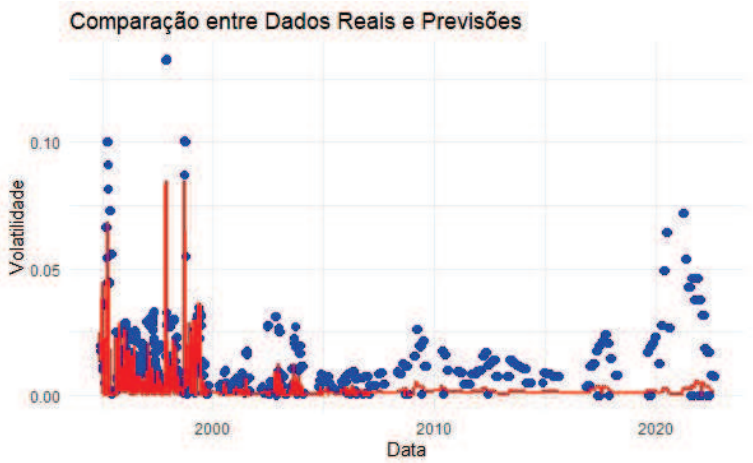

Figura 27:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade da taxa SELIC, utilizando um modelo GARCH

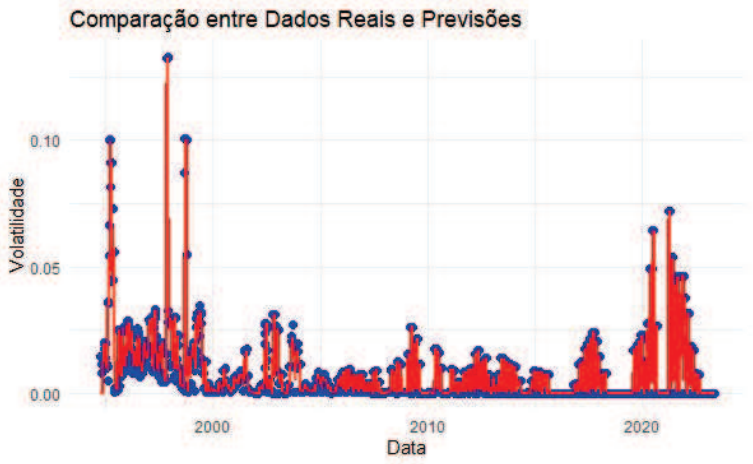

Figura 28:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade da taxa SELIC, utilizando um Filtro de Kalman

Obtivemos um R^2 de 0.8294 para o modelo SARIMA, aproximadamente zero para o GARCH e de 0.8163 para o Filtro de Kalman. O MAE do SARIMA e Filtro de Kalman foi de 0,001480 e 0,00226, respetivamente. Na análise exploratória da volatilidade havíamos observado que existem diversos changepoints na série o que indica que a série como um todo não é identicamente distribuída, mas que existem diferentes regimes. Tanto a abordagem SARIMA em janela móvel como o Filtro de Kalman, apresentaram performances similares e satisfatórias. Além disso, tanto visualmente, como pela boa performance em ambas as métricas, podemos notar que ambos os modelos parecem ter bom desempenho por todo o dataset, sem outliers.

5.2.1.4.3 Análise com EMD

Na análise de volatilidade SELIC obtivemos 11 IMF's, mais a residual, a somatória das previsões ajustadas para cada modelo SARIMA rolante resultou no seguinte gráfico:

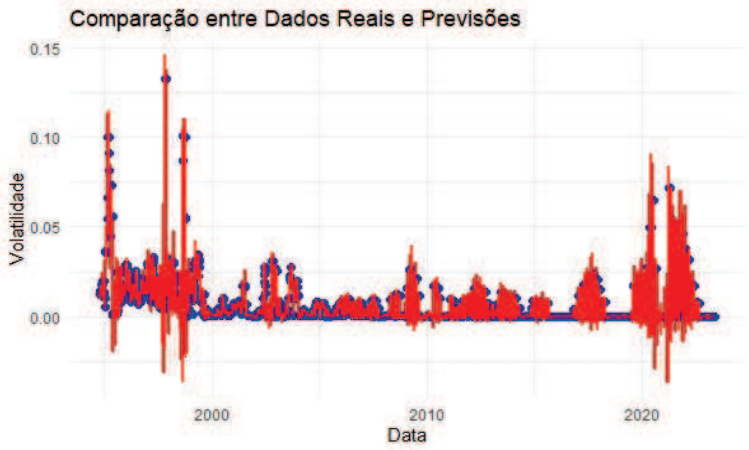

Figura 29:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade da taxa SELIC, utilizando uma combinação de modelos SARIMA pra cada IMF

Obtivemos nesta análise um R^2 de 0,8586 e um MAE de 0,00211, observamos que o R^2 foi ligeiramente melhor e o MAE foi ligeiramente pior do que o observado na análise univariada sem decomposição. Isso nos leva a supor que, no caso específico da SELIC o uso de EMD gerou uma performance em geral ligeiramente pior, mas uma capacidade ligeiramente melhor de lidar com valores extremos (outliers).

3.2.2 Preço do Ouro

O Ouro é historicamente considerado como um porto seguro financeiro, no caso de guerras, insurreições e outras calamidades públicas que podem destruir valor de ativos. Ainda hoje, diversos países, como Portugal, detém um valor expressivo de suas reservas internacionais em ouro físico. (World Gold Council, 2023)

A seguir apresentaremos o gráfico de dispersão e o histograma do preço do ouro, os dados compreendem o período de 1979 até 2023.

Figura 30:Preço do ouro, aonde observamos distintos regimes.

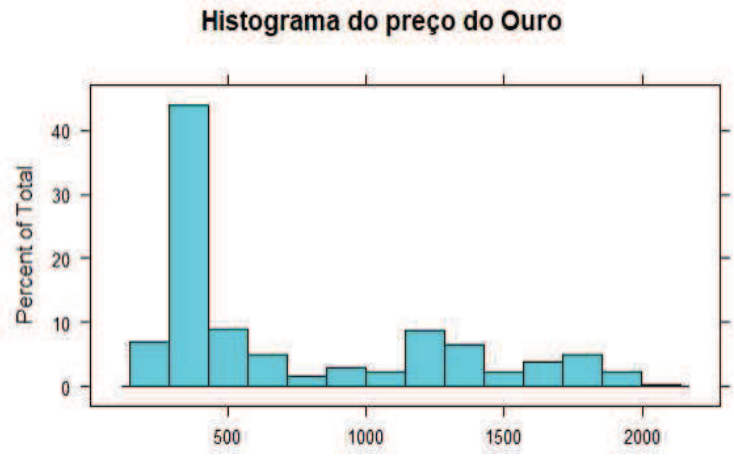

Figura 31:Histograma do Preço do Ouro, observa-se uma cauda longa a direita

O gráfico do ouro apresenta valores mais suaves do que o taxa de juros, mas aparenta regimes distintos nos anos 1980 (quando da crise do petróleo e aumento significativo da inflação a nível global) e 2000 (período de maior afrouxamento quantitativo) em comparação aos anos 1990, nos quais os preços do ouro se comportaram em relativa estabilidade.

Os dados acima, em uma análise visual, são claramente não-estacionários, no entanto, é comum que séries financeiras sejam estacionárias após sua primeira diferenciação, aplicaremos então o teste aumentado de Dick-Fulley na série diferenciada uma única vez:

```
##
##  Augmented Dickey-Fuller Test
##
## data:  na.omit(df_Ouro)$diferenciada
## Dickey-Fuller = -23.233, Lag order = 22, p-value = 0.01
## alternative hypothesis: stationary
```

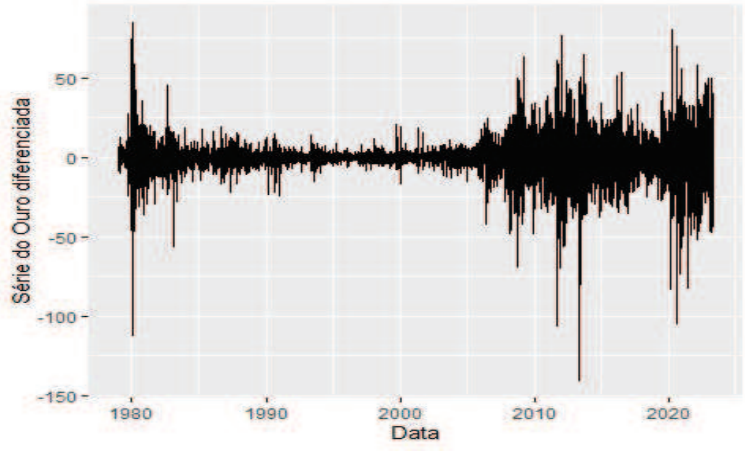

Figura 32 histórica diferenciada do Ouro, podemos observar dois períodos de alta variância na série, separados por um de relativa calmaria

A série do ouro diferenciada apresenta maiores valores absolutos próximos ao período de 1980 Até aproximadamente 1984, ficando - em termos relativos - mais baixos do final da década de 1980 até o final dos anos 2000. O período de 1980 é

consistente com crises econômicas mundiais devido ao choque do Petróleo, crise da dívida latino-americana.

Já nos anos 2000-2010 observamos a crise do suprime (2008) e uma década dominada por juros baixos e afrouxamento quantitativo nas principais economias. Isto sugere um padrão na série histórica do ouro que possivelmente é modelável. Faremos uma análise de auto correlação na sequência como parte de uma investigação do caráter (se estocástica, determinística ou aleatória) da série. (Toloi & Morettin, 2018) (Mankiw, 2015)

Figura 33: Autocorrelação da série original do ouro, aonde observamos um padrão constante de auto correlação

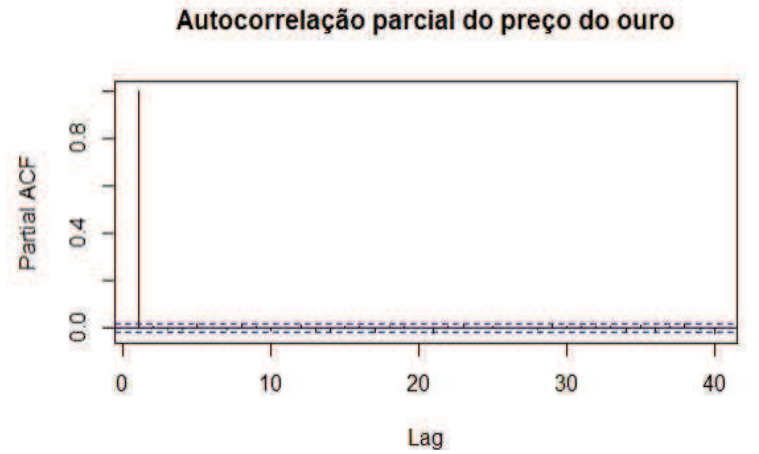

Figura 34: Autocorrelação parcial da série original do preço do ouro, aonde vemos apenas o lag 1 como significativo

Observamos um padrão de relevância de todos os lags mostrados no gráfico de autocorrelação da série, o que não ocorre no gráfico de autocorrelação da série diferenciada e apenas com o primeiro lag significativo para a série não diferenciada, no entanto, ao contrário do ocorrido na SELIC, torna-se necessário efetuar a análise para a série diferenciada.

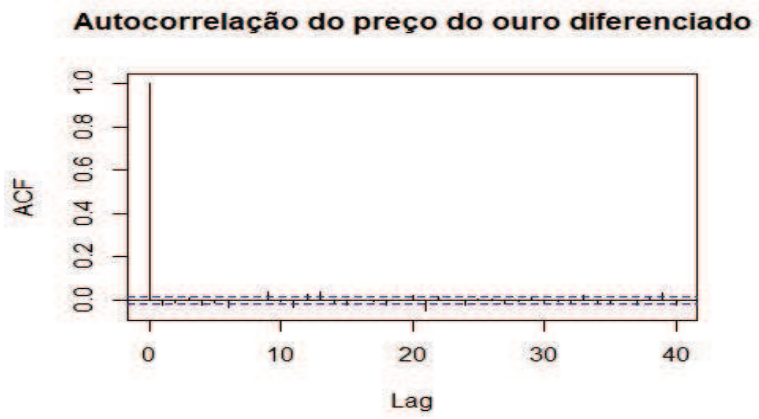

Figura 35:Autocorrelação do preço do ouro diferenciado, aonde se observa a estacionaridade por diferenciação –não havendo autocorrelação significativa

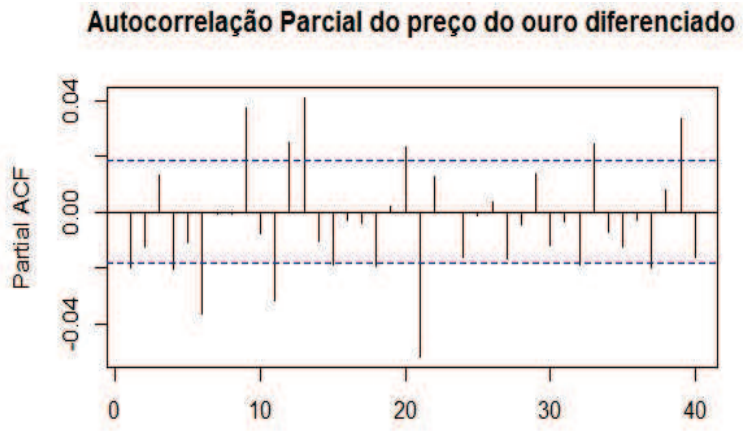

Figura 36:Autocorrelação parcial do preço do Ouro diferenciado

O gráfico de autocorrelação da série diferenciada do ouro não indica autocorrelação significativa, o mesmo acontecendo no gráfico de autocorrelação parcial, isso sugere que estamos tratando de um processo do tipo *random walk*, completamente aleatório, ou um processo MA, integrado de ordem 1. Na sequência, exibiremos o recurrence plot da série original e da diferenciada

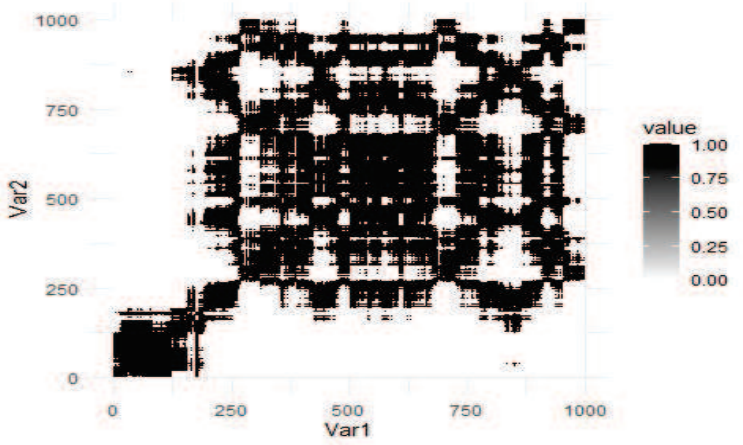

Figura 37:gráfico de recorrência da série original do ouro, podemos observar um cluster ocupando boa parte da centro-direita bem como manchas brancas

O gráfico de recorrência da série mostra um cluster central com algumas faixas brancas, indicando descontinuidades ou ocorrências de transições, indicando potenciais aleatoriedades ou influências não mapeadas. Seguiremos com o Recurrence plot do preço do ouro diferenciado

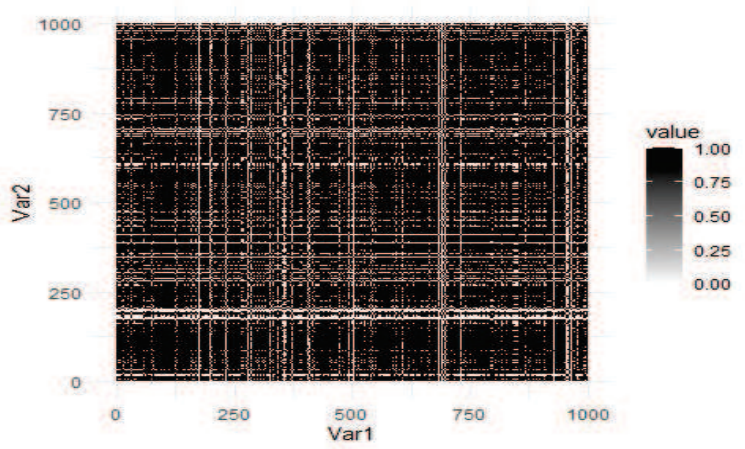

Figura 38:Gráfico de recorrência da Série, aonde se observam claramente um padrão de linhas horizontais e verticais..

Já o da série diferenciada apresenta várias linhas verticais e horizontais, indicando que nesta série, alguns estados mudam de forma lenta por muito tempo, característicos de tendências. Isso posto, faremos os testes de aleatoriedade na série diferenciada a fim de identificarmos se a série é determinística ou aleatória:

```
data:  Binarize_Factorize(na.omit(df_Ouro)$diferenciada)
Standard Normal = 38.185, p-value < 2.2e-16
alternative hypothesis: two.sided
```

O runs test indica que a série não é consistente com o esperado para uma aleatória, provavelmente há um caráter determinístico na mesma. No entanto o preço do ouro parece sofrer grande ocorrência de descontinuidades, sendo, portanto, um campo interessante para análise de changepoints, além do uso de técnicas multivariadas.

5.2.2.2 Volatilidade Ouro

A seguir, faremos a análise de volatilidade do preço do ouro. Nossa análise se iniciará com o gráfico da volatilidade do ouro, iniciando-se no final dos anos 1970. Como será visto, a série aparenta estacionariedade:

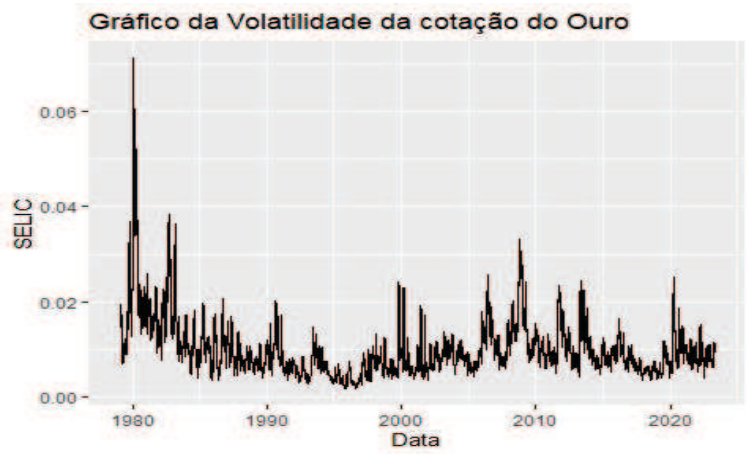

Figura 39:Volatilidade da cotação do ouro.

O gráfico mostra um pico ao final dos anos 1970/início dos 1980, que corresponde aos choques do petróleo na sequência da revolução islâmica na república do Irão. Os picos subsequentes são de intensidade muito menor. Seguiremos com um teste de Dickey-Fuller para estacionariedade

```
##
##  Augmented Dickey-Fuller Test
##
## data:  na.omit(df_Ouro)$volatilidade
## Dickey-Fuller = -6.7354, Lag order = 22, p-value = 0.01
## alternative hypothesis: stationary
```

Observamos que o teste de Dickey-Fuller rejeitou a hipótese nula de raiz unitária, sendo assim, temos uma série estacionária, concordando com a análise visual do gráfico. Seguiremos agora com uma análise de autocorrelação e autocorrelação parcial:

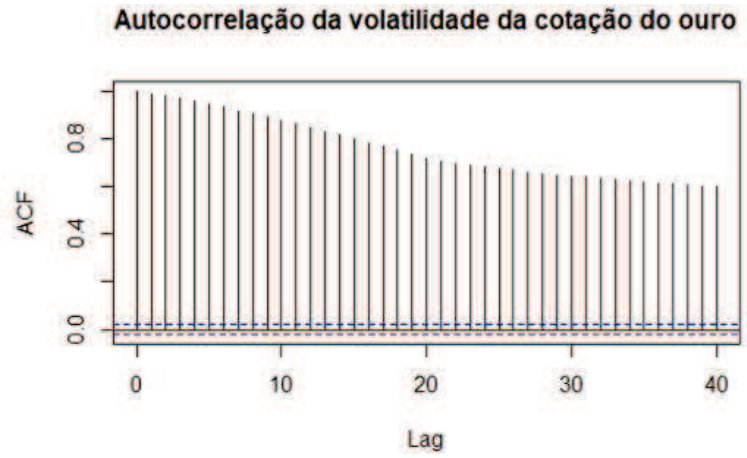

Figura 40:Autocorrelação da volatilidade do Ouro

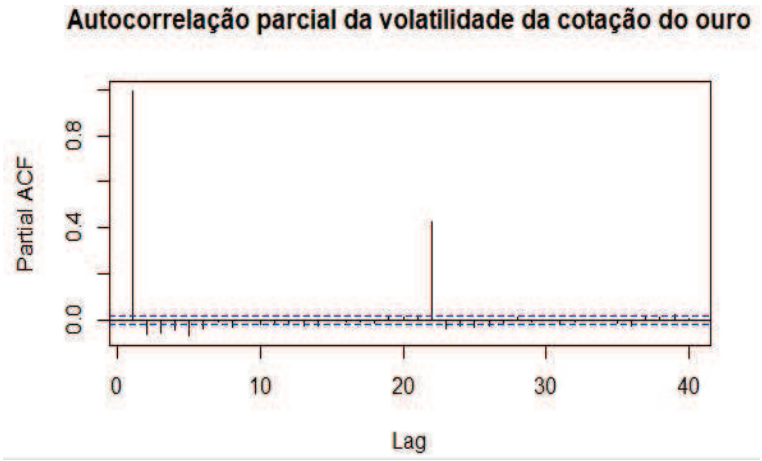

Figura 41:Autocorrelação Parcial da cotação do ouro

O gráfico de autocorrelação indica relevância em todos os lags, com queda suave do primeiro até o vigésimo e estabilização após isso até o 40, o gráfico de autocorrelação parcial, de forma análoga ao que ocorre no da volatilidade da SELIC, indica lags relevantes no primeiro e no 22. Ambas as informações sugerem uma série de autocorrelação igual a 1. Seguiremos agora com um gráfico de recorrência.

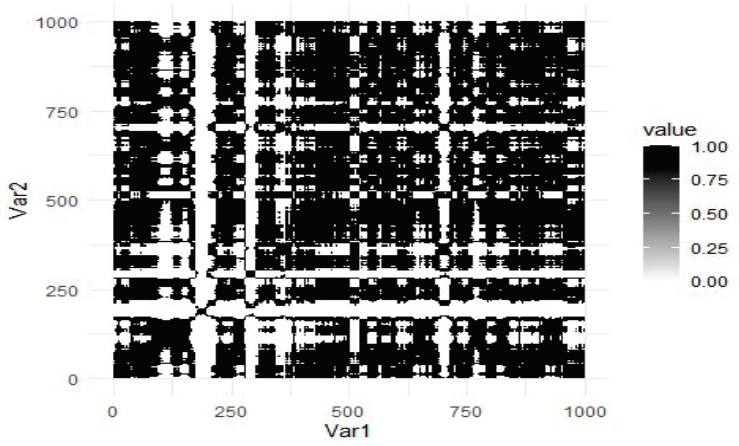

Figura 42: Gráfico de recorrência da volatilidade do ouro

O gráfico de recorrência mostra padrões de linhas horizontais e verticais periódicos, indicando uma possível componente sazonal, faremos um periodograma para avaliar esta possibilidade:

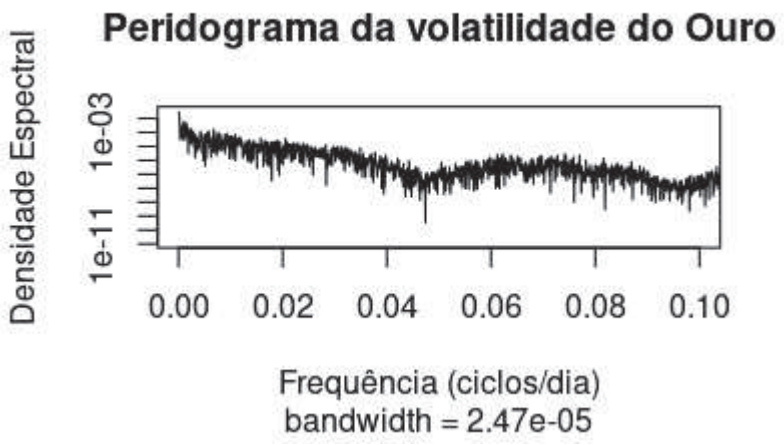

Figura 43: Periodograma da Volatilidade do Ouro.

Novamente, são aparentes aqui ciclos de baixa frequência, exigindo o uso de alguma estratégia para tratamento dos mesmos. Seguiremos agora com um teste de corridas para aleatoriedade:

```
        Runs Test

data:  Binarize_Factorize(na.omit(df_Ouro)$volatilidade)
Standard Normal = -2.031, p-value = 0.04226
alternative hypothesis: two.sided
```

O teste de corridas rejeitou a hipótese de aleatoriedade, observamos então que a série de volatilidade do ouro é uma série potencialmente modelável por meio de algoritmos de séries temporais.

5.2.2.3 Changepoints na variância do Preço do Ouro

Conforme foi mencionado anteriormente, para o ouro, bitcoin e EWZ a SONDE não encontrou quaisquer changepoints. Logo seguiremos com análise apenas utilizando o método PELT-SIC. Apresentamos abaixo os resultados de changepoints na variância da cotação do Ouro em dólares dos Estados Unidos:

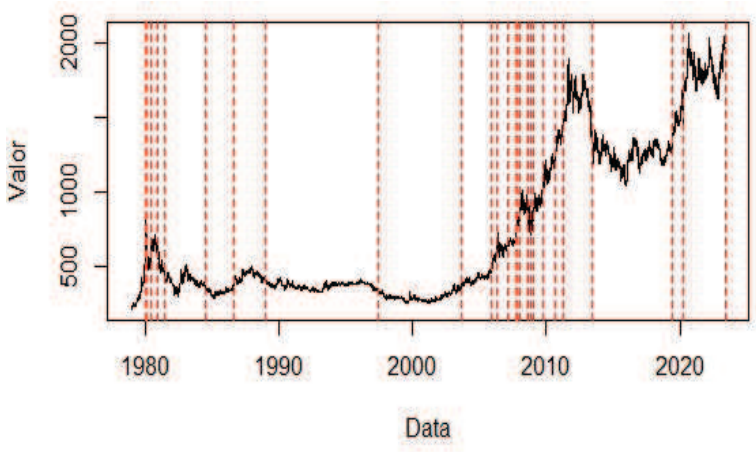

Figura 44: Changepoints na variância da cotação do Ouro calculados por meio do método PELT com penalidade SIC, plotados contra o próprio preço

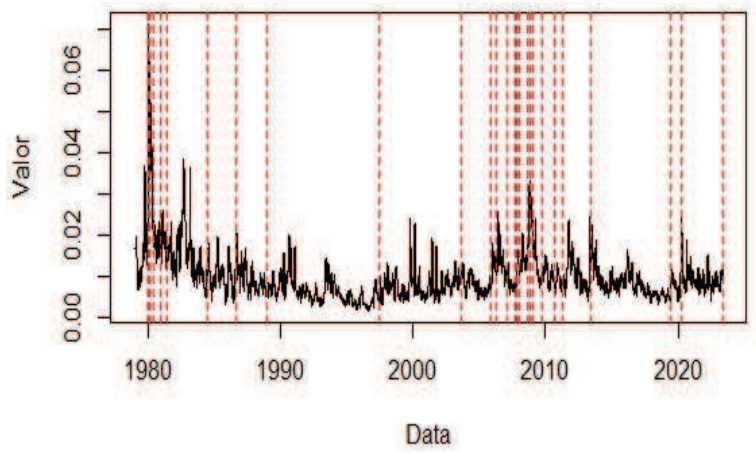

Figura 45: Changepoints na variância da cotação do Ouro calculados por meio do método PELT com penalidade SIC, plotados contra a variância

Observamos que o método PELT-SIC detetou um grande número de changepoints, no início dos anos 1980 (consistentes com os choques do petróleo e crise da américa latina), apenas 1 nos anos 1990 (Correspondente a crise da Ásia) um grande número ao final dos anos 2008 (consistentes com a crise do subprime em 2008), além de um em 2020, consistente com o início da COVID-19.

De uma forma geral, para o Ouro, podemos observar que a maior concentração de changepoints ocorre em períodos de maior instabilidade (mais imprevisíveis) e os períodos calmos se apresentam com ausência de changepoint por anos. Portanto o Método PELT-SIC parece interessante para análise de changepoints neste set.

5.2.2.4 Modelagem Univariada da volatilidade do Preço do Ouro

5.2.2.4.1 Ajuste em Janela Fixa

O Ajuste de um modelo ARIMA à série de volatilidade do ouro resultou em um modelo de ordem (1,1,4). Vamos apresentar abaixo os resultados dos testes estatísticos de Shapiro-Wilk, Ljung-Box e Box-Pierce:

```
##
##  Box-Ljung test
##
## data:  residuos
## X-squared = 39.33, df = 20, p-value = 0.006065

print(lb_test)

##
##  Box-Pierce test
##
## data:  residuos
## X-squared = 39.277, df = 20, p-value = 0.006158

##
##  Shapiro-Wilk normality test
##
## data:  amostra
## W = 0.68765, p-value < 2.2e-16
```

Ao contrário do que houve na taxa Selic, a ordem do modelo de máxima verossimilhança não é de um processo de características aleatórias. Porém, assim, como ocorreu com a taxa Selic, tivemos a rejeição das hipóteses nulas de todos os testes, isto é, os resíduos são não-normais (Shapiro-Wilk) e não-aleatórios (Ljung-Box e Box-Pierce) e, portanto, Heterocedásticos. Desta forma, um modelo ARIMA ajustado a série completa não consegue explicar de forma satisfatória esta série também

5.2.2.4.2 Ajuste em Janela Móvel

A seguir, apresentaremos os resultados da predição da volatilidade do preço do Ouro:

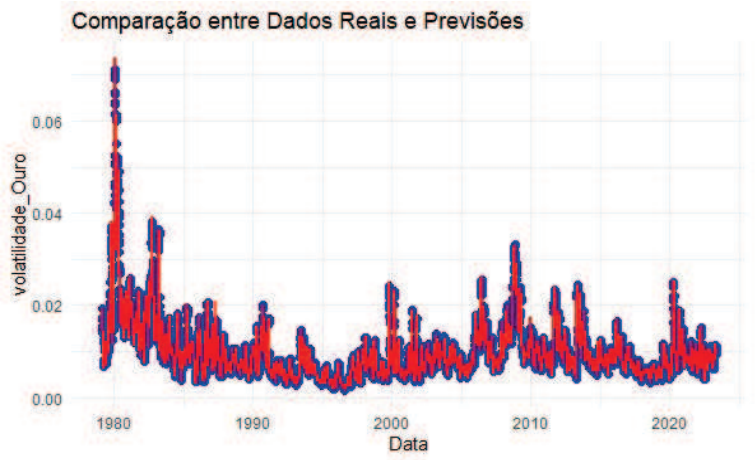

Figura 46:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade do preço do ouro, utilizando um modelo SARIMA

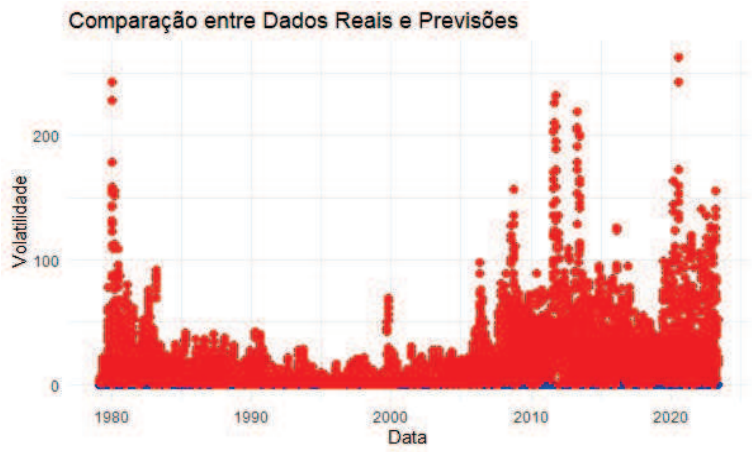

Figura 47:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade do preço do ouro, utilizando um modelo GARCH

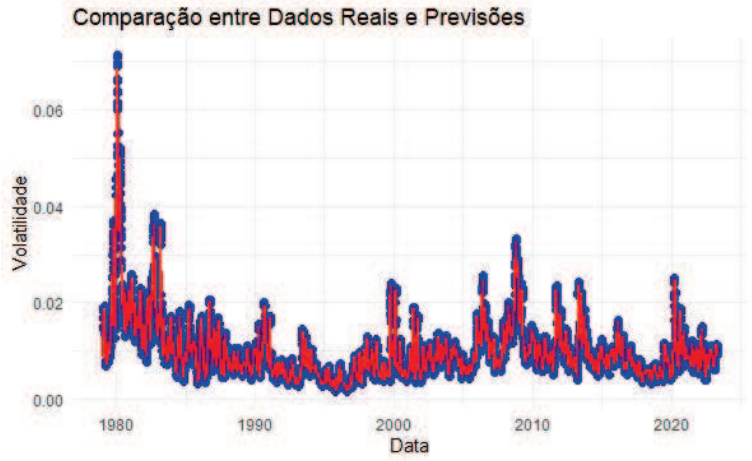

Figura 48:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade do preço do ouro, utilizando um filtro de Kalman

Aqui observamos um R^2 de 0,975 no modelo SARIMA, fortemente negativo no caso do GARCH, o que indica que estimativas usando este modelo são piores que utilizando o valor médio da série (o qual resultaria em zero) e de 0.975 para o filtro de Kalman. Aonde o MAE para o SARIMA e Filtro de Kalman foi de 0,00046 e 0,00056. A abordagem SARIMA rolante teve desempenho ligeiramente melhor do que o Filtro de Kalman. Aparentemente, não temos outliers nesta predição.

5.2.2.4.3 Análise com EMD

Na análise de volatilidade do preço do ouro obtivemos 12 IMF's, mais a residual, a somatória das previsões ajustadas para cada modelo SARIMA rolante resultou no seguinte gráfico:

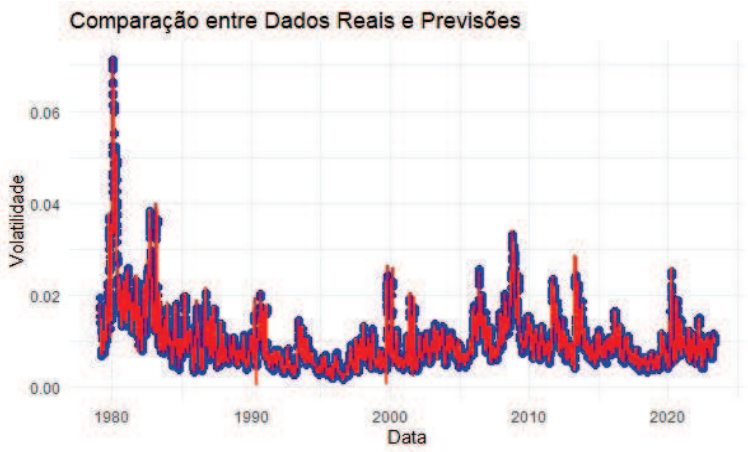

Figura 49:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade do preço do ouro , utilizando uma combinação de modelos SARIMA pra cada IMF

Aqui obtivemos um R^2 de 0,9847 e um MAE de 0,00042, ambos ligeiramente melhores do que o observado na análise univariada. Para o caso do Ouro, observamos que o uso de EMD leva a performances melhores nas duas métricas

3.2.3 Cotação do Fundo negociado em Bolsa EWZ

O EWZ é um índice de ações contendo apenas empresas negociadas na bolsa de valores de São Paulo (B3) e representa grosso modo uma exposição ao setor produtivo da economia brasileira. Além do índice existe também um fundo negociado em bolsa (ETF em inglês), listado na bolsa de Nova York que detém uma carteira de ações igual ao índice, fornecendo aos investidores uma forma de ter exposição ao mesmo. A seguir, exibiremos um gráfico e um histograma contendo a cotação de fechamento deste fundo, no período de 2000 até 2023. (IShares, 2023)

Figura 50:Série Histórica do EWZ, aonde observamos uma trajetória ascendente até a crise de 2008 e depois uma tendência descendente

O gráfico de preço mostra dois momentos bastante distintos: um período de alta desde 2000 até aproximadamente 2008 (período do “Boom das commodities”) queda abrupta e recuperação parcial entre 2008 e 2010 (crise do subprime em 2008) e

um segundo período de queda. Visualmente, é possível identificar que a série não é estacionária. A Seguir exibiremos um histograma dos preços:

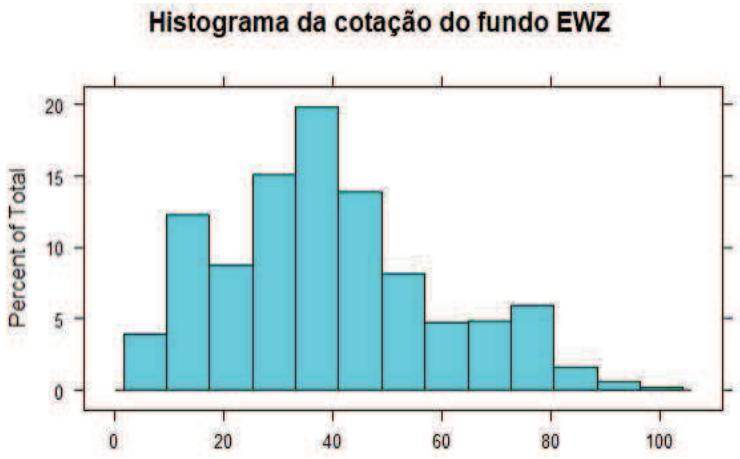

Figura 51:Histograma da cotação do fundo EWZ

O histograma revela uma cauda longa a direita, aonde temos o pico dos preços por volta dos USD 100, mas na maior parte do tempo o preço permaneceu entre USD 20 e USD 40. Seguiremos agora com uma análise da série diferenciada:

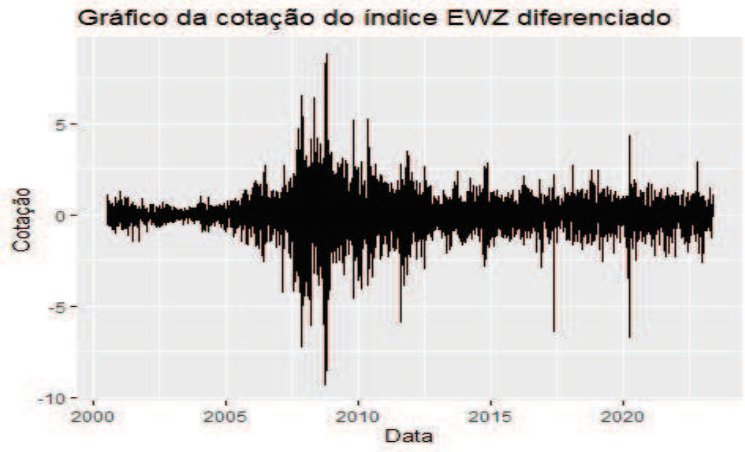

Figura 52:EWZ Diferenciado, aonde é possível observar um momento de grande variação na crise de 2008

O gráfico diferenciado revela um momento de maior variação do índice no período da crise do subprime, um período de menor variação na primeira década do século XXI e, após 2010, um terceiro período de variação aproximadamente constante. Visualmente a série aparenta ser estacionária. Para confirmarmos, Seguiremos com um teste de Dickey-Fuller:

```
##
##  Augmented Dickey-Fuller Test
##
## data:  na.omit(df_EWZ)$Close
## Dickey-Fuller = -1.962, Lag order = 17, p-value = 0.5944
## alternative hypothesis: stationary
```

```
## Resultado do teste de estacionariedade - Série diferenciada(ADF):

print(adf_result)

##
##  Augmented Dickey-Fuller Test
##
## data:  na.omit(df_EWZ)$diferenciada
## Dickey-Fuller = -17.421, Lag order = 17, p-value = 0.01
## alternative hypothesis: stationary
```

O teste de Dick-Fulley indica que a série original é não estacionária, mas que a série diferenciada é estacionária. Vamos analisar a auto correlação destas duas séries:

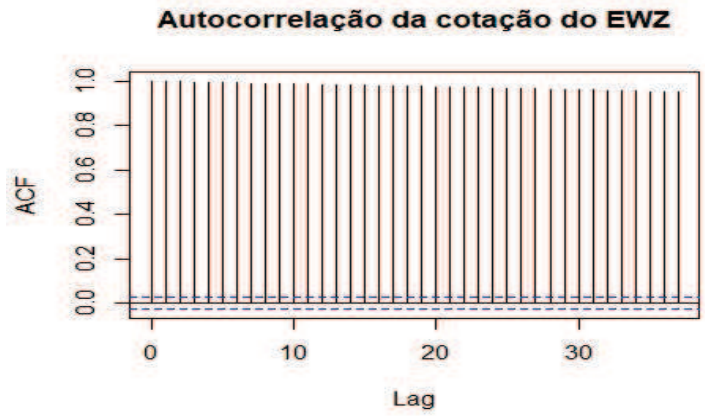

Figura 53:Autocorrelação da série original, aonde podemos observar um padrão quase constante de relevância dos lags

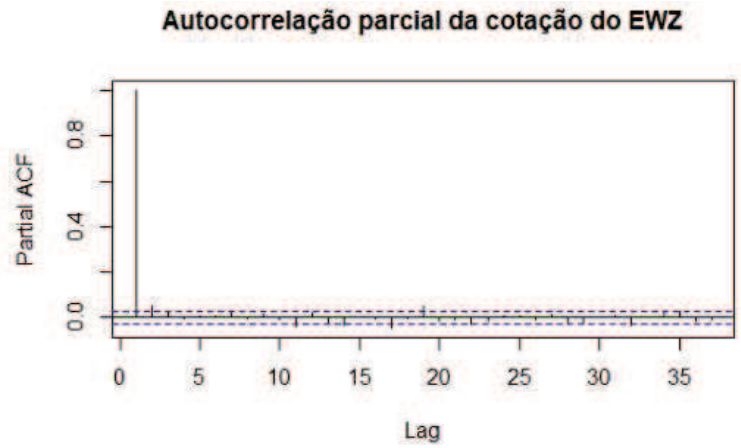

Figura 54:Autocorrelação parcial do preço do Ouro

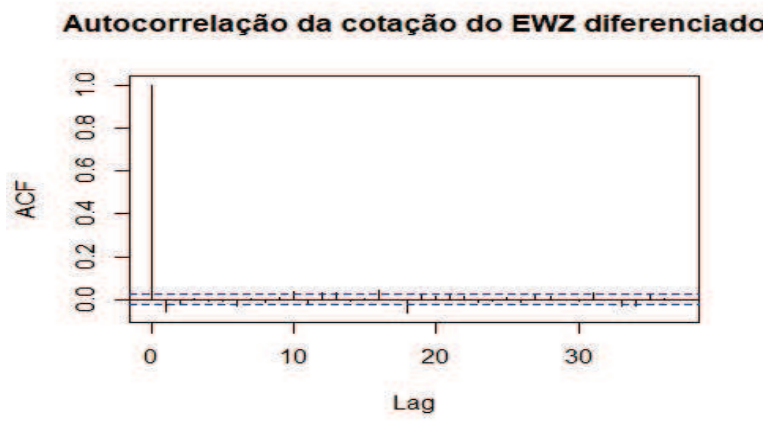

Figura 55:Autocorrelação da série diferenciada, aonde não se observa autocorrelação

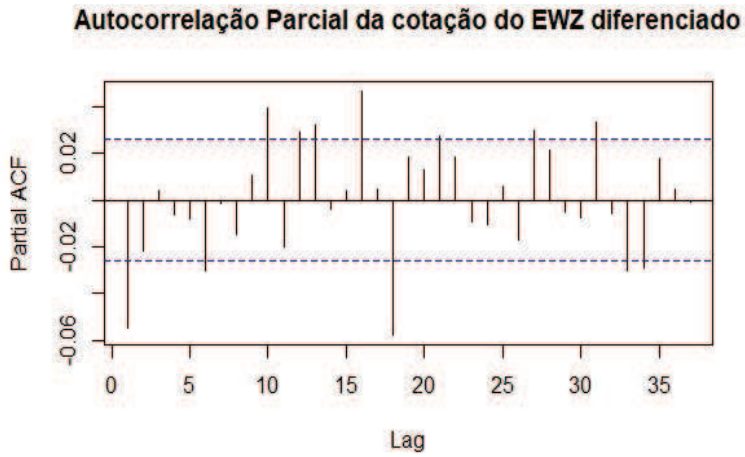

Figura 56:Autocorrelação parcial da cotação do EWZ diferenciada

Observamos um padrão similar ao reportado para o preço do ouro: relevância de todos os lags mostrados no gráfico de autocorrelação da série, o que não ocorre no gráfico de autocorrelação da série diferenciada, juntamente com os gráficos de autocorrelação parcial, relevantes apenas para o lag 1 na série original e não relevantes para a série diferenciada, indicam uma série sem autocorrelação. Seguiremos com uma análise de recorrência a fim de identificar o caráter da série:

Recurrence plot do preço do EWZ

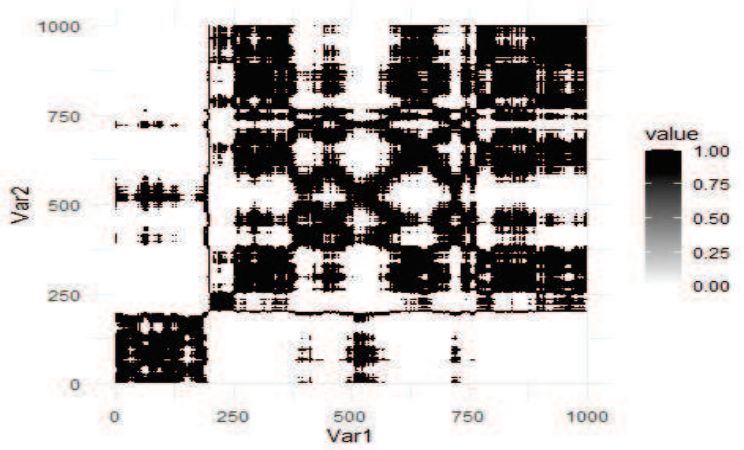

Figura 57:Gráfico de recorrência do EWZ, observamos um cluster na centro-direita com descontinuidades

Temos um gráfico de recorrência para o EWZ bastante similar ao observado para o ouro: O gráfico da série mostra um cluster grande na centro-direita superior com algumas faixas brancas, indicando descontinuidades ou ocorrências de transições, A seguir apresentamos o recurrence plot do preço do EWZ diferenciado

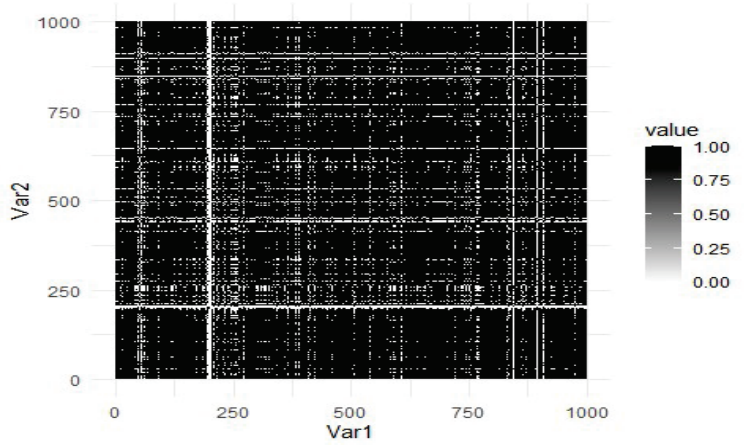

Figura 58:Gráfico de recorrência da série diferenciada do EWZ, aonde observamos um padrão de linhas verticais e horizontais

já o da série diferenciada apresenta uma linha vertical e outra horizontal branca claramente destacada, indicando que existe um estado único que não se repetiu, isso indica a possibilidade de ocorrência de um changepoint.

Dado estes pontos, faremos os testes de aleatoriedade a fim de identificarmos se a série é determinística ou aleatória:

```
        Runs Test

data:  Binarize_Factorize(na.omit(df_EWZ)$diferenciada)
Standard Normal = 24.499, p-value < 2.2e-16
alternative hypothesis: two.sided
```

Observamos que a série não é consistente com uma série aleatória, existe a presença de um estado único, provavelmente causado por um changepoint, razão pela qual pode ser interessante investigarmos o mesmo. Além disso, o gráfico em função do tempo revela padrões na série que possivelmente possam ser modelados em futuras análises multivariadas.

3.2.3.1 Volatilidade EWZ

A seguir faremos uma análise da volatilidade do índice EWZ. No gráfico abaixo, podemos observar dois picos significativos na crise de 2008 e no período inicial da pandemia de COVID 19:

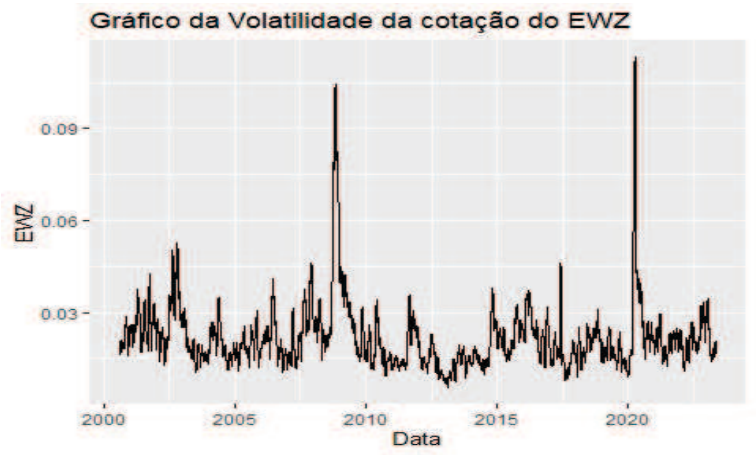

Figura 59:Volatilidade da cotação do EWZ

Como aconteceu com as séries anteriores, a série aparenta estacionariedade. Na sequência aplicaremos o teste de Dickey-Fuller:

```
## Resultado do teste de estacionariedade - Série original(ADF):

print(adf_result)

##
##  Augmented Dickey-Fuller Test
##
## data:  na.omit(df_EWZ)$volatilidade
## Dickey-Fuller = -8.9705, Lag order = 17, p-value = 0.01
## alternative hypothesis: stationary
```

O teste de Dickey-Fuller confirmou nossa análise visual de que a série é estacionária. Seguiremos com uma análise de autocorrelação:

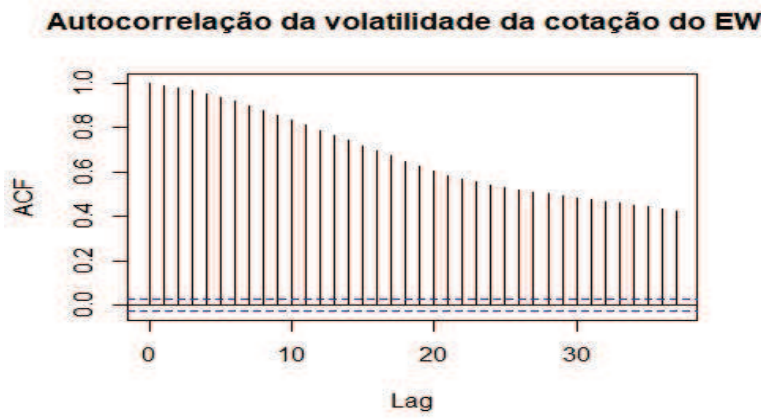

Figura 60:Autocorrelação da cotação do EWZ

De forma análoga as séries anteriores, vemos autocorrelações significativas, mas apenas o primeiro e vigésimo segundo lags relevantes na autocorrelação parcial. Sugerindo uma série autorregressiva.

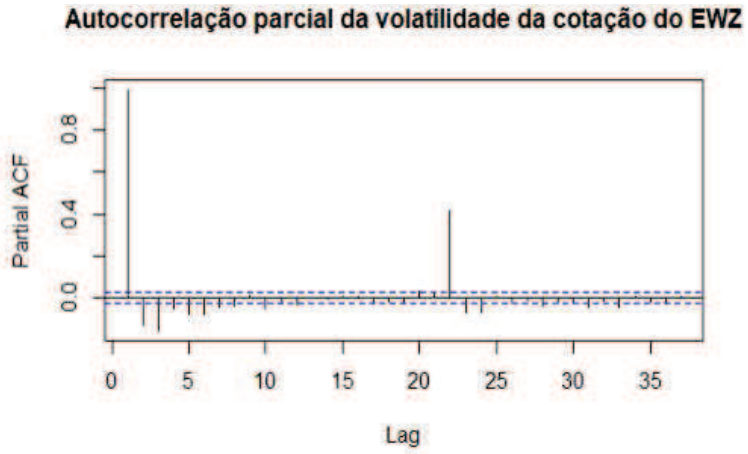

Figura 61:Autocorrelação Parcial da Cotação do EWZ

De forma análoga as séries anteriores, vemos autocorrelações significativas, mas apenas o primeiro e vigésimo segundo lags relevantes na autocorrelação parcial. Sugerindo uma série autorregressiva de ordem 1, seguiremos com a análise do gráfico de recorrência.

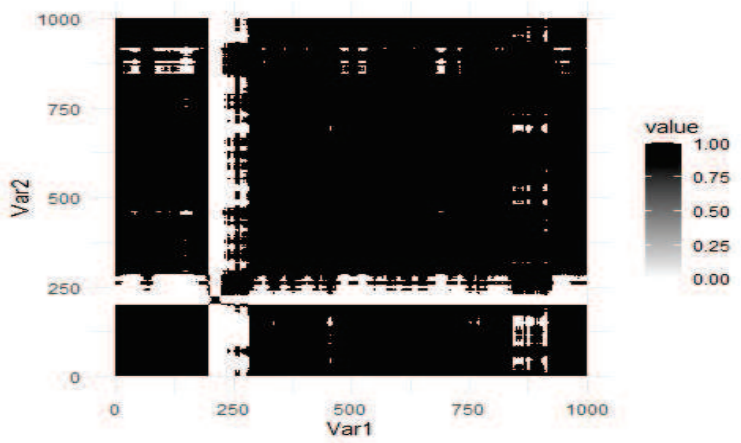

Figura 62:Gráfico de recorrência da cotação do EWZ

O gráfico de recorrência mostra uma linha horizontal e outra vertical, indicativo de estados sem repetição (correspondentes aos dos picos nas crises de 2008 e na de 2020, após o covid-19) ou disrupção, não há indicativo de periodicidades/ciclos. Seguiremos com os testes de aleatoriedade

```
Runs Test

data:  Binarize_Factorize(na.omit(df_EWZ)$volatilidade)
Standard Normal = -0.60791, p-value = 0.5432
alternative hypothesis: two.sided
```

O Runs test não rejeitou a hipótese nula de aleatoriedade da série de volatilidade do EWZ, isto indica que provavelmente teremos dificuldade para modelar esta série com algoritmos de séries temporais. Porém a presença de estados sem repetição indica algum tipo de interrupção, a qual será melhor avaliada nas análises de changepoints.

3.2.3.2 Changepoints na variância do preço do fundo EWZ

Na sequência, mostraremos os changepoints observados no preço do fundo de índice EWZ utilizando o método PELT-SIC:

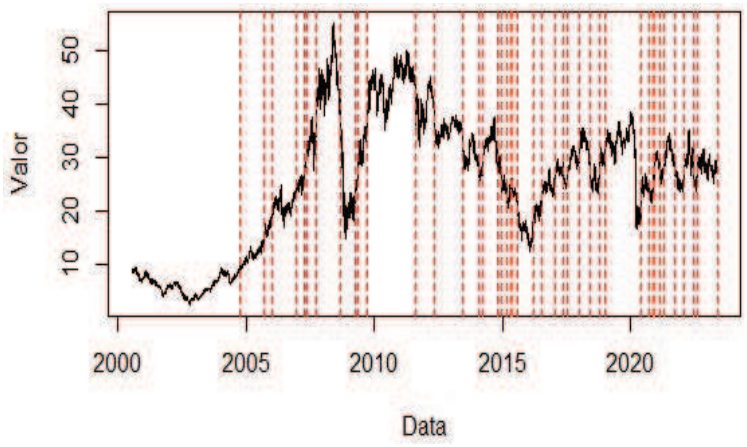

Figura 63:Changepoints na variância da cotação do preço do EWZ calculados por meio do método PELT com penalidade SIC, plotados contra o próprio preço

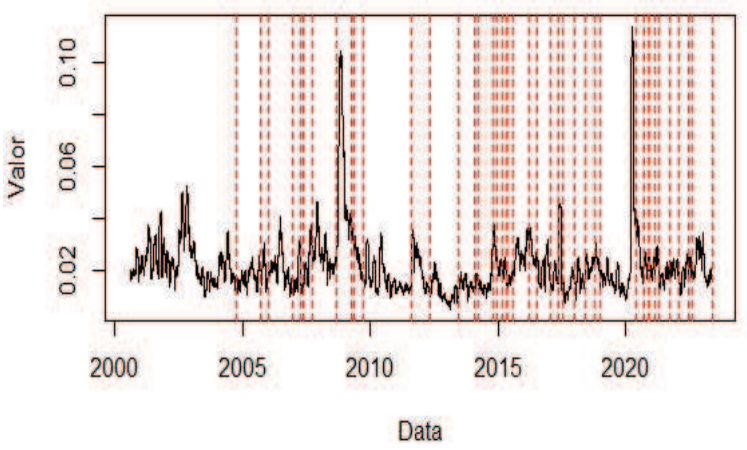

Figura 64:Changepoints na variância da cotação preço do EWZ, calculados por meio do método PELT com penalidade SIC, plotados contra a variância

A análise de changepoints por PELT-SIC indicou uma quantidade grande de changepoints dispersa por toda a série, com maior concentração nas proximidades de 2015 (impeachment da então presidente do Brasil) e 2020 (COVID-19), existem dois grandes picos de volatilidade em 2020 e 2008, que provavelmente geram distorções nesta análise. Desta forma, a análise de changepoints por PELT-SIC não parece retornar resultados satisfatórios para essa série.

3.2.3.3 Modelagem Univariada da volatilidade da cotação do EWZ

3.2.3.3.1 Ajuste em Janela Fixa

O modelo ARIMA de máxima verossimilhança ajustado a série do BTC possui ordem (2,1,2), seguiremos com a avaliação dos resultados dos testes de Ljung-Box e Box-Pierce e Shapiro-Wilk:

```
##
##  Box-Ljung test
##
## data:  residuos
## X-squared = 52.447, df = 20, p-value = 9.796e-05
```

```
print(lb_test)

##
##  Box-Pierce test
##
## data:  residuos
## X-squared = 52.323, df = 20, p-value = 0.0001021

##
##  Shapiro-Wilk normality test
##
## data:  amostra
## W = 0.74978, p-value < 2.2e-16
```

A Análise da série de resíduos indicam resultados similares aos obtidos no ouro: resíduos não aleatórios e não normais, desta forma, não há homocedasticidade, não sendo adequada uma modelagem ARIMA ajustada a série toda.

5.2.3.4.2 Ajuste em Janela Móvel

Por fim, apresentaremos os resultados observados na predição dos preços do fundo de índice EWZ:

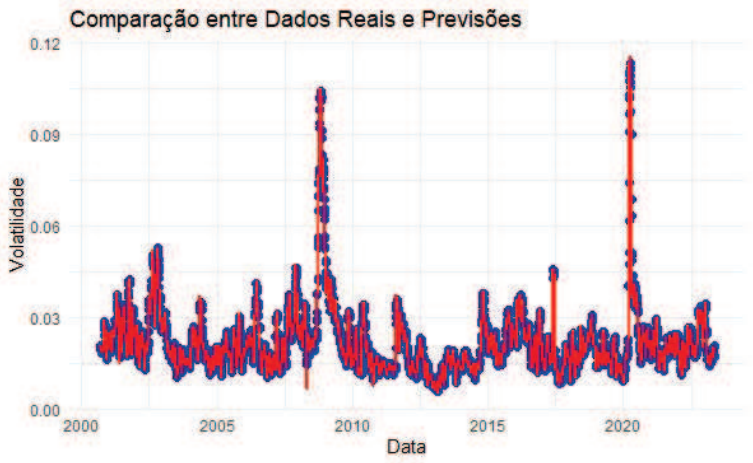

Figura 65:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade do preço do EWZ, utilizando um modelo SARIMA

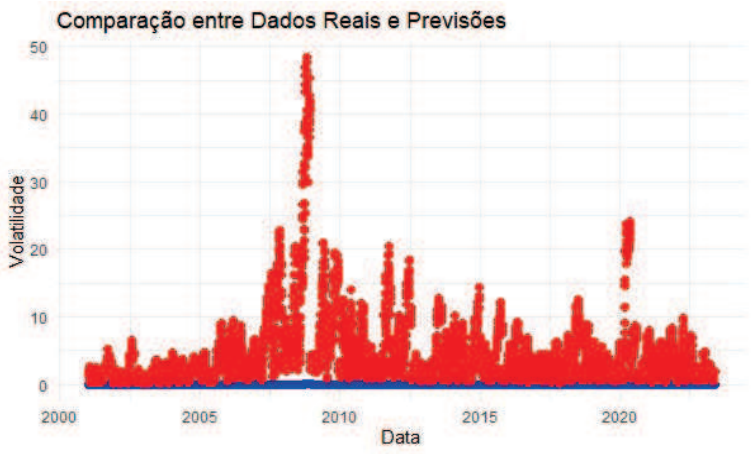

Figura 66:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade do preço do EWZ, utilizando um modelo GARCH

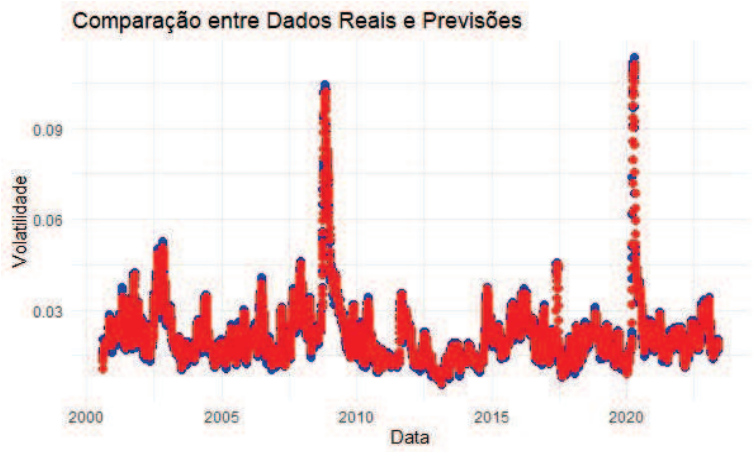

Figura 67:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade do preço do EWZ, utilizando um Filtro de Kalman

Os resultados foram de um R Quadrado de 0,97 para o SARIMA, Negativo para o GARCH e de 0.969 para o Filtro de Kalman. MAE observado foi de 0,000939 no SARIMA e 0,0011 no filtro de Kalman . A análise exploratória da Série de volatilidades do EWZ identificou uma série estacionária, não aleatória e provavelmente não linear. Com dois picos, próximos as crises de 2008 e ao início da COVID-19 que distorceram os valores. No entanto, o SARIMA foi capaz de capturar estes fatos de forma similar ao Filtro de Kalman.

5.2.3.4.3 Análise com EMD

Por fim, na análise do preço do fundo de índice EWZ, obtivemos 11 IMF's, mais a residual, o gráfico das predições x real está abaixo:

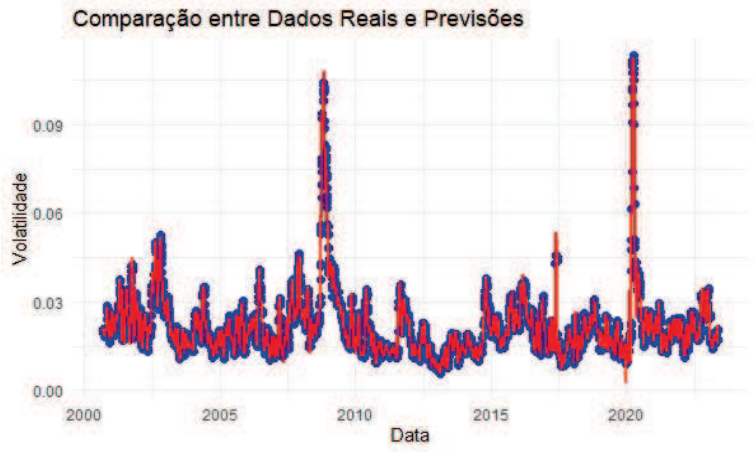

Figura 68:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade do preço do EWZ, utilizando uma combinação de modelos SARIMA pra cada IMF

Obtivemos um valor de R^2 de 0,9818 e de 0,00080 para o MAE, mantendo o mesmo padrão observado nas séries do Ouro e Bitcoin, aonde o uso de EMD resultou em um ligeiro aumento de performance.

3.2.4 Preço da Criptomoeda Bitcoin

Pouco se sabe sobre o criador(a) do Bitcoin, exceto que atendia pelo nome de Satoshi Nakamoto, embora exista uma pessoa com este nome, já apontada como

possível criador, ela nega a autoria e não se sabe se este é o nome real, um pseudônimo, ou se trata-se de um grupo de pessoas. *(Rohr, 2014)*

Bitcoin consiste em um protocolo de pagamentos eletrônico *Peer-to-Peer* (Pessoa-para-Pessoa), isto é, permite que transações financeiras sejam liquidadas de forma puramente eletrônica, sem a necessidade de um intermediador financeiro utilizando uma tecnologia denominada *Blockchain* (sequência de blocos). Este sistema possui uma unidade de conta, com o mesmo nome do protocolo e que pode ser cotada em moedas tradicionais em ambientes de negociação específicos. *(Nakamoto, 2008)*

A seguir apresentaremos um gráfico de dispersão contendo a cotação do bitcoin, de 2014 até 2023, expressa em dólares dos estados unidos.

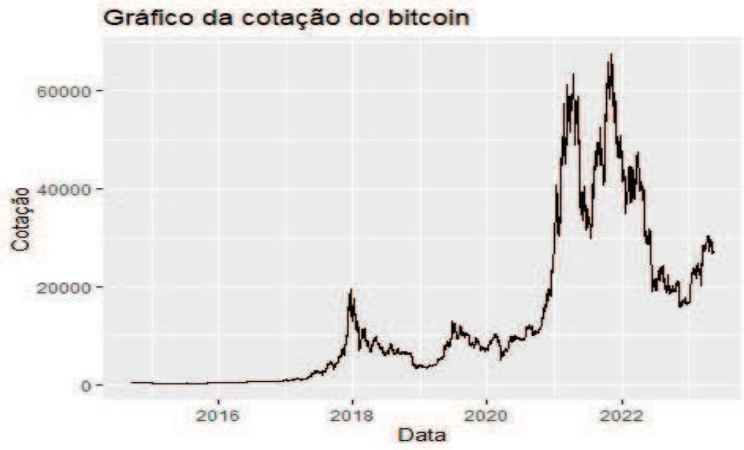

Figura 69:Cotação do Bitcoin em função do tempo, podemos observar uma tendência de alta até 2022, quando houve uma queda significativa.

O gráfico do Bitcoin revela alguns momentos distintos, primeiro desde o início até 2017 quando o preço era estável e próximo a zero. Houve um pico em 2018, com estabilização em novo patamar de 2018 até o final de 2020.Entre 2021 e 2022 a cotação atingiu novas máximas recordes, posteriormente caindo a um novo patamar médio. A seguir, visualizaremos a distribuição dos preços no histograma:

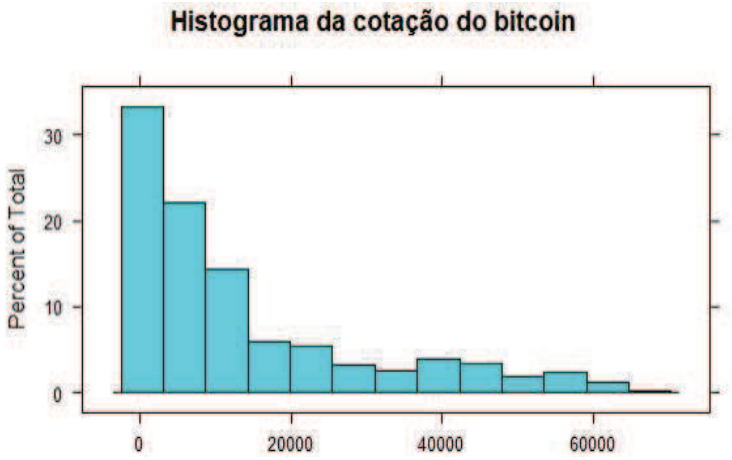

Figura 70:Histograma da cotação do Bitcoin

O Histograma nos revela a cauda mais longa de todas as séries que analisamos. Consistente com os picos de preços ocorridos no passado recente. A série, como as do Ouro e EWZ, aparenta não-estacionariedade, faremos a primeira diferenciação e exibiremos os resultados abaixo:

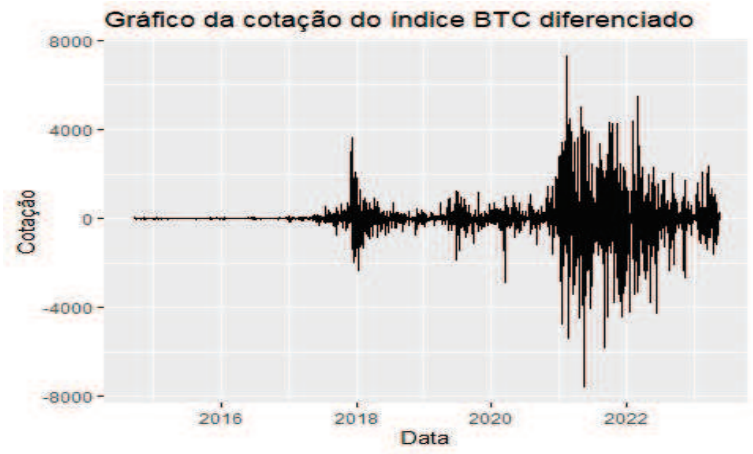

Figura 71:Gráfico do preço do Bitcoin diferenciado

O gráfico do BTCUSD diferenciado indica as maiores variações ao redor do pico de 2018 e na sequência nas novas máximas de 2021-22. Na sequência, em 2023 os preços parecem estabilizar. O que indica uma possível estacionariedade, seguiremos com o Teste de Dickey-Fuller na mesma:

```
## Resultado do teste de estacionariedade - Série diferenciada(ADF):

print(adf_result)

##
##  Augmented Dickey-Fuller Test
##
## data:  na.omit(df_BTC)$diferenciada
## Dickey-Fuller = -14.474, Lag order = 14, p-value = 0.01
## alternative hypothesis: stationary
```

O Teste de Dickey-Fuller rejeitou a hipótese nula, indicando que a série é estacionária, faremos agora uma análise de correlação e autocorrelação da série original e da diferenciada:

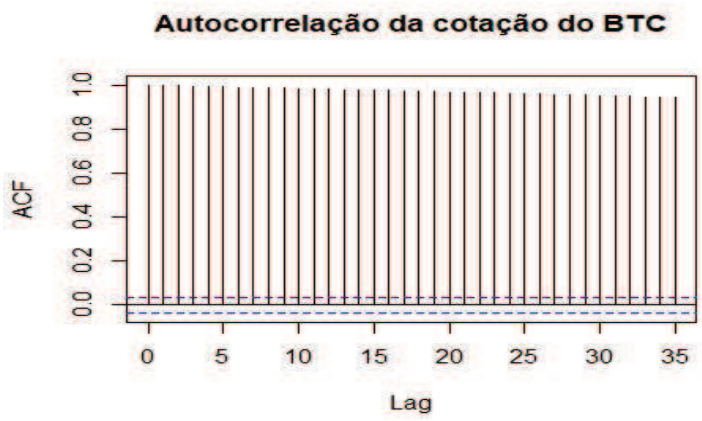

Figura 72:Autocorrelação da cotação do Bitcoin

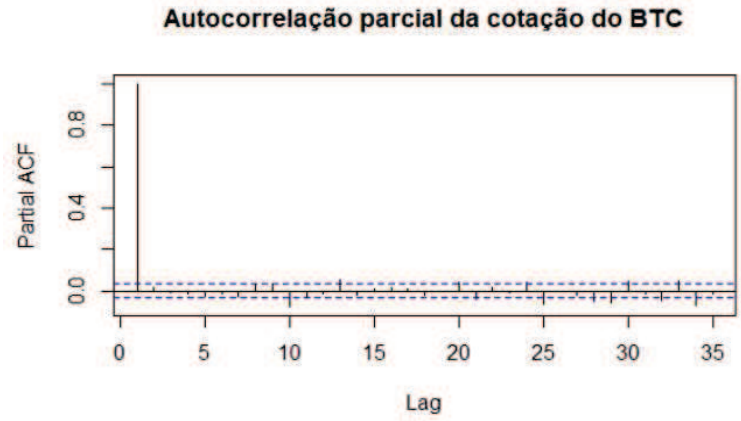

Figura 73:Autocorrelação parcial da cotação do BTC

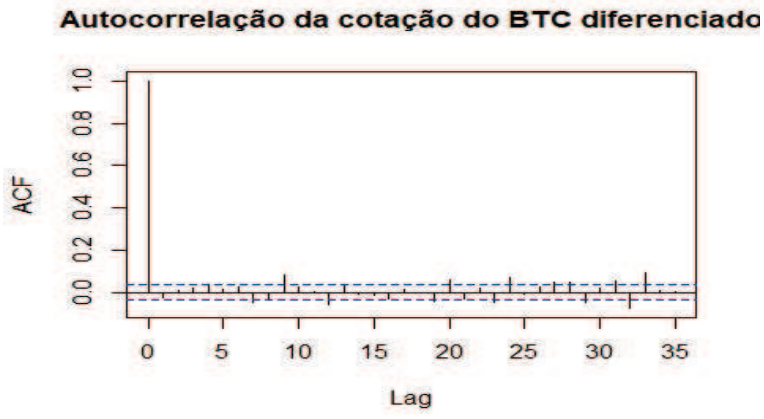

Figura 74:Autocorrelação da cotação do Bitcoin diferenciada

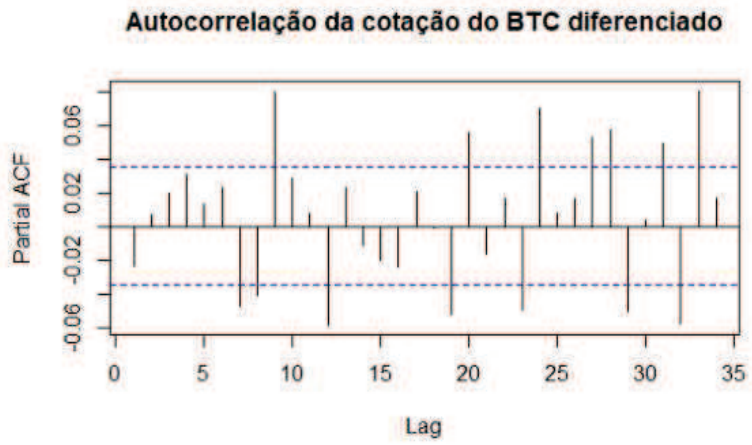

Figura 75:Autocorrelação Parcial do preço do BTC diferenciado

Os gráficos de Autocorrelação e Autocorrelação Parcial das séries original e diferenciada se mostram bastante similares aos das séries do Ouro e EWZ: aparentemente não há autocorrelação significativa na série diferenciada. Seguiremos agora com uma análise do recurrence plot das mesmas:

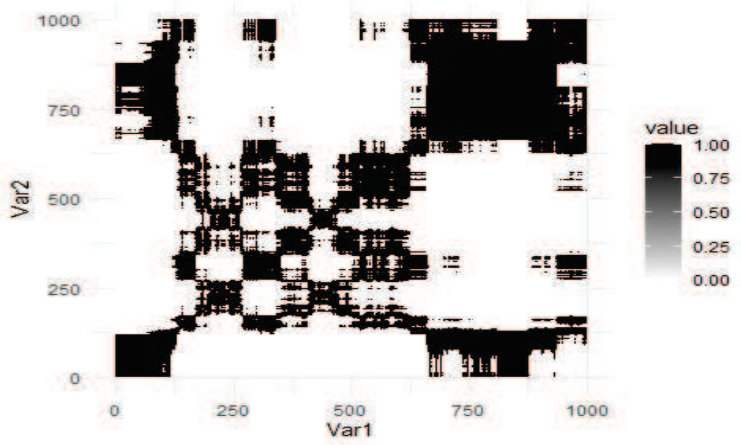

Figura 76:Gráfico de recorrência do preço do Bitcoin

O gráfico da série mostra linhas negras diagonais sugerindo estados que mudam de forma similar. Seguiremos com um recurrence plot do BTC diferenciado

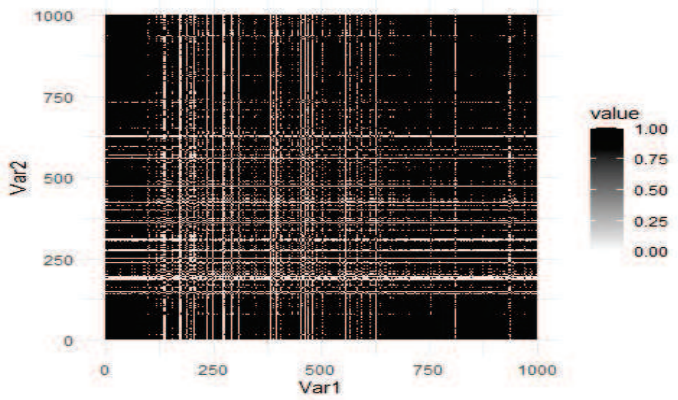

Figura 77:Autocorrelação do preço do Bitcoin diferenciado, aonde se observa o padrão de linhas horizontais e verticais

O Gráfico de recorrência da série diferenciada apresenta linhas vertical e horizontais em clusters, indicando que existem estados que mudam lentamente. Seguiremos com um runs test para avaliar se a série é ou não aleatória.

```
Runs Test

data:  Binarize_Factorize(na.omit(df_BTC)$diferenciada)
Standard Normal = 19.937, p-value < 2.2e-16
alternative hypothesis: two.sided
```

Aparentemente não temos uma série consistente com uma totalmente aleatória. Os gráficos de recorrência para o BTC, por sua vez, adicionam uma informação relevante: O Fato de a série original possuir estados que mudam de forma similar. Enquanto o da série diferenciada apresenta estados que mudam lentamente, levanta a possibilidade de que a série do Bitcoin seja governada por etapas relativamente homogêneas, mas com transições abruptas entre elas, sugerindo a possibilidade de que abordagens locais possam obter bons resultados

5.2.4.2 Volatilidade BTC

A seguir, analisaremos a volatilidade da série do Bitcoin os dados se iniciam em 2015, para excluir os valores mais baixos de uma época em que a moeda não era conhecida e possuía pouca liquidez:

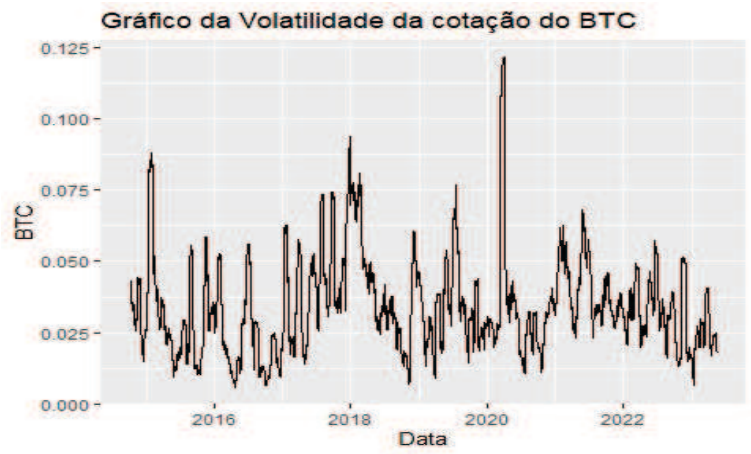

Figura 78:Série histórica da volatilidade do Bitcoin no período posterior a 2015

A volatilidade do BTC aparenta estacionariedade, seguiremos com um teste aumentado de dickey-fuller para confirmar esta hipótese:

```
## Resultado do teste de estacionariedade - Série original(ADF):

print(adf_result)

##
##  Augmented Dickey-Fuller Test
##
## data:  na.omit(df_BTC)$volatilidade
## Dickey-Fuller = -8.5236, Lag order = 14, p-value = 0.01
## alternative hypothesis: stationary
```

O Teste aumentado de Dickey-Fuller rejeitou a hipótese de raiz unitária. Confirmando a estacionariedade. Seguiremos agora com uma análise dos gráficos de autocorrelação e autocorrelação parcial:

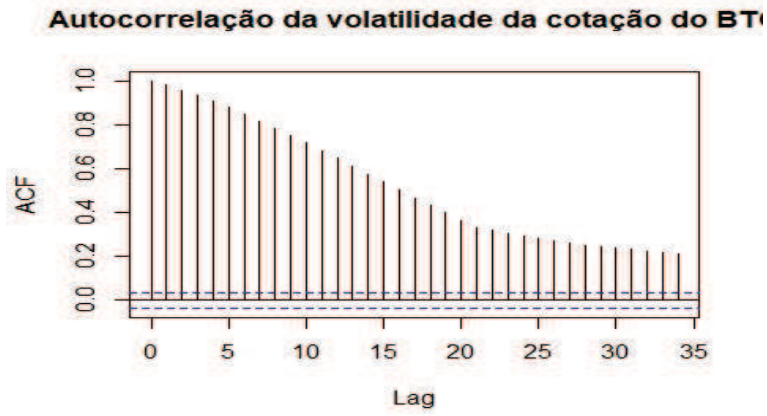

Figura 79:Autocorrelação da volatilidade do Bitcoin

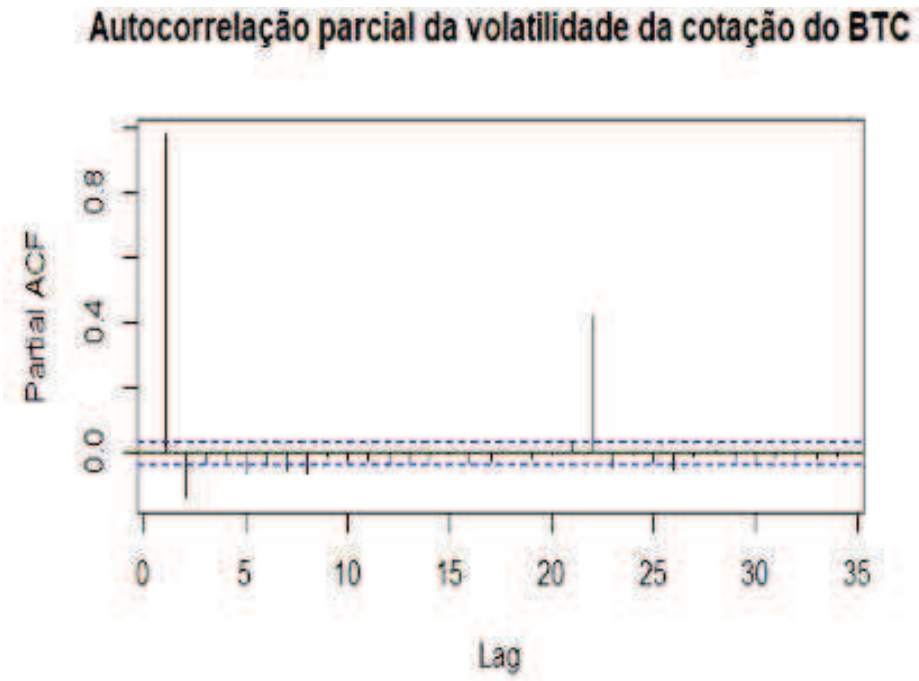

Figura 80:Autocorrelação parcial da volatilidade do BTC

Os gráficos de autocorrelação e autocorrelação parcial, continuam semelhantes aos das demais séries, sugerindo uma série autorregressiva de ordem 1.Vamos avaliar agora o gráfico de recorrência:

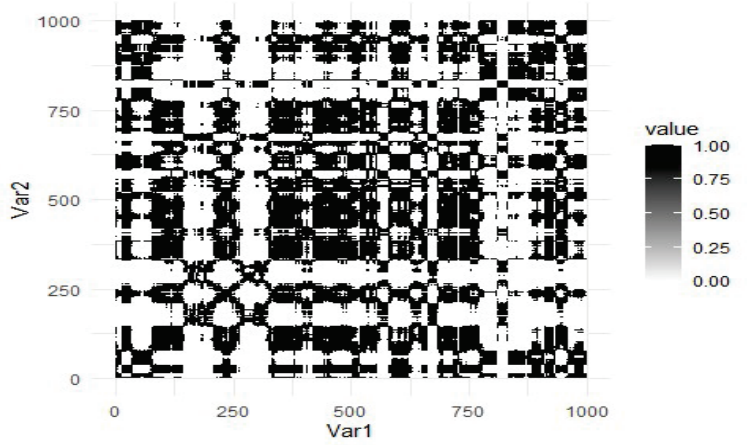

Figura 81:Gráfico de recorrência da volatilidade do Bitcoin

O gráfico de recorrência, aparece ser bastante homogêneo, de uma forma geral consistente com um processo estacionário. Seguiremos com um teste de aleatoriedade

```
Runs Test

data:  Binarize_Factorize(na.omit(df_BTC)$volatilidade)
Standard Normal = -3.2407, p-value = 0.001192
alternative hypothesis: two.sided
```

O runs test indica que a série não é aleatória. Isto, junto com o observado nas análises anteriores indicam tratar-se de uma série potencialmente modelável com algoritmos de séries temporais.

5.2.4.3 Changepoints na variância do preço do Bitcoin

Por fim, apresentaremos os changepoints no preço do Bitcoin:

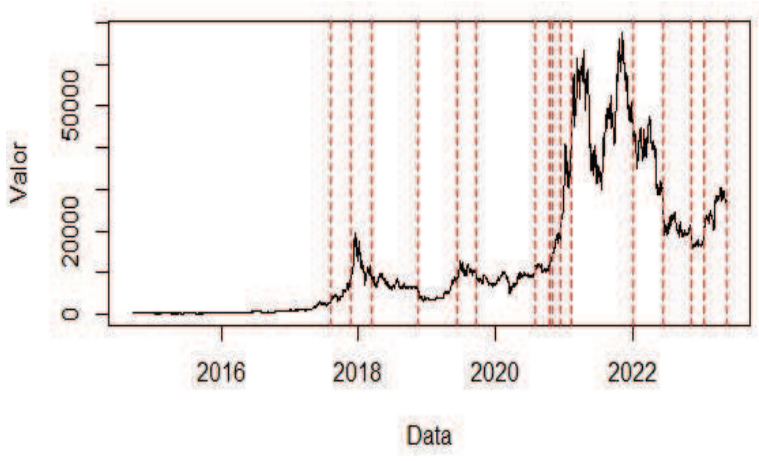

Figura 82: Changepoints na variância da cotação do preço do Bitcoin, calculados por meio do método PELT com penalidade SIC, plotados contra o próprio preço

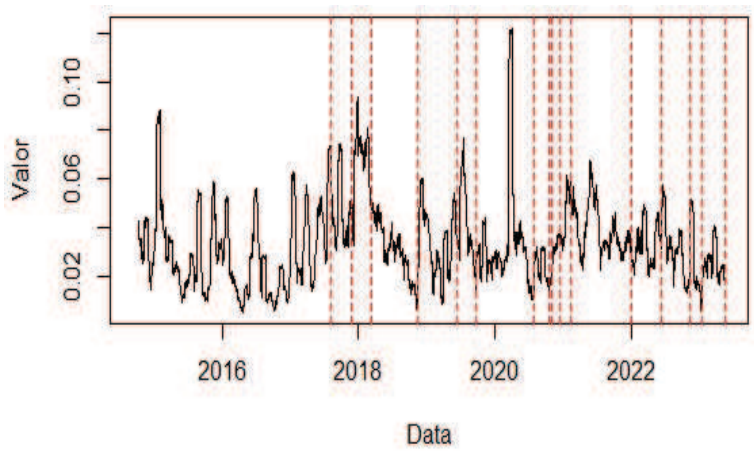

Figura 83:Changepoints na variância da cotação preço do Bitcoin, calculados por meio do método PELT com penalidade SIC, plotados contra a variância

O método PELT com penalidade SIC identificou changepoints na série de preços do Bitcoin ao redor de 2018 (que foi um período considerado de alta volatilidade, com preços atingindo um pico na época e depois caindo de forma abrupta), bem como na proximidade de 2020 (pandemia do COvid-19) e fora isso changepoints esporádicos próximos dos principais picos de volatilidade entre 2018-2020 e no ano de 2022. De uma forma geral, a deteção de changepoints aparenta boa consistência para esta série.

5.2.4.4 Modelagem Univariada do Preço do Bitcoin

5.2.4.4.1 Ajuste em Janela Fixa

Passamos agora à série de volatilidade do Bitcoin. Os parâmetros do modelo ARIMA de máxima verossimilhança resultarem em (2,0,2). Vamos agora observar os resultados dos testes de Ljung-Box e Box-Pierce e Shapiro-Wilk:

```
##
##  Box-Ljung test
##
## data:  residuos
## X-squared = 24.811, df = 20, p-value = 0.2087

print(lb_test)

##
##  Box-Pierce test
##
## data:  residuos
## X-squared = 24.71, df = 20, p-value = 0.2127

##
##  Shapiro-Wilk normality test
##
## data:  amostra
## W = 0.60632, p-value < 2.2e-16
```

Os testes de box-ljung e box-Pierce não rejeitaram a hipótese nula de nenhuma autocorrelação. O que indica que esta série, ao contrário da do Ouro e SELIC, se aproximou da homocedasticidade de resíduos. No entanto o teste de Shapiro-Wilk rejeitou a hipótese nula de normalidade dos resíduos. Analisando apenas estes dados, isto nos leva a conclusão de que, potencialmente a volatilidade do bitcoin possa ser

modelada por um único modelo ajustado a toda a série, mas provavelmente este modelo será não-linear.

5.2.4.4.2 Ajuste em Janela Móvel

A seguir, apresentamos os resultados na análise de volatilidade do bitcoin:

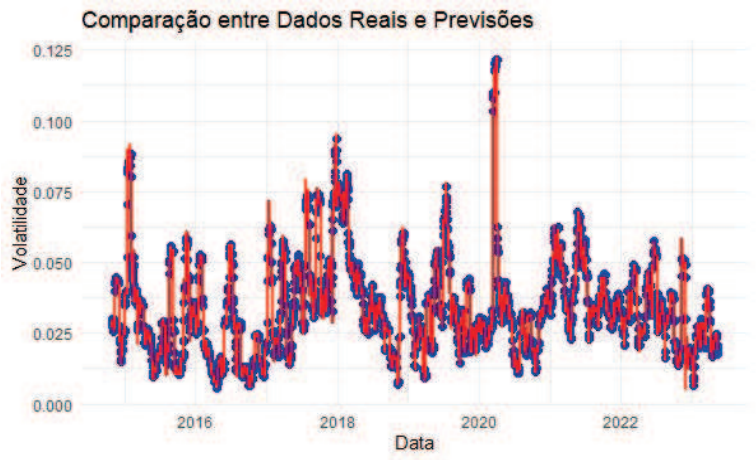

Figura 84: Valores reais (Azuis) x Previstos (Vermelhos) da volatilidade do preço do Bitcoin, utilizando um modelo SARIMA

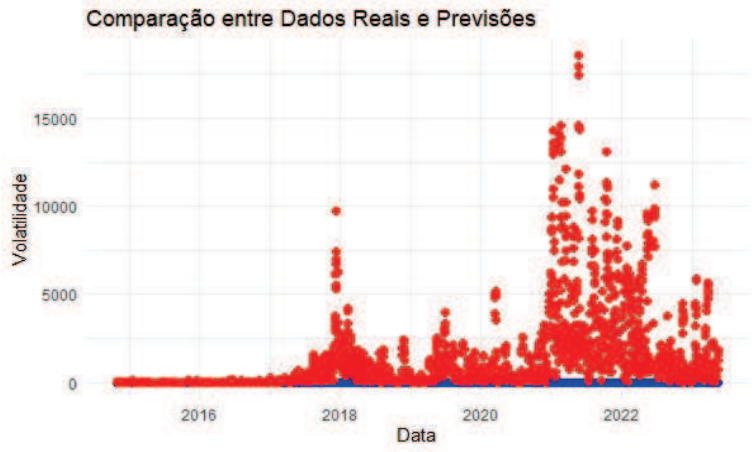

Figura 85: Valores reais (Azuis) x Previstos (Vermelhos) da volatilidade do preço do Bitcoin, utilizando um modelo GARCH

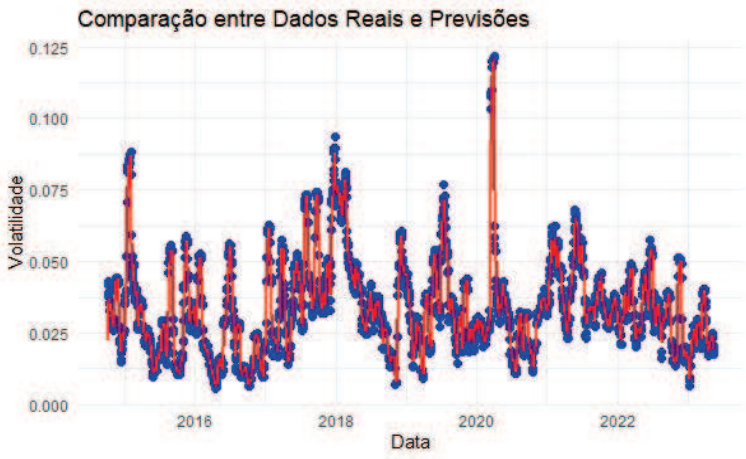

Figura 86:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade do preço do Bitcoin, utilizando um Filtro de Kalman

Os resultados da modelagem do Preço do Bitcoin foram bastante similares aos observados no preço do ouro, com novamente um R^2 de 0,93 para o SARIMA, de 0.943 para o Filtro de Kalman e negativo para o GARCH (as predições deste modelo foram em ordens de grandeza distintas das dos valores).O MAE observado foi de 0,00186 no SARIMA e de 0,00243 no Filtro de Kalman . Observamos que embora tanto o SARIMA como o Filtro de Kalman tenham performado bem nas duas métricas, SARIMA obteve um MAE melhor (mais baixo) e o Filtro de Kalman um R^2 melhor, o que indica que o Filtro de Kalman parece lidar ligeiramente melhor com outliers.

5.2.4.4.3 Análise com EMD

Na análise de volatilidade do preço do bitcoin obtivemos 10 IMF's, mais a residual, o menor número de IMFs de todas as séries. A somatória das previsões ajustadas para cada modelo SARIMA rolante resultou no seguinte gráfico:

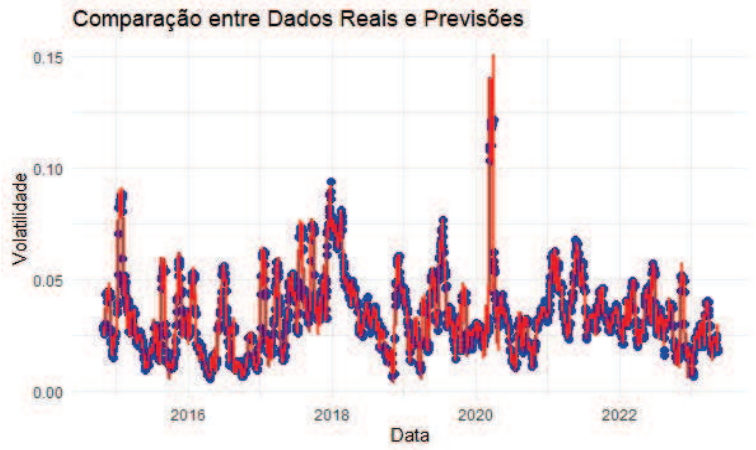

Figura 87:Valores reais(Azuis) x Previstos(Vermelhos) da volatilidade do preço do Bitcoin, utilizando uma combinação de modelos SARIMA pra cada IMF

No caso do Bitcoin, obtivemos um R^2 de 0,9578 e um MAE de 0,00181 e aqui observamos o mesmo padrão do preço do ouro: ambas as métricas ligeiramente melhores do que na análise univariada.

3.2.5 Considerações Gerais sobre as Análises Univariadas

Nesta etapa comentaremos as análises univariadas de forma conjunta. Iniciaremos pelas considerações nos changepoints, seguidas por comentários conjuntos sobre a performance da abordagem de modelagem em janela única, em janela móvel e por fim com a aplicação da decomposição por modo empírico.

3.2.5.1 Considerações sobre a Análise de Changepoints

De uma forma geral, percebemos que o Método PELT-SIC tende a ser um método mais geral, podendo ser utilizado em todas as séries, SONDE somente identificou changepoints na SELIC, razão pela qual não foram exibidos resultados em demais séries. Observa-se que o SONDE performou de forma razoável na única série macroeconômica (SELIC), mas não nas séries de ativos de mercado.

Observamos então que dos datasets avaliados, temos resultados aparentemente consistentes para a SELIC, Ouro e Bitcoin. O Preço do fundo de índice EWZ, por outro lado, aparenta ter demasiadas quebras para ter qualquer utilidade prática. No entanto, é sabido que o Brasil no século XXI, além de ter sofrido impacto de eventos globais (crise do subprime, covid-19) também passou por eventos locais que não foram refletidos nas demais séries (impeachment da presidente Dilma Rousseff, eleição do presidente Lula), isso pode indicar uma série com mais rutura e com maior dificuldade de modelagem.

A seguir, utilizaremos os changepoints encontrados como subsídio na modelagem de cada uma das séries.

3.2.5.2 Considerações sobre a Análise Univariada com ajuste em janela fixa

Nossa primeira abordagem de ajuste de um modelo ARIMA em janela fixa se revelou inadequada para a modelagem de nossas séries temporais. No entanto, dado o observado na análise exploratória e de changepoints, isto era esperado, pois no decorrer da análise exploratória das volatilidades, observamos que de uma forma geral, as séries apresentam comportamento complexo, com pouca ou nenhuma autocorrelação longa ocorrendo. Na seção de análise de changepoints, existem diversos pontos de mudança em todas as séries.

Provavelmente a origem do comportamento complexo das séries está nestes pontos de mudança. Desta forma, uma abordagem utilizando um único algoritmo ajustado a série toda, provavelmente não seria mesmo capaz de fazer predições com performance aceitável.

Nesta análise, não exibiremos os erros absolutos e o R^2 pois, uma vez que premissas do ARIMA foram violadas e os ajustes foram feitos à série como um todo, existe uma grande chance dos modelos terem sobreajuste (*overfit*), isto é, ainda que apresentem boa performance por estas métricas nos valores observados, sua performance em dados futuros muito provavelmente será inadequada.

3.2.5.3 Considerações sobre a Análise Univariada com ajuste em janela móvel

Abaixo está uma tabela contendo as métricas MAE e R^2 para o Filtro de Kalman e Sarima:

Série	Série	Algorítimo	R-Quadrado	MAE
SELIC	SELIC_Volatilidade	SARIMA	0,820948	0,001480
SELIC	SELIC_Volatilidade	Filtro de Kalman	0,816255	0,002257
Ouro	Ouro_Volatilidade	SARIMA	0,975407	0,000457
Ouro	Ouro_Volatilidade	Filtro de Kalman	0,975388	0,000556
BTC	BTC_Volatilidade	SARIMA	0,939052	0,001856
BTC	BTC_Volatilidade	Filtro de Kalman	0,942897	0,002437
EWZ	EWZ_Volatilidade	SARIMA	0,975259	0,000939
EWZ	EWZ_Volatilidade	Filtro de Kalman	0,969100	0,001160

Tabela 4:Métricas MAE e R^2 para análise univariada das séries financeiras em questão utilizando SARIMA e Filtro de Kalman .

O filtro de Kalman teve um bom desempenho, o que era esperado, dado que estamos fazendo predições para o valor imediatamente posterior, que é a situação para a qual o filtro de Kalman foi desenvolvido. No entanto, o Algoritmo SARIMA também se apresentou robusto nas predições, indicando que volatilidades de séries financeiras, embora apresentem estrutura complexa, são localmente divisíveis em regiões que podem ser modeladas com abordagens clássicas.

Foi surpreendente o mau desempenho do modelo GARCH em todas as séries, durante as janelas de treinamento, observamos que houve falhas de ajuste do mesmo em diversas janelas das séries (nesta situação o código utilizava o último modelo ajustado e/ou a média), provavelmente a existência de changepoints e outliers nas séries impediu o correto ajuste.

3.2.5.4 Considerações sobre a Análise com EMD

A seguir apresentaremos uma tabela consolidando os resultados da modelagem com EMD:

						Deltas	
Série	Série	Algorítimo	R-Quadrado	MAE	IMFs (exclusive residual)	R-Quadrado	MAE
SELIC	SELIC_Volatilidade	SARIMA	0,858624	0,002111	11	0,037676	0,000631
Ouro	Ouro_Volatilidade	SARIMA	0,984719	0,000417	12	0,009312	-0,000041
BTC	BTC_Volatilidade	SARIMA	0,957877	0,001813	10	0,018825	-4,292E-05
EWZ	EWZ_Volatilidade	SARIMA	0,981883	0,000802	11	0,006624	-0,000137

Tabela 5: Tabela com a performance consolidada da análise univariada com EMD. As colunas "delta" consistem na diferença entre a métrica observada utilizando EMD e a métrica na análise univariada simples

Observamos aqui uma diferença no comportamento da SELIC contra as demais séries: enquanto Ouro, BTC e EWZ apresentaram melhoria na performance utilizando EMD tanto no R^2 (aumento do valor da métrica) como no MAE (diminuição do valor da métrica), na SELIC observamos melhoria apenas no R^2. Na análise exploratória foi observado um comportamento bastante distinto da Selic em relação às demais séries, então esta diferença está dentro do que seria esperado.

Observamos também que todas as séries apresentaram decomposição em número similar de IMF's e não há, aparentemente nenhuma relação entre o número de IMF's e a melhoria da performance.

Neste estudo, aplicamos o mesmo algoritmo a todas as IMF's, porém uma possível abordagem seria a aplicação de uma análise exploratória a cada IMF de forma separada e a escolha de um algoritmo para cada uma de acordo com seu caráter (determinístico ou estocástico). Todavia, este estudo está fora do escopo do presente trabalho e fica como sugestão para uma próxima pesquisa.

3.2.6 Análise Multivariada

Nas seções anteriores, cada uma das séries foi analisada de forma separada. Nesta sessão faremos uma análise multivariada, aonde objetivamos identificar possíveis inter-relações entre as mesmas. Da mesma forma que nas análises univariadas, iniciaremos com uma análise dos preços e depois das volatilidades, seguidas por uma etapa de modelagem.

5.2.6.1 Análise Multivariada dos preços

A seguir exibiremos todas as séries temporais, normalizadas para facilitar a visualização, e mantendo apenas o período pós 2014, nas quais há valores em comum para todas as séries:

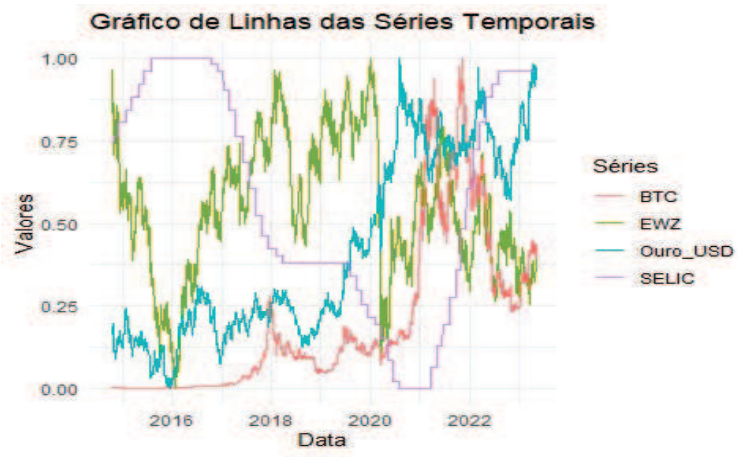

Figura 88:Gráfico contendo todas as séries temporais, normalizadas para estarem na mesma escala:

Não existe aparência visível de nenhum relacionamento entre as séries, a seguir criaremos uma matriz de correlação entre as séries. Uma vez que estamos interessados em relacionamentos não-paramétricos (isto é, não necessariamente lineares, mas se as séries variam na mesma direção), o coeficiente aplicado será o de Spearman:

```
##                 EWZ         BTC    Ouro_USD      SELIC
## EWZ       1.00000000  0.04751789 -0.03755478 -0.4274851
## BTC       0.04751789  1.00000000  0.86433230 -0.5175959
## Ouro_USD -0.03755478  0.86433230  1.00000000 -0.4586247
## SELIC    -0.42748515 -0.51759587 -0.45862475  1.0000000
```

Observamos valores de correlação negativos expressivos entre a SELIC e as demais variáveis. Isto é esperado, pois ciclos de alta de juros reduzem a atratividade de investimentos de risco

em comparação, uma vez que o investimento livre de risco (títulos soberanos denominados na moeda do Estado Emissor) passa a ser uma alternativa mais rentável. Nos demais investimentos, percebemos correlações significativas entre o Bitcoin e o Ouro, indicando que estes ativos apresentam comportamento similar.

Historicamente, o ouro foi considerado como um ativo seguro em tempos de crise, já que não pode ser inflacionado (emitido de forma indiscriminada por pessoas ou instituições), mantendo seu valor independentemente das condições econômicas e políticas. O Bitcoin, que apresenta similaridade no sentido de não ser emitido por nenhuma entidade central, aparentemente está assumindo um papel similar. (Moro, 2017)

5.2.6.2 Análise de Cointegração

Apesar de termos identificado correlações entre as séries, estas podem ser transientes e implicar em relacionamentos espúrios. Seguiremos com uma análise de cointegração entre as variáveis a fim de entendermos mais a fundo os mesmos:

```
##
## ######################
## # Johansen-Procedure #
## ######################
##
## Test type: maximal eigenvalue statistic (lambda max) , with linear trend in
cointegration
##
## Eigenvalues (lambda):
## [1]  2.122672e-02  8.554550e-03  3.427291e-03  1.082204e-03 -1.464644e-18
##
## Values of teststatistic and critical values of test:
##
##           test 10pct  5pct  1pct
## r <= 3 |  2.27 10.49 12.25 16.26
## r <= 2 |  7.21 16.85 18.96 23.65
## r <= 1 | 18.05 23.11 25.54 30.34
## r = 0  | 45.08 29.12 31.46 36.65
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##                   EWZ.l2        BTC.l2  Ouro_USD.l2      SELIC.l2     trend.l2
## EWZ.l2        1.000000000  1.0000000000  1.000000000  1.0000000000  1.000000000
## BTC.l2      -20.218477973 -0.0249649344  0.603323310  0.2997948995 -4.901730880
## Ouro_USD.l2   0.053630538  1.2250340367 -2.768078047 -0.9667401623  4.330739669
## SELIC.l2     -3.068467778  0.3464412634  0.804784447 -1.0670175475  2.795086851
## trend.l2      0.003289619 -0.0004908185  0.001090909  0.0006379468  0.002157069
##
## Weights W:
## (This is the loading matrix)
##
##                  EWZ.l2        BTC.l2   Ouro_USD.l2      SELIC.l2
## EWZ.d      -1.953014e-06 -0.0095944885 -0.0009118778 -0.0006955039
## BTC.d       2.355848e-04 -0.0014401566 -0.0012913143  0.0004147278
## Ouro_USD.d  2.037296e-05 -0.0040402522  0.0009872173  0.0004217728
## SELIC.d    -3.326945e-04 -0.0001928832 -0.0003021246  0.0001095037
##                 trend.l2
## EWZ.d      -1.190136e-19
```

```
## BTC.d       3.322045e-20
## Ouro_USD.d -6.737729e-20
## SELIC.d    -1.527069e-19
```

O teste de Engle-Granger, rejeita a hipótese nula de ausência de cointegração a 99% de significância (estatística de teste: 45.08, valor crítico para 99%: 36.65), mas falha em estabelecer mesmo uma única relação (estatística de teste:18.05, valor crítico para 90%: 23.11).

Observamos pelo auto vetor de maior autovalor (o primeiro) que aparentemente o EWZ e o Bitcoin têm alguma relação de cointegração, aplicaremos uma transformação linear multiplicando o auto vetor pelas séries e plotando a mesma. Esperamos que o resultado seja uma série aparentemente estacionária, caso existam relações de cointegração:

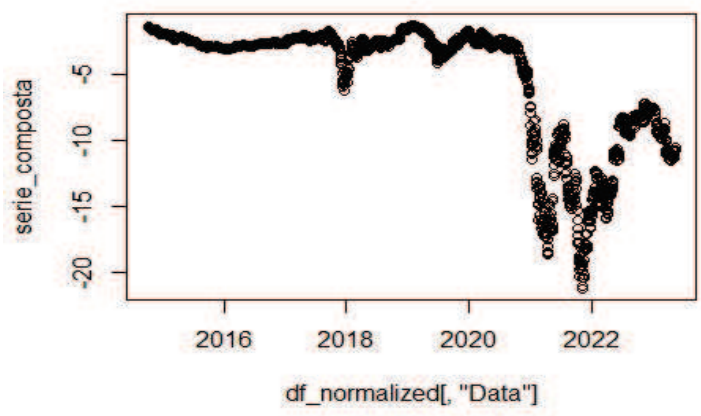

Figura 89:Série composta, contendo os valores das séries multiplicadas por seus respectivos autovalores e somadas

```
##
##  Augmented Dickey-Fuller Test
##
## data:  serie_composta
## Dickey-Fuller = -2.1228, Lag order = 12, p-value = 0.5263
## alternative hypothesis: stationary
```

O gráfico acima mostra uma série aparentemente estacionária até o início de 2020 (quando ocorreu a pandemia da covid-19), a partir daí o resultado é claramente distinto, indicando uma mudança no comportamento das séries, possivelmente um changepoint. O Teste aumentado de Dick-Fulley não rejeitou a hipótese nula de raiz unitária, indicando que a série não é estacionária, provavelmente por conta da rutura observada no período pós-covid-19.

5.2.6.3 Causalidade de Granger

No estudo de causalidade de Granger aplicaremos este algoritmo a combinações duas a duas das séries, para lags de 1 até 10 e exibiremos na sequência apenas os lags considerados relevantes a uma significância de 5%, na tabela abaixo, série 1 é utilizada para prever a série 2:

	Série_1	Série_2	Lag	P_Value
1	EWZ	Ouro_USD	4	0.000477751530133009
2	EWZ	Ouro_USD	5	0.00263748400270941
3	EWZ	Ouro_USD	6	0.0129217804035394
4	EWZ	Ouro_USD	8	0.0445270557455816
5	EWZ	SELIC	4	0.022266987654088
6	EWZ	SELIC	5	0.0234480458953806
7	EWZ	SELIC	6	0.0221264709315748
8	BTC	Ouro_USD	6	0.00414741977369044
9	BTC	Ouro_USD	7	0.0061546515259361
10	BTC	Ouro_USD	8	0.0113819989579396
11	BTC	Ouro_USD	9	0.00166343798759039
12	Ouro_USD	SELIC	1	0.00113243196078301
13	Ouro_USD	SELIC	2	0.00177005770956704
14	Ouro_USD	SELIC	3	0.00888901590758044
15	Ouro_USD	SELIC	4	0.000236426480825053
16	Ouro_USD	SELIC	5	0.000209726506488559
17	Ouro_USD	SELIC	6	0.00326485774100568
18	Ouro_USD	SELIC	7	0.0110083627392977
19	Ouro_USD	SELIC	8	0.0210890235132843
20	Ouro_USD	SELIC	9	0.0445456255001847

Tabela 6: Relações de causalidade de Granger encontradas entre as séries e suas significâncias

Na tabela acima, verificamos que existe relação de causalidade de Granger aonde o EWZ antecipa o Ouro e a SELIC, o BTC antecipa o Ouro e ao mesmo tempo o Ouro antecipa a SELIC. A Existência de relação entre ouro e SELIC revela alguns pontos importantes:

Primeiro, é provável que a relação de antecipação entre EWZ para o Ouro e SELIC seja devida a fatores macroeconômicos e a relação de antecipação ocorra porque o mercado de ações é, provavelmente o ambiente aonde opiniões sobre as mudanças econômicas refletem primeiro.

Como o EWZ é dado por uma cesta de ações brasileiras e a SELIC é a taxa básica de juros do brasil, esta relação se deva à correlação negativa reconhecidamente existente entre taxa de juros e performance do mercado de ações, por conta do aumento do custo de oportunidade da respetiva economia. Ao mesmo tempo, em momentos de crise econômica, aonde as ações perdem valor, o ouro tende a se tornar mais atrativo. (Securato, 2008)

Segundo, A relação entre ouro e SELIC no curto prazo (até 10 lags) é esperada e provavelmente devida a uma relação envolvendo inflação e custo de oportunidade: à medida que a taxa de juros sobe, a posse de ouro perde atratividade devido ao rendimento que poderia ser obtido por meio da aplicação em títulos soberanos. Ao mesmo tempo, a taxa de juros e o ouro tendem a ter uma relação positiva com a inflação: Momentos de alta inflação levam os bancos centrais a subirem as taxas de juros para combater a mesma, ao mesmo tempo, a posse de moeda perde atratividade em relação ao ouro que em geral é considerado um porto seguro nestes momentos. (Securato, 2008)

5.2.6.2 Análise Multivariada das volatilidades

Vamos verificar agora se existe alguma inter-relação entre as volatilidades dos diferentes ativos, iniciaremos com o gráfico de dispersão, contendo as séries de volatilidade normalizadas.

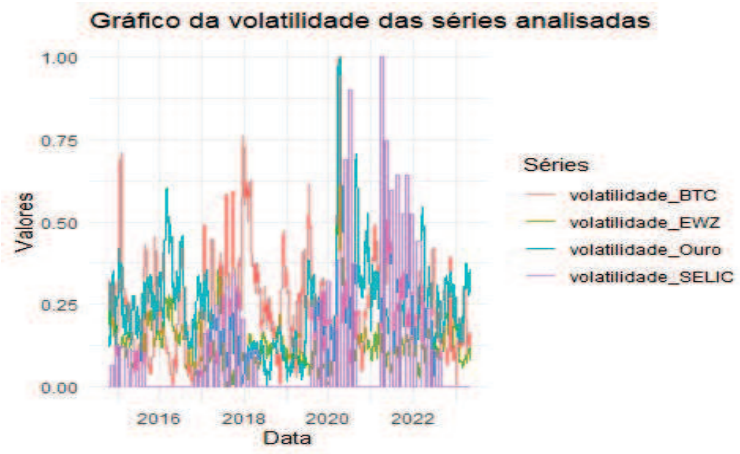

Figura 90 contendo as volatilidades das séries históricas normalizadas

As séries de volatilidade parecem ter algum comportamento próximo, mas visualmente é difícil de identificá-lo, da mesma forma que fizemos com as séries de preços, avaliaremos a correlação entre as volatilidades:

```
##                    volatilidade_EWZ volatilidade_BTC volatilidade_Ouro
## volatilidade_EWZ          1.0000000      -0.14153051        0.33904016
## volatilidade_BTC         -0.1415305       1.00000000       -0.08057075
## volatilidade_Ouro         0.3390402      -0.08057075        1.00000000
## volatilidade_SELIC       -0.1601527       0.17501880        0.06707713
##                    volatilidade_SELIC
## volatilidade_EWZ          -0.16015274
## volatilidade_BTC           0.17501880
## volatilidade_Ouro          0.06707713
## volatilidade_SELIC         1.00000000
```

As volatilidades apresentam um relacionamento muito menor entre si do que as séries de preços. A correlação mais significativa é a observada entre a volatilidade do ouro e do EWZ, provavelmente devido aos eventos da crise de 2008 quando observamos aumento da volatilidade em ambos os ativos.

5.2.6.2.1 Análise de Cointegração

Novamente, estes relacionamentos podem ser transientes, seguiremos com o teste de Engle-Granger para identificarmos se existe alguma relação de longo prazo:

```
##
## ######################
## # Johansen-Procedure #
## ######################
##
## Test type: maximal eigenvalue statistic (lambda max) , with linear trend in
cointegration
```

```
##
## Eigenvalues (lambda):
## [1] 4.205910e-02 2.207982e-02 1.683004e-02 1.042622e-02 1.255784e-20
##
## Values of teststatistic and critical values of test:
##
##           test 10pct  5pct  1pct
## r <= 3 | 22.02 10.49 12.25 16.26
## r <= 2 | 35.66 16.85 18.96 23.65
## r <= 1 | 46.91 23.11 25.54 30.34
## r = 0  | 90.28 29.12 31.46 36.65
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##                       volatilidade_EWZ.l2 volatilidade_BTC.l2
## volatilidade_EWZ.l2          1.0000000000        1.0000000000
## volatilidade_BTC.l2          5.0704963848       40.7247229423
## volatilidade_Ouro.l2        -0.5810324492       -3.5582735470
## volatilidade_SELIC.l2       -9.1365092714        8.6349916424
## trend.l2                     0.0007397187       -0.0008274207
##                       volatilidade_Ouro.l2 volatilidade_SELIC.l2    trend.l2
## volatilidade_EWZ.l2           1.000000e+00           1.00000000
## volatilidade_BTC.l2          -1.106021e-01         -1.014382e-01 -1.08443064
## volatilidade_Ouro.l2         -7.826015e-01          6.300302e-01 -0.13904655
## volatilidade_SELIC.l2         3.981262e-02         -4.046624e-02 -1.54749043
## trend.l2                      1.414211e-05         -8.663878e-06  0.03640727
##
## Weights W:
## (This is the loading matrix)
##
##                      volatilidade_EWZ.l2 volatilidade_BTC.l2
## volatilidade_EWZ.d          0.0004217020        5.545723e-05
## volatilidade_BTC.d         -0.0002316160       -7.583384e-04
## volatilidade_Ouro.d         0.0005778891        1.363071e-04
## volatilidade_SELIC.d        0.0079055706       -3.045446e-04
##                      volatilidade_Ouro.l2 volatilidade_SELIC.l2      trend.l2
## volatilidade_EWZ.d           -0.012706070          -0.006825082  2.348457e-21
## volatilidade_BTC.d            0.001597802          -0.007911308 -6.543671e-21
## volatilidade_Ouro.d           0.027803256          -0.010468721 -3.629492e-22
## volatilidade_SELIC.d          0.009446665           0.010713690  1.160582e-20
```

O resultado dos testes de cointegração das variáveis indica um relacionamento muito mais próximo entre as mesmas, a estatística de teste para a existência de cointegração entre a totalidade das séries é de 22.02, o que nos permite afirmar sua existência com 99% de significância (valor crítico 16.26). Vamos avaliar o gráfico da

série composta: assim como na análise de preços, esperamos que as séries originais sejam cointegradas se houver estacionariedade da série composta:

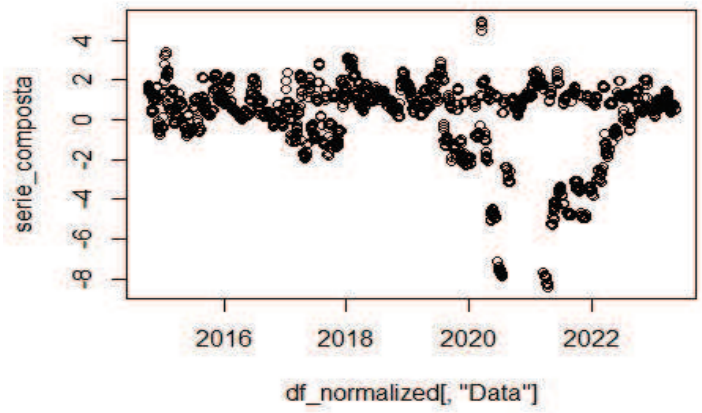

Figura 91: Série histórica composta, constituída pela soma das volatilidades, multiplicadas pelos respetivos autovalores

A série composta aparenta estacionariedade visualmente, exceto pelo período da Pandemia aonde houve um maior desvio, mas com convergência ao final da série. Faremos agora o teste aumentado de Dickey-Fuller para análise de estacionariedade:

```
##
##  Augmented Dickey-Fuller Test
##
## data:  serie_composta
## Dickey-Fuller = -8.0453, Lag order = 12, p-value = 0.01
## alternative hypothesis: stationary
```

O teste aumentado de Dick-Fulley rejeitou a hipótese nula, indicando que as volatilidades são cointegradas entre si.

5.2.6.2.2 Causalidade de Granger

Foi feita uma análise de causalidade de Granger entre as volatilidades das séries financeiras, nos mesmos padrões da feita para os preços, os valores estão abaixo:

	Série_1	Série_2	Lag	P_Value
1	volatilidade_EWZ	volatilidade_BTC	5	0.0475061653727683
2	volatilidade_EWZ	volatilidade_BTC	7	0.0108423895697872
3	volatilidade_EWZ	volatilidade_BTC	8	0.0237037621266731
4	volatilidade_EWZ	volatilidade_BTC	9	0.0499156442550267
5	volatilidade_EWZ	volatilidade_BTC	10	0.00324125865651629
6	volatilidade_EWZ	volatilidade_Ouro	1	0.0230775919316556
7	volatilidade_EWZ	volatilidade_Ouro	2	0.000592186092184117
8	volatilidade_EWZ	volatilidade_Ouro	3	0.00046782367237178
9	volatilidade_EWZ	volatilidade_Ouro	4	0.02286286564976
10	volatilidade_EWZ	volatilidade_Ouro	6	0.0338451204785297
11	volatilidade_EWZ	volatilidade_Ouro	10	0.0256262622196898
12	volatilidade_EWZ	volatilidade_SELIC	2	0.00910195442301026
13	volatilidade_EWZ	volatilidade_SELIC	3	0.00198901363694552
14	volatilidade_EWZ	volatilidade_SELIC	7	0.0129867803622961
15	volatilidade_EWZ	volatilidade_SELIC	8	0.0318287620110131
16	volatilidade_EWZ	volatilidade_SELIC	9	0.0467636738385735
17	volatilidade_BTC	volatilidade_Ouro	1	0.0196710441984217
18	volatilidade_BTC	volatilidade_Ouro	4	0.00198760990184327
19	volatilidade_BTC	volatilidade_Ouro	5	0.0027368484245481
20	volatilidade_BTC	volatilidade_Ouro	6	0.0240622854058245
21	volatilidade_BTC	volatilidade_SELIC	1	0.0160360263363357
22	volatilidade_BTC	volatilidade_SELIC	2	0.0477935192754777
23	volatilidade_BTC	volatilidade_SELIC	3	0.00328832864437062
24	volatilidade_BTC	volatilidade_SELIC	4	0.00641962365982175
25	volatilidade_Ouro	volatilidade_SELIC	3	0.0338510336992684

Tabela 7:Causalidade de Granger das volatilidades:

Nesta análise, chamam a atenção o maior número de relações encontradas em relação a análise de preços (25 e 20, respetivamente). Observamos também que a ordem das relações é similar e que aqui temos uma relação nova com o Bitcoin causando a SELIC. De qualquer forma, a existência de maior número de relações indica que uma modelagem multivariada entre as séries apresente uma boa probabilidade de sucesso em relação a de preços.

5.2.6.3 Modelagem Multivariada

Na análise exploratória encontramos relações de cointegração e de causalidade de Granger entre as séries, tais fatos sugerem que uma análise multivariada possa ser útil na modelagem das séries.

Nesta análise, excluiremos a taxa SELIC, pois embora as análises de causalidade de Granger e de cointegração tenham revelado um possível relacionamento com as demais variáveis, ela é conceitualmente diferente das demais: enquanto o EWZ o BTC e o Ouro são

ativos com preço cotado em bolsa, a SELIC é uma taxa de juros soberana. Este fato introduz alguns comportamentos na SELIC que são distintos das demais, como por exemplo o fato de permanecer com volatilidade zero por alguns períodos, devido a intervenções da autoridade monetária com a intenção de estabilizar a mesma. Estes períodos com valor zero fazem com que o modelo VAR não seja capaz de retornar predições nestes momentos.

Neste estudo, optamos pelo uso da técnica de autorregressão vetorial (VAR) para as séries, da mesma forma que na análise univariada, optamos por efetuar predições rolantes das séries, utilizado os 22 valores imediatamente anteriores. Os resultados estão na tabela abaixo:

Série	Série	R-Quadrado	MAE
EWZ	EWZ_Volatilidade	-1,157633	0,030805
BTC	BTC_Volatilidade	0,181811	0,061386
Ouro	Ouro_Volatilidade	0,2047164	0,055079

Tabela 8:Tabela contendo R^2 e MAE de uma análise VAR para todas as séries de volatilidade

Aqui, observamos que o R^2 das predições é baixo, chegando a negativo no caso do EWZ, mas o MAE possui erros relativamente baixos, vamos plotar agora gráficos da predição de cada uma das variáveis contra suas séries observadas:

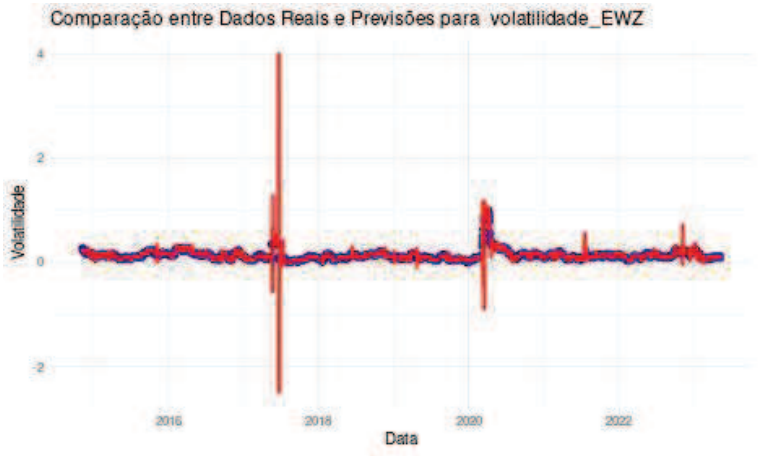

Figura 92: Dados reais(azuis) e previstos (vermelhos) para a volatilidade do índice EWZ, por meio de modelo VaR

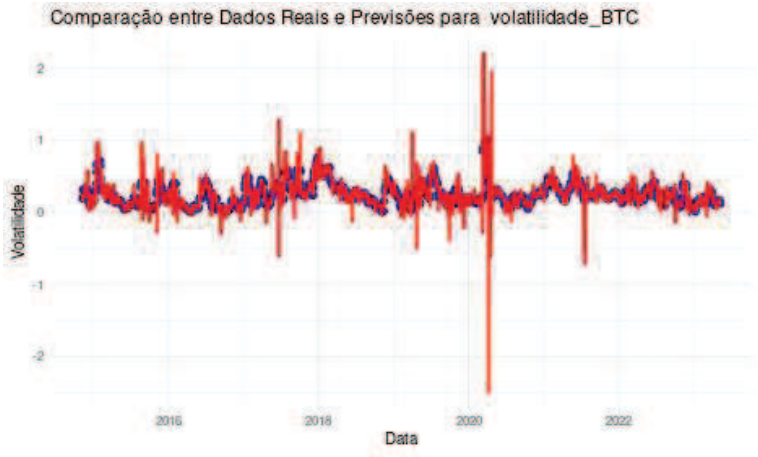

Figura 93: Dados reais(azuis) e previstos (vermelhos) para a volatilidade do preço do Bitcoin, por meio de modelo VAR

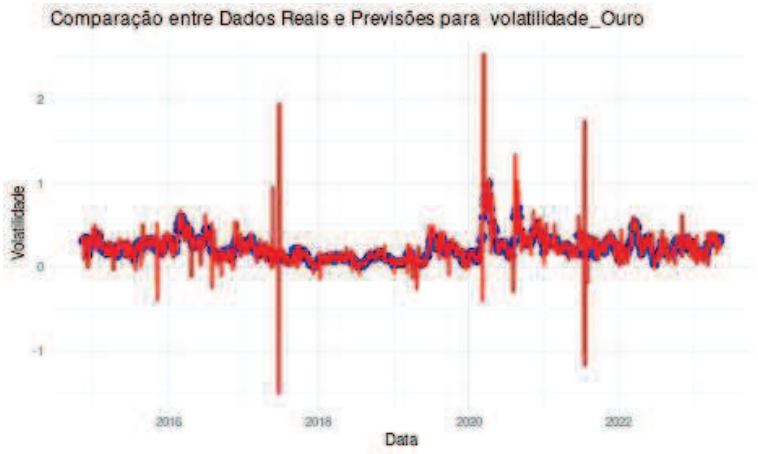

Figura 94: Dados reais(azuis) e previstos (vermelhos) para a volatilidade do preço do Ouro por meio de modelo VAR

Observamos visualmente que, de uma forma geral, os valores previstos estão bastante próximos dos valores reais. No entanto, em todas as séries, existem outliers em todas as séries, com valores significativamente discrepantes da série original, sendo que em alguns casos, o valor previsto é negativo, o que conceitualmente não é possível pela definição de volatilidade.

Uma vez que o R^2 eleva os resíduos ao quadrado, estes outliers tem uma influência muito maior no cálculo desta métrica em relação ao MAE, que não o faz. Desta forma, observamos que, de uma forma geral, um algoritmo VAR pode ser adequado para modelar volatilidades de séries financeiras, no entanto, a performance do mesmo em todas as séries foi inferior tanto em MAE como –principalmente – no R^2 em relação às abordagens univariadas.

Chamou a atenção obter-se um valor negativo de R^2 para o EWZ. Isto ocorreu porque definimos esta grandeza como $R^2 = 1 - \frac{\sum_{i=1}^{k}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{k}(y_i - \bar{y}_i)^2}$ isto é, caso $\sum_{i=1}^{k}(y_i - \hat{y}_i)^2 > \sum_{i=1}^{k}(y_i - \bar{y}_i)^2$ teremos $R^2 < 0$, isto significa que nosso modelo multivariado apresentou erros maiores do que um estimador baseado na média para o EWZ.

4 Conclusões e Perspetivas

Observamos que a volatilidade das séries temporais financeiras investigadas no presente estudo apresenta diferentes comportamentos e uma grande diversidade de pontos de mudança. Relativamente a esta análise, utilizando os modelos PELT com penalidade SIC e o SONDE, verificou-se que o método O PELT-SIC se mostrou mais robusto, tendo dentre os pontos de mudança detetados por ele alguns consistentes com eventos de mercado conhecidos, tais como crises financeiras, eventos políticos e a pandemia de covid-19.

O método SONDE, por sua vez, apresentou grande flutuação no número de changepoints detetados de acordo com os hiperparâmetros fornecidos. Obtivemos um resultado interessante com este algoritmo para a SELIC, na qual, foram detetados changepoints consistentes com eventos de mercado (mencionados anteriormente), mas não com as demais séries. Sugere-se que um trabalho posterior avalie em quais situações o SONDE pode ou não ser aplicado e que defina métodos objetivos de ajuste de seus hiperparâmetros, um possível caminho seriam outras séries macroeconômicas (inflação, desemprego, etc.) além da taxa de juros.

A análise do gráfico de recorrência revelou existência de ciclos na série da taxa de juros SELIC, a qual foi confirmada pelo periodograma. Observamos a existência de períodos longos, de mais de um ano, os quais são consistentes com os ciclos econômicos, os quais apresentam prazo longo.

A Presença do grande número de pontos de mudança impede que as séries sejam adequadamente modeladas por um único modelo ajustado a série completa. No entanto, uma abordagem rolante, na qual ajustamos modelos a cada segmento da série, permite resultados satisfatórios neste desafio.

Dentre as abordagens univariadas, foi possível confirmar que um modelo clássico como o SARIMA, combinado com uma abordagem em janela móvel, é capaz de gerar bons resultados. Assim como o Filtro de Kalman. Chamou a atenção o fato dos modelos GARCH revelarem um mau desempenho, não tendo sido capazes de se ajustar a série de forma satisfatória embora sejam a abordagem mais comumente recomendadas pela literatura para modelagem de volatilidade.

Observamos que a decomposição por modo empírico das séries gera uma melhoria nas métricas de performance calculadas (MAE e R^2). Em estudos posteriores, aonde estivermos interessados em uma única série, seria interessante aplicar uma análise exploratória em cada IMF e utilizar algoritmos mais adequados para cada uma delas. A Aplicação do Teorema de Takens pode ser necessária nas IMF's que se mostrarem claramente determinísticas.

Neste trabalho, não aplicamos a transformação de box-Cox em nenhuma etapa, devido à menções na literatura de que esta tende a não melhorar a performance das predições, no entanto, seria interessante em um trabalho futuro, avaliar o impacto da mesma nas métricas de performance e, cada etapa a fim de comprovar ou não esta afirmação.

Por fim, observamos que embora existam indícios de relação entre as séries, uma abordagem multivariada utilizando VAR em janela rolante apresentou um ajuste inferior ao obtido nas análises univariadas (tanto como e sem EMD) principalmente por conta dos valores extremos. Todavia, isso pode ser devido a premissa de linearidade do VAR. Em trabalhos posteriores, sugere-se a aplicação de métodos multivariados não-lineares a fim de observar se os mesmos apresentam um melhor desempenho.

A gestão de portfólios de investimento pode beneficiar dos resultados do estudo aqui realizado, por exemplo: a teoria moderna de portfólio sugere o uso do desvio padrão passado como uma mensuração da volatilidade. Todavia, os resultados aqui permitem o uso de um valor previsto para a volatilidade que é mais consistente com o valor real, permitindo uma mensuração mais adequada do risco assumido.

Outro possível uso é como insumo em processos de CRM (*Customer Relationship Management*): aumento de volatilidade de um ativo pode gerar desinteresse por parte de clientes com abordagem mais conservadora (visando proteger o valor investido) no mesmo, porém, pode ter o efeito oposto em clientes de perfil arrojado. Este fato permite uma recomendação de investimento mais adequada a cada perfil.

Neste trabalho exploramos um conjunto restrito de séries. Sugere-se a aplicação desta metodologia em demais séries financeiras, tais como índices de bolsas, taxas de câmbio, dentre outros.

Em estudos posteriores, nos quais sejam evidentes periodicidades mais curtas nas séries, sugere-se a aplicação das transformadas de Fourier para a deteção de ciclos senoidal (movimento periódico em formato de seno) e/ou as transformadas wavelets com o objetivo de detetarmos ciclos mais irregulares, de formato distinto do senoidal. O Uso do método de decomposição LOESS pode ser uma alternativa interessante para a remoção da componente sazonal antes da análise, e comparar com a EMD já utilizada neste trabalho.

Sugere-se também estudos de causalidade de Granger/cointegração entre mais séries temporais financeiras e também com as chamadas “séries alternativas” (Séries temporais não-financeiras, mas que podem apresentar algum poder preditivo sobre as financeiras), em sendo encontrada alguma causalidade, a série alternativa pode ser usada em uma abordagem hibrida, aonde modelos clássicos univariados efetuam predição da parte estocástica da série financeira e uma abordagem determinística para a parte que dependa da série alternativa, por exemplo por meio do uso da mesma como variável explicativa em um modelo de *machine learning*.

Concluímos, sobre um conjunto de casos práticos, que a volatilidade das séries financeiras podem ser previstas com modelos clássicos; que a decomposição das séries, em especial pelo método EMD, apresenta um ganho nas métricas de avaliação destes modelos; e estes resultados aplicados dentro da indústria financeira. Porém, conclui-se que existem áreas em aberto, por exemplo o problema do estudo de changepoints ainda é uma área aonde a investigação pode continuar a ser desenvolvida.

5 Bibliografia

(2023). Fonte: World Gold Council: https://www.gold.org

A. Garcia, C. (2022). nonlinearTseries: Nonlinear Time Series Analysis. Fonte: https://github.com/constantino-garcia/nonlinearTseries

Albertini, M. K., & Mello, R. F. (11 de 03 de 2007). A Self-Organizing Neural Network for Detecting Novelties. *Proceedings of the 2007 ACM symposium on Applied computing*, pp. 462-466. Fonte: https://dl.acm.org/doi/abs/10.1145/1244002.1244110?casa_token=MUqXvryUCX4AAAAA:Y8a1Ts231F5JrxrgPVwYVmD1V5f1Vidrp03bQiHFjQXRj82wUyemwrtdQ69yBMo_iLZc3L7zJLxBS2I

Artes, R. (2014). Acesso em 2024, disponível em IME USP: https://www.ime.usp.br/~mbranco/MedidasdeAssimetria_2014.pdf

Banco Central do Brasil. (08 de 02 de 2024). *bcb*. Fonte: Taxa Selic: https://www.bcb.gov.br/controleinflacao/taxaselic

Bastos, J. A., & Caiado, J. (2010). Recurrence quantification analysis of global stock markets. *Physica A*.

Bedendo, M., & Hodges, D. S. (june de 2009). The dynamics of the volatility skew: A Kalman filter approach. *Journal of Banking & Finance, 33*(6), pp. 1156-1165. doi:https://doi.org/10.1016/j.jbankfin.2008.12.014

Brownlee, J. (2020). *Time Series Forecasting with Python.*

C. Hull, J. (2009). *Fundamentos dos Mercados Futuros e de Opções.* São Paulo: BM&F Bovespa.

Di Narzo, F. a. (2019). tseriesChaos: Analysis of Nonlinear Time Series. Fonte: https://rdrr.io/cran/tseriesChaos/man/false.nearest.html

Diez, D., Çetinkaya-Russel, M., & D Barr, C. (2019). *Open Intro Statistics.*

E. Meyer, P. (2022). infotheo: Information-Theoretic Measures. Fonte: http://homepage.meyerp.com/software

ECKMAN, J.-P., OLIFFSON KAMPHORST, S., & Ruelle, D. (1987). Recurrence Plots of Dynamical Systems. *EUROPHYSICS LETTERS*.

Fava, V. L., & Alves, D. C. (1998). Longa Persistência Nas Taxas De Inflação. *Brazilian Review of Econometrics*. doi:https://doi.org/10.12660/bre.v18n21998.2837

Flandrin, P., Rilling, G., & Gonçalvés, P. (02 de 2004). Empirical Mode Decomposition as a Filter Bank. *IEEE SIGNAL PROCESSING LETTERS*.

Géron, A. (2019). *Mãos a Obra: Aprendizado de Máquina com Scikit-Learn e Tensorflow.* Rio de Janeiro: AltaBooks.

Google, Inc. (2015). Hidden Technical Debt in Machine Learning Systems. Fonte: https://papers.neurips.cc/paper/5656-hidden-technical-debt-in-machine-learning-systems.pdf

Guazzeli Benatti, V. (2020). APREÇAMENTO DE OPÇÕES EUROPEIAS EM DISTRIBUIÇÕES NÃO GAUSSIANAS. *Tese de especialização*. (E. P. Paulo, Ed.) São Paulo.

Hastie, T., Tibshirani, R., & Friedman, J. (2008). *The Elements of Statistical Learning.* Stanford: Springer.

Helske J, L. P. (2021). Introducing libeemd: A program package for performing the ensemble empirical mode decomposition. doi:https://doi.org/10.1007/s00180-015-0603-9

Hossain, Z., Rahman, A., Hossain, M., & Hasan Karami, j. (08 de 2018). Over-Differencing and Forecasting with Non-Stationary Time Series Data. *Dhaka University Journal of Science*. Fonte: https://www.banglajol.info/index.php/DUJS/article/view/54568/38340#:~:text=In%20time%20series%20analysis%2C%20over,stationarity%20of%20time%20series%20data.

IShares. (2023). Fonte: IShares by BlackRock: https://www.ishares.com/us/products/239612/ishares-msci-brazil-capped-etf

Ishii, R. P., Rios, R. A., & Mello, R. F. (2011). Classification of time series generation processes using experimental tools: a survey and proposal of an automatic and systematic approach. *International Journal of Computational Science and Engineering*.

J Msigwa, O. (23 de 08 de 2022). Data Science and Machine Learning — Neural Network (Part 02): Feed forward NN Architectures Design. *MQL5*. Fonte: https://www.mql5.com/en/articles/11334

Killick R, H. K. (2022). changepoint: An R package for changepoint analysis. Fonte: https://CRAN.R-project.org/package=changepoint

Kohonen, T., & Honkela, T. (2007). *Kohonen network*. Fonte: Scholarpedia: http://www.scholarpedia.org/article/Kohonen_network

Mankiw, N. G. (2015). *Macroeconomia.* Rio de Janeiro: LTC.

Mohd Razali, N., & Wah, Y. B. (2011). Power comparisons of Shapiro-Wilk, Kolmogorov-Smirnov,Lilliefors and Anderson-Darling tests. *Journal of Statistical Modeling and Analytics*. Shah Alam, Selangor, Malaysia. Fonte: https://www.nbi.dk/~petersen/Teaching/Stat2017/Power_Comparisons_of_Shapiro-Wilk_Kolmogorov-Smirn.pdf

Morettin, P. (2017). *Econometria Financeira: Um Curso em Séries temporais financeiras.* São Paulo: Blucher.

Morettin, P., & Toloi, C. (2018). *Análise de Séries Temporais, Modelos Lineares univariados.* São Paulo: Blucher.

Moro, P. G. (2017). Bitcoin, Uma Análise de riscos e oportunidades do ponto de vista financeiro. São Paulo. Fonte: https://www.linkedin.com/in/pedro-guilherme-frade-

moro-79046579/overlay/1506881417544/single-media-viewer/?profileId=ACoAABCejBwBLNHGHwn5mqjFw7zfbnbfv7qKjiY

N. Marwan, M. C. (2007). Recurrence Plots for the Analysis of Complex Systems. *Physics Reports*, pp. 438(5-6), 237-329.

N.E. Huang, e. a. (2006). Fonte: The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis: http://www.keck.ucsf.edu/~schenk/Huang_etal98.pdf

Nakamoto, S. (2008). *Bitcoin.org.* Fonte: Bitcoin: https://bitcoin.org/bitcoin.pdf

Narkowich, F. J., & Boggess, A. (2009). *A First Course in Wavelets using Fourier Analysis.* Wiley.

Nesbitt, J. (27 de 10 de 2016). Fonte: NumXL Pro: https://support.numxl.com/hc/en-us/articles/214621606-GARCH-AIC-Akaike-s-Information-Criterion-AIC-of-an-GARCH-Model

Nicholas Taleb, N. (2008). *The Black Swan.*

Ortis-Gracia, L., & W. Oosterlie, C. (2016). A HIGHLY EFFICIENT SHANNON WAVELET INVERSE FOURIER TECHNIQUE FOR PRICING EUROPEAN OPTIONS . *Society for industrial and applied mathmatics*.

P. Chan, E. (2021). *Quantitative Trading.* New Jersey: Wiley.

Pollock, D. (23 de 03 de 2023). *CONTINUOUS-TIME STOCHASTIC PROCESSES.* Fonte: University of Leicester: https://www.le.ac.uk/users/dsgp1/COURSES/DERIVATE/PROCESSES.PDF

Potsdam Institute for Climate Impact Ressearch . (2023). Fonte: RECURRENCE PLOTS AND CROSS RECURRENCE PLOTS: http://www.recurrence-plot.tk/glance.php

Póvoa, A. (2012). *Valuation, Como Precificar Ações.* Rio de Janeiro: Elsevier.

R Core Team. (2021). R: A Language and Environment for Statistical Computing. Vienna, Austria. Fonte: https://stat.ethz.ch/R-manual/R-devel/library/stats/html/box.test.html

R. B. Cleveland, W. S. (1990). STL: A Seasonal-Trend Decomposition Procedure Based on Loess. *Journal of Official Statistics*. Fonte: https://www.wessa.net/download/stl.pdf

Reis, E., Melo, P., Andrade, R., & Calapez, T. (1996). *Estatística Aplicada.* (Sílabo, Ed.) Lisboa, Portugal: Sílabo.

Rilling, G., & Flandrin, P. (01 de 2008). One or Two Frequencies? The Empirical Mode Decomposition Answers. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*.

Rohr, A. (2014). Fonte: G1: https://g1.globo.com/tecnologia/noticia/2014/03/reportagem-identifica-satoshi-nakamoto-criador-do-bitcoin.html

Rynne, B., & Youngson, M. (2000). *Linear Functional Analysis.* Edinburgh: Springer.

Securato, J. R. (2008). *Cálculo FInanceiro das Tesourarias.* São Paulo: Saint Paul.

Shumway, R., & Stoffer, D. (2016). *Time Series Analysis and its Applications.* New York: Springer.

Takens, F. (1981). Detecting strange attractors in turbulence. *Dynamical Systems and Turbulence, Lecture Notes in Mathematics*. Fonte: https://link.springer.com/content/pdf/10.1007/BFb0091924.pdf

Teixeira, L. (2012). *Análise de change-points em séries temporais.* Universidade do Minho. Fonte: https://hdl.handle.net/1822/23451

Toledo, T. J. (Feb1 de 2022). *Time series forecasting with dynamical systems methods.* Fonte: Towards Data Science: https://towardsdatascience.com/time-series-forecasting-with-dynamical-systems-methods-fa4afdf16fd0

Toloi, C., & Morettin, P. (2020). *Análise de Séries Temporais - Modelos Multivariados e não lineares - volume 2.* São Paulo: Blucher.

Trading Economics. (08 de 02 de 2024). Fonte: Trading Economics: https://tradingeconomics.com/country-list/interest-rate

University of Albany. (23 de 03 de 2023). *Financial Economics Slides.* Fonte: University of Albany: https://www.albany.edu/~bd445/Economics_466_Financial_Economics_Slides_Spring_2014/Testing_the_Random-Walk_Theory.pdf

Van Boxtel, G. e. (2021). gsignal: Signal processing. Fonte: https://github.com/gjmvanboxtel/gsignal

Walpole E., R., & Myers H., R. (1989). Probability and Statistics for Engineers and Scientists. New York.

Xiannong, M. (2002). Fonte: Buckenell University - college of engineering: https://www.eg.bucknell.edu/~xmeng/Course/CS6337/Note/master/node44.html

Zeileis A, H. T. (2002). *Diagnostic Checking in Regression Relationships.* Fonte: R-CRAN: https://rdrr.io/cran/lmtest/man/grangertest.html

6 Apêndices

6.1 Códigos

6.1.1 Análise Exploratória

````
---
title: "Análise exploratória - word"
author: "Pedro Guilherme Frade Moro"
date: '2023-08-01'
output: word_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```

```{r}
# Carregar o pacote ggplot2 (plotar gráficos)
library(ggplot2)
# Carregar o pacote dplyr (operações com dataframes):
library(dplyr)
# carregar tidyr (manipular dados em formato longo)
library(tidyr)
#Carregar pacote tseries (ferramentas de séries temporais)
library(tseries)
#Carregar pacote nonlinearTseries, que contém ferramentas para séries não lineares
library(nonlinearTseries)
#pacote stats, contendo testes de aleatoriedade:
````
````

```
library(stats)
#pacote forecast, contendo auto.arima
library(forecast)
#pacote zoo, contendo funções rolantes
library(zoo)
#pacote urca, metodo para aplicação da cointegração:
library(urca)
library(lattice)
library(tseries) #t series para o runs test
```

Função para aplicação do teste de friedman

```{r}
# Função para aplicar o teste de Friedman
apply_friedman_test <- function(data, period,column_name) {
  # Agrupar os dados em blocos de acordo com a periodicidade
  # Calcular o número de linhas a serem mantidas
  n_rows <- floor(nrow(data) / period) * period

  # Truncar o dataframe para ter um número de linhas múltiplo da periodicidade
  data <- data[1:n_rows, ]
  data <- as.data.frame(data)

  data$block <- rep(1:(nrow(data) / period), each = period)
  data$time <- rep(1:period, times = nrow(data) / period)
  test_column <- data[[column_name]]
```

```
  # Aplicar o teste de Friedman
  friedman_result <- friedman.test(test_column, groups=data$time, blocks = data$block)
  return(friedman_result)
}
```

Função para transforar em "Runs" para runs test:

```{r}
Binarize_Factorize <- function(vetor){

# Inicializa o vetor de resultado com o primeiro valor substituído por zero
resultado <- numeric(length(vetor))
resultado[1] <- 0

# Aplica a lógica para os valores subsequentes
for (i in 2:length(vetor)) {
  if (vetor[i] > vetor[i - 1]) {
    resultado[i] <- 1
  } else {
    resultado[i] <- 0
  }
}
return(factor(resultado))
}
```

Função para Calcular matriz de Recorrência:

```{r}
library(ggplot2)

recurrence_plot <- function(dados, eps = 0.1, dimensao = 1, tempo = 1, lmax = 1000) {
  l <- length(dados)  # Comprimento total da série temporal
  if (l > lmax) {
    dados <- dados[(l - lmax + 1):l]  # Considera apenas as últimas lmax linhas
    l <- lmax
  }

   # Aplica Min-Max Scaler aos dados
  dados<- (dados-min(dados))/(max(dados)-min(dados))

  # Inicialize a matriz de recorrência
  recurrence_matrix <- matrix(0, nrow = l, ncol = l)

  # Preencha a matriz de recorrência
  for (i in 1:l) {
    for (j in 1:l) {
      if (abs(dados[i] - dados[j]) < eps) {
        recurrence_matrix[i, j] <- 1
      }
    }
  }

  # Converta a matriz de recorrência em um formato adequado para o ggplot2
  recurrence_data <- reshape2::melt(recurrence_matrix)

  return(recurrence_data)
}
```

```

##Análise da taxa SELIC

A Taxa SELIC é a taxa básica de juros da economia brasileira. Ela consiste em na taxa utilizando em empréstimos que utilizam títulos públicos Federais Brasileiros como garantias, de prazo de um dia - overnight - entre instituições financeiras. como o Comitê de política monetária - COPOM - Fixa uma meta para esta taxa, o Banco central pode atuar neste mercado, oferecendo ou tomando empréstimos com a finalidade de trazer a taxa para os valores definidos nesta meta. [https://www.bcb.gov.br/controleinflacao/taxaselic]

```{r}

# Ler o arquivo CSV de taxas de juros

df_SELIC <- read.csv("Taxa SELIC.csv", sep = ";")

# Converter a coluna da série temporal para o formato de data

df_SELIC$Data <- as.Date(df_SELIC$Data, format = "%d/%m/%Y")

# Converter a coluna de Valor para número rea

# Verificar o número de vírgulas em cada valor

num_virgulas <- sapply(strsplit(df_SELIC$SELIC, ","), function(x) length(x) - 1)

# Substituir a primeira vírgula por nada e a segunda vírgula por ponto, se houver duas vírgulas

df_SELIC$SELIC[num_virgulas == 2] <- gsub(",", "", df_SELIC$SELIC[num_virgulas == 2], fixed = TRUE)  # Substituir a primeira vírgula por nada

df_SELIC$SELIC[num_virgulas == 2] <- gsub(",", ".", df_SELIC$SELIC[num_virgulas == 2], fixed = TRUE)  # Substituir a segunda vírgula por ponto

# Substituir uma única vírgula por ponto, se houver apenas uma vírgula

df_SELIC$SELIC[num_virgulas == 1] <- gsub(",", ".", df_SELIC$SELIC[num_virgulas == 1], fixed = TRUE)


```
```

```
# Converter para tipo numérico
df_SELIC$SELIC <- as.numeric(df_SELIC$SELIC)
```

```{r}
summary(df_SELIC$SELIC)
```

```{r}
hist(df_SELIC$SELIC, main = "Histograma da taxa selic")
```

```{r}
#histograma com frequências relativas no lugar de absolutas
histogram(df_SELIC$SELIC, main = "Histograma da taxa selic")
```

```{r}
# Plotar o gráfico
ggplot(data = df_SELIC, aes(x = Data, y = SELIC)) +
  geom_line() +
  labs(x = "Data", y = "SELIC") +
  ggtitle("Gráfico da Taxa SELIC")
```

Os gráficos acima mostram valores bastante discrepantes no período pré 1994 e pós esta data, sendo consideravelmente maiores no período pré. além disso, é nitida uma descontinuidade na série nesta mesma data. Ambos os fatos se devem ao "Plano Real", que consistiu em uma

mudança de moeda, bem como um novo arcabouço macroeconômico com o objetio - bem sucedido - de estabilizar a inflação massiva que existia até então.

Criaremos uma segunda vizualização apenas com os dados pós- primeiro de julho de 1994 que foi a data de implantação da nova moeda brasileira que estabilizou a inflação:

```{r}
# Filtrar os dados a partir de 1º de julho de 1994
df_SELIC_filtrados <- df_SELIC[df_SELIC$Data >= as.Date("1994-07-01"), ]
```

```{r}
hist(df_SELIC_filtrados$SELIC, main = "Histograma da taxa selic pós Julho 1994")
```

```{r}
#histograma com frequências relativas no lugar de absolutas
histogram(df_SELIC_filtrados$SELIC, main = "Histograma da taxa selic pós Julho 1994")
```

```{r}
# Plotar o gráfico
ggplot(data = df_SELIC_filtrados, aes(x = Data, y = SELIC)) +
  geom_line() +
  labs(x = "Data", y = "SELIC") +
  ggtitle("Gráfico da Taxa SELIC pós 1994")
```

Nesta visualização, os valores de taxa de juros são consistentes com o que se espera de uma econômia de alta inflação com tendência a estabilização: partimos de taxas altas, como 40%, mas que foram sendo paulatinamente reduzidas até chegarem a valores entre 0 e 20% dependendo do momento em que foram auferidas.

Devido aos efeitos do plano real, seguiremos as análises da taxa de juros nessa seção considerando apenas aquelas pós-1994.

É bastante conhecido que séries de taxas de juros são ciclicas, desejamos então aplicar uma análise espectral para identificar sazonalidade, no entanto, para isso, precisamos identificar o regime da série, se estocástico ou determinístico e se estacionaria ou não, a fim de definir nossos próximos passos. [Macroeconomia, Mankiw][(Toloi & Morettin, 2018)]

```{r}
# Teste de estacionariedade - Teste de Dickey-Fuller aumentado (ADF)
adf_result <- adf.test(df_SELIC_filtrados$SELIC)
cat("Resultado do teste de estacionariedade (ADF):\n")
print(adf_result)
```

```{r}
# Converta a matriz de recorrência em um formato adequado para o ggplot2
recurrence_data <- recurrence_plot(df_SELIC_filtrados$SELIC, dimensao=11, eps=0.16)

# Crie o gráfico de heatmap
ggplot(recurrence_data, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "black") +
  theme_minimal()
rm(recurrence_data)
```

```{r}
```

```
# Análise de Autocorrelação
acf(df_SELIC_filtrados$SELIC)
```

```{r}
#Teste de estocasticidade:
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(df_SELIC_filtrados$SELIC)
# Obter os resíduos do modelo
residuos <- residuals(modelo)

# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)

# Verificar tamanho dos resíduos
n <- length(residuos)

# Definir tamanho da amostra
tam_amostra <- ifelse(n > 5000, 5000, n)

# Realizar amostragem
amostra <- sample(residuos, tam_amostra)

# Aplicar o teste de Shapiro-Wilk
resultado_teste <- shapiro.test(amostra)
```

```
# Imprimir resultado
print(resultado_teste)
```

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(df_SELIC_filtrados$SELIC)

#consideraremos 20 defasagens, pois a partir disso o gráfico de autocorrelação se estabiliza
k=20

# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)

# Obter os resíduos do modelo
residuos <- residuals(modelo)

# Realizar o teste de Box-Pierce
bp_test <- Box.test(residuos, lag = k, type = "Ljung-Box")

# Realizar o teste de Ljung-Box
lb_test <- Box.test(residuos, lag = k, type = "Box-Pierce")

#Realizar o Runs Test:
runs_test_result <- runs.test(Binarize_Factorize(df_SELIC_filtrados$SELIC))

# Exibir os resultados dos testes
print(bp_test)
print(lb_test)
print(runs_test_result)
```

Observamos que existe uma autocorrelação significativa na série das taxas SELIC. O Gráfico de recorrência mostra uma linha ortogonal à diagonal principal, bem como regiões homogeneas ao centro e no canto superior direito, o que sugere que os estados evoluem de forma similar em periodos distintos, podendo ser deterministico, mas com clusters homogeneos ao centro e ao canto superior direito, indicando que estes estados são estacionários ou mudam de forma lenta.

O teste de Shapiro-Wilk rejeitou a hipótese nula, indicando que a série de resíduos não é normal (consistente com os resultados do histograma), além disso os testes de Ljung-Box e Box-Pierce ambos rejeitaram a hipótese nula de aleatoriedade.Assim como o Teste de Corridas (Runs Test). Todos estes pontos indicam um componente determinístico na série.

```{r}

peridograma <- spec.pgram(df_SELIC_filtrados$SELIC)

plot(peridograma, main = "Peridograma da Série Temporal", xlab = "Frequência (ciclos/dia)", ylab = "Densidade Espectral",xlim=c(0, 0.1))

```

Observamos que a densidade espectral é maior nas frequências mais baixas. consistente com periodicidades mais longas (maiores do que 0.02 - 1 ano), características de ciclos econômicos, aos quais a taxa de juros está intimamente relacionada. Outra possibilidade é a existência de tendências de longo prazo, no entanto, como o teste aumentado de dick-fulley rejeitou a existência de raizes unitárias, descartaremos essa hipótese. Existe também uma outra hipótese que é a presença de mudanças estruturais nas condições econômicas do Brasil. Neste caso, a análise de change-points pode auxiliar a detectar as mesmas.

```{r}

# Calcula o peridograma

peridograma <- spec.pgram(df_SELIC_filtrados$SELIC)

# Calcula a frequência correspondente ao período

frequencia <- peridograma$freq

# Calcula o período (em dias)
```

```
periodo <- 1 / frequencia

# Plota o peridograma com o eixo x em período
plot(periodo, peridograma$spec, type = "l", main = "Peridograma da Série Temporal",
    xlab = "Período (dias)", ylab = "Densidade Espectral", xlim = c(0, 8000))
```

Analise de sazonalidade:

```{r}
# Testar sazonalidade anual (365 dias)
data<- df_SELIC_filtrados[, c("Data", "SELIC")]

annual_result <- apply_friedman_test(data, 347,"SELIC")
print("Resultado do teste de Friedman para sazonalidade anual:")
print(annual_result)

# Testar sazonalidade de 5 anos (1825 dias)
data<- df_SELIC_filtrados[, c("Data", "SELIC")]
five_year_result <- apply_friedman_test(data, 1041,"SELIC")
print("Resultado do teste de Friedman para sazonalidade de 5 anos:")
print(five_year_result)

# Testar sazonalidade longa  (2430 dias)
data<- df_SELIC_filtrados[, c("Data", "SELIC")]
ten_year_result <- apply_friedman_test(data, 3300,"SELIC")
print("Resultado do teste de Friedman para sazonalidade de 10 anos:")
print(ten_year_result)
```

Análise do preço do Ouro

O Ouro é históricamente considerado como um porto seguro financeiro, no caso de guerras, insurreições e outras calamidades públicas que podem destruir valor de ativos. AInda hoje, diversos países, como Portugal, detém um valor expressivo de suas reservas internacionais em ouro físico. [https://www.gold.org]

```{r}
# Ler o arquivo CSV de preços do Ouro
df_Ouro <- read.csv("Ouro USD.csv", sep = ";")
# Converter a coluna da série temporal para o formato de data
df_Ouro$Data <- as.Date(df_Ouro$Data, format = "%d/%m/%Y")
# Converter a coluna de Valor para número   real
df_Ouro$Ouro_USD <- as.numeric(gsub(",", "", df_Ouro$Ouro_USD))
```

```{r}
hist(df_Ouro$Ouro_USD, main = "Histograma do preço do Ouro")
```

```{r}
#histograma com frequências relativas no lugar de absolutas
histogram(df_Ouro$Ouro_USD, main = "Histograma do preço do Ouro")
```

```{r}
# Plotar o gráfico
ggplot(data = df_Ouro, aes(x = Data, y = Ouro_USD)) +
  geom_line() +
  labs(x = "Data", y = "Ouro_USD") +
  ggtitle("Gráfico do Preço do Ouro em USD")
```

O gráfico do ouro apresenta valores mais suaves do que o taxa de juros, mas apresentam-se regimes distintos nos anos 1980 (quando da crise do petróleo e aumento significativo da

inflação a nível global) e 2000 (período de maior afrouxamento quantitativo) em comparação aos anos 1990, nos quais os preços do ouro se comportaram em relativa estabilidade.

OS dados acima, em uma análise visual, são claramente não-estacionários, no entanto, é comum que séries financeiras sejam estacionárias após sua primeira diferenciação, aplicaremos este processo e depois o teste aumentado de dick-fulley na série.

```{r}
# Realizar a primeira diferenciação
df_Ouro$diferenciada <- c(NA, diff(df_Ouro$Ouro_USD))

# Aplicar o teste aumentado de Dickey-Fuller
adf_result <- adf.test(na.omit(df_Ouro)$diferenciada)
cat("Resultado do teste de estacionariedade (ADF):\n")
print(adf_result)
```

```{r}
# Plotar a série diferenciada versus a data
ggplot(na.omit(df_Ouro), aes(x = Data, y = diferenciada)) +
  geom_line() +
  labs(x = "Data", y = "Série do Ouro diferenciada")
```

A série do ouro diferenciada apresenta maiores valores absolutos próximos ao período de 1980 Até aproximadamente 1984, ficando - em termos relativos mais baixos do final da década de 1980 até o final dos anos 2000. O período de 1980 é consistente com crises econômicas mundiais devido ao choque do Petróleo, crise da dívida latino-americana; já nos anos 2000-2010 observamos a crise do suprime (2008) e uma década dominada por juros baixos e afrouxamento quantitativo nas principais economias. Isto sugere um padrão na série histórica do ouro que possivelmente é modelável. Faremos uma análise de autocorrelação na sequência como parte de uma investigação do caráter (se estocástica, determinística ou aleatória) da série. [Macroeconomia, Mankiw][Toloi e Moretin]

```{r}
```

```
# Análise de Autocorrelação do preço do ouro diferenciado
acf(na.omit(df_Ouro)$diferenciada, main="Autocorrelação do preço do ouro diferenciado")
```

Recurrence plot do preço do ouro

```{r}
# Converta a matriz de recorrência em um formato adequado para o ggplot2
recurrence_data <- recurrence_plot(df_Ouro$Ouro_USD, dimensao=11, eps=0.16)

# Crie o gráfico de heatmap
ggplot(recurrence_data, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "black") +
  theme_minimal()
rm(recurrence_data)
```

Recurrence plot do preço do ouro diferenciado

```{r}
# Converta a matriz de recorrência em um formato adequado para o ggplot2
recurrence_data <- recurrence_plot(na.omit(df_Ouro)$diferenciada, dimensao=11, eps=0.16)

# Crie o gráfico de heatmap
ggplot(recurrence_data, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "black") +
  theme_minimal()
rm(recurrence_data)
```

```{r}
```

```
# Análise de Autocorrelação do preço do ouro

acf(df_Ouro$Ouro_USD)
```

Observamos um padrão de relevância de todos os lags mostrados no gráfico de autocorrelação da série, o que não ocorre no gráfico de autocorrelação da série diferenciada, aonde apenas o lag zero é relevante. O gráfico de recorrência da série mostra um cluster central com algumas faixas brancas, indicando descontinuidades ou ocorrências de transições, já o da série diferenciada apresenta varias linhas verticais e horizontais, indicando que nesta série, alguns estados mudam de forma lenta por muito tempo. Isso posto, faremos os testes de aleatoriedade a fim de identificarmos se a série é determinística ou aleatória:

```{r}
# Ajustar o modelo ARIMA automático à série temporal

modelo <- auto.arima(na.omit(df_Ouro)$diferenciada)

# Obter os parâmetros p, q e d do modelo ajustado

# Obter os parâmetros p, q e d do modelo ajustado

summary(modelo)



#consideraremos 1 defasagens, pois é a única relevante

k=1

# Obter os resíduos do modelo

residuos <- residuals(modelo)

# Realizar o teste de Box-Pierce

bp_test <- Box.test(residuos, lag = k, type = "Ljung-Box")

# Realizar o teste de Ljung-Box

lb_test <- Box.test(residuos, lag = k, type = "Box-Pierce")
```

```
#Realizar o Runs Test:
runs_test_result <- runs.test(Binarize_Factorize(na.omit(df_Ouro)$diferenciada))

# Exibir os resultados dos testes
print(bp_test)
print(lb_test)
print(runs_test_result)
```

```{r}
#Teste de estocasticidade:
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(df_Ouro$diferenciada)
# Obter os resíduos do modelo
residuos <- residuals(modelo)


# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)


# Verificar tamanho dos resíduos
n <- length(residuos)

# Definir tamanho da amostra
tam_amostra <- ifelse(n > 5000, 5000, n)

# Realizar amostragem
amostra <- sample(residuos, tam_amostra)

# Aplicar o teste de Shapiro-Wilk
```

resultado_teste <- shapiro.test(amostra)

Imprimir resultado

print(resultado_teste)

```

Tanto os testes de Box-Ljung como o de Box-Pierce não rejeitaram a hipótese nula. Indicando que temos dados aleatórios. Ao mesmo tempo, o teste de shapiro wilk rejeitou a hipótese de estocasticidade dos dados. Ambos os dados sugerem que a série de preços do ouro apresenta uma componente de ruído muito grande para que qualquer modelagem tenha resultados satisfatórios. No entanto o preço do ouro parece sofrer grande ocorrência de descontinuidades, sendo portanto um campo interessante para análise de change-points, além do uso de técnicas multivariadas.

## ANálise do preço do fundo negociado em Bolsa EWZ

O EWZ é um índice de ações contendo apenas empresas negociadas na bolsa de valores de são Paulo (B3) e representa grosso modo uma exposição ao setor produtivo da econômia brasileira. Além do índice existe também um fundo negociado em bolsa (ETF em inglês), listado na bolsa de Nova York que detém uma carteira de ações igual ao índice, fornecendo aos investidores uma forma de ter exposição ao mesmo. Os dados aqui análisados consistem na cotação de fechamento diária deste fundo.

https://www.ishares.com/us/products/239612/ishares-msci-brazil-capped-etf

```{r}

# Ler o arquivo CSV de preço do EWZ

df_EWZ <- read.csv("EWZ.csv", sep = ",")

# Converter a coluna da série temporal para o formato de data

df_EWZ$Data <- as.Date(df_EWZ$Data, format = "%Y-%m-%d")

# Converter a coluna de Valor para número real

df_EWZ$Close <- as.numeric(gsub(",", ".", df_EWZ$Close))

```

```{r}

```
hist(df_EWZ$Close, main = "Histograma da cotação do fundo EWZ")
```

```{r}
#histograma com frequências relativas no lugar de absolutas
histogram(df_EWZ$Close, main = "Histograma da cotação do fundo EWZ")
```

```{r}
# Plotar o gráfico
ggplot(data = df_EWZ, aes(x = Data, y = Close)) +
  geom_line() +
  labs(x = "Data", y = "Cotação") +
  ggtitle("Gráfico da cotação do índice EWZ")
```

```{r}
# Realizar a primeira diferenciação
df_EWZ$diferenciada <- c(NA, diff(df_EWZ$Close))
#plotar o gráfico
ggplot(data = na.omit(df_EWZ), aes(x = Data, y = diferenciada)) +
  geom_line() +
  labs(x = "Data", y = "Cotação") +
  ggtitle("Gráfico da cotação do índice EWZ diferenciado")
```

O histograma revela uma cauda longa a direita e o gráfico de preço mostra dois momentos bastante distintos: um período de alta desde 2000 até aproximadamente 2008 (período do "Boom das commodities") queda abrputa e recuperação parcial entre 2008 e 2010 (crise do subprime em 2008) e um segundo período de queda.

O gráfico diferenciado revela um momento de maior variação do indice no período da crise do subprime, um período de menor variação na primeira decada do século XXI e, após 2010, um terceiro período de variação aproximadamente constante.

Seguiremos com uma análise de estacionariedade desta série:

```{r}

# Aplicar o teste aumentado de Dickey-Fuller na série original

adf_result <- adf.test(na.omit(df_EWZ)$Close)

cat("Resultado do teste de estacionariedade - Série original(ADF):\n")

print(adf_result)

# Aplicar o teste aumentado de Dickey-Fuller

adf_result <- adf.test(na.omit(df_EWZ)$diferenciada)

cat("Resultado do teste de estacionariedade - Série diferenciada(ADF):\n")

print(adf_result)
```

O teste de dick-fulley indica que a série original é não estacionária, mas que a série diferenciada é estacionária. Vamos analisar a autocorrelação destas duas séries:

```{r}

# Análise de Autocorrelação da cotação do EWZ

acf(df_EWZ$Close, main="Autocorrelação da cotação do EWZ")
```

```{r}

# Análise de Autocorrelação da cotação do EWZ diferenciado

acf(na.omit(df_EWZ)$diferenciada, main="Autocorrelação da cotação do EWZ diferenciado")
```

Observamos um padrão similar ao reportado para o preço do ouro: relevância de todos os lags mostrados no gráfico de autocorrelação da série, o que não ocorre no gráfico de autocorrelação da série diferenciada, aonde apenas o lag zero é relevante. Seguiremos com uma análise de recorrência.

Recurrence plot do preço do EWZ

```{r}
# Converta a matriz de recorrência em um formato adequado para o ggplot2
recurrence_data <- recurrence_plot(na.omit(df_EWZ$Close), dimensao=11, eps=0.16)

# Crie o gráfico de heatmap
ggplot(recurrence_data, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "black") +
  theme_minimal()
rm(recurrence_data)
```

Recurrence plot do preço do EWZ diferenciado

```{r}
# Converta a matriz de recorrência em um formato adequado para o ggplot2
recurrence_data <- recurrence_plot(na.omit(df_EWZ)$diferenciada, dimensao=11, eps=0.16)

# Crie o gráfico de heatmap
ggplot(recurrence_data, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "black") +
  theme_minimal()
rm(recurrence_data)
```

Temos gráficos de recorrência para o EWZ bastante similares aos dos observados para o ouro: O gráfico da série mostra um cluster grande na centro-direita superior com algumas faixas brancas, indicando descontinuidades ou ocorrências de transições, já o da série diferenciada

apresenta uma linha vertical e outra horizontal branca claramente destacada, indicando que existe um estado único que não se repetiu, isso indica a possibilidade de ocorrência de um changepoint.

Dado estes pontos, faremos os testes de aleatoriedade a fim de identificarmos se a série é determinística ou aleatória:

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(na.omit(df_EWZ)$diferenciada)

# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)

#consideraremos 1 defasagens, pois é a única relevante
k=1

# Obter os resíduos do modelo
residuos <- residuals(modelo)

# Realizar o teste de Box-Pierce
bp_test <- Box.test(residuos, lag = k, type = "Ljung-Box")

# Realizar o teste de Ljung-Box
lb_test <- Box.test(residuos, lag = k, type = "Box-Pierce")

#Realizar o Runs Test:
runs_test_result <- runs.test(Binarize_Factorize(na.omit(df_EWZ)$diferenciada))

# Exibir os resultados dos testes
```

```
print(runs_test_result)
print(bp_test)
print(lb_test)
```

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(df_EWZ$diferenciada)
# Obter os resíduos do modelo
residuos <- residuals(modelo)

# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)

# Verificar tamanho dos resíduos
n <- length(residuos)

# Definir tamanho da amostra
tam_amostra <- ifelse(n > 5000, 5000, n)

# Realizar amostragem
amostra <- sample(residuos, tam_amostra)

# Aplicar o teste de Shapiro-Wilk
resultado_teste <- shapiro.test(amostra)

# Imprimir resultado
print(resultado_teste)
```

```

A exemplo do que ocorreu com a série de preços do ouro. Tanto os testes de Box-Ljung como o de Box-Pierce rejeitaram a hipótese nula. Indicando que temos dados aleatórios. Ao mesmo tempo, o teste de shapiro wilk rejeitou a hipótese de estocasticidade dos dados.

Ambos os dados sugerem que a cotação do EWZ apresenta uma componente de ruído muito grande para que qualquer modelagem tenha resultados satisfatórios. No entanto, existe a presença de um estado único, provavelmente causado por um changepoint, razão pela qual pode ser interessante investigarmos o mesmo. Além disso, o gráfico em função do tempo revela padrões na série que possivelmente possam ser modelados em futuras análises multivariadas.

## ANálise do preço da criptomoeda bitcoin

```{r}
# Ler o arquivo CSV de taxas de juros
df_BTC <- read.csv("BTC-USD.csv", sep = ",")
# Converter a coluna da série temporal para o formato de data
df_BTC$Data <- as.Date(df_BTC$Data, format = "%Y-%m-%d")
# Converter a coluna de Valor para número real
df_BTC$Close <- as.numeric(gsub(",", ".", df_BTC$Close))
```

```{r}
hist(df_BTC$Close, main = "Histograma da cotação do bitcoin")
```

```{r}
#histograma com frequências relativas no lugar de absolutas
histogram(df_BTC$Close, main = "Histograma da cotação do bitcoin")
```
```

```

```{r}
# Plotar o gráfico
ggplot(data = df_BTC, aes(x = Data, y = Close)) +
  geom_line() +
  labs(x = "Data", y = "Cotação") +
  ggtitle("Gráfico da cotação do bitcoin")
```

O gráfico do bitcoin revela alguns momentos distintos, primeiro desde o início até 2017 quando o preço era estável e próximo a zero. Houve um pico em 2018, com estabilização em novo patamar de 2018 até o final de 2020.Entre 2021 e 2022 a cotação atingiu novas máximas recordes, posteriormente caindo a um novo patamar médio.

```{r}
# Realizar a primeira diferenciação
df_BTC$diferenciada <- c(NA, diff(df_BTC$Close))
#plotar o gráfico
ggplot(data = na.omit(df_BTC), aes(x = Data, y = diferenciada)) +
  geom_line() +
  labs(x = "Data", y = "Cotação") +
  ggtitle("Gráfico da cotação do índice BTC diferenciado")
```

O gráfico do BTCUSD diferenciado indica as maiores variações ao redor do pico de 2018 e na sequência nas novas máximas de 2021-22. Na sequência, em 2023 os preços parecem estabilizar.

```{r}
# Aplicar o teste aumentado de Dickey-Fuller na série original
adf_result <- adf.test(na.omit(df_BTC)$Close)
cat("Resultado do teste de estacionariedade - Série original(ADF):\n")
```
```

```
print(adf_result)

# Aplicar o teste aumentado de Dickey-Fuller
adf_result <- adf.test(na.omit(df_BTC)$diferenciada)
cat("Resultado do teste de estacionariedade - Série diferenciada(ADF):\n")
print(adf_result)
```

```{r}
# Análise de Autocorrelação da cotação do BTC
acf(df_BTC$Close, main="Autocorrelação da cotação do BTC")
```

```{r}
# Análise de Autocorrelação da cotação do EWZ diferenciado
acf(na.omit(df_BTC)$diferenciada, main="Autocorrelação da cotação do BTC diferenciado")
```

Recurrence plot do preço do BTC

```{r}
# Converta a matriz de recorrência em um formato adequado para o ggplot2
recurrence_data <- recurrence_plot(na.omit(df_BTC$Close), dimensao=11, eps=0.16)

# Crie o gráfico de heatmap
ggplot(recurrence_data, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "black") +
  theme_minimal()
rm(recurrence_data)
```

Recurrence plot do BTC diferenciado

```{r}
# Converta a matriz de recorrência em um formato adequado para o ggplot2
recurrence_data <- recurrence_plot(na.omit(df_BTC)$diferenciada, dimensao=11, eps=0.16)

# Crie o gráfico de heatmap
ggplot(recurrence_data, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "black") +
  theme_minimal()
rm(recurrence_data)
```

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(na.omit(df_BTC)$diferenciada)

# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)

#consideraremos 1 defasagens, pois é a única relevante
k=1

# Obter os resíduos do modelo
residuos <- residuals(modelo)
```

```
# Realizar o teste de Box-Pierce
bp_test <- Box.test(residuos, lag = k, type = "Ljung-Box")

# Realizar o teste de Ljung-Box
lb_test <- Box.test(residuos, lag = k, type = "Box-Pierce")

#Realizar o Runs Test:
runs_test_result <- runs.test(Binarize_Factorize(na.omit(df_BTC)$diferenciada))



# Exibir os resultados dos testes
print(bp_test)
print(lb_test)
print(runs_test_result)
```

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(df_BTC$diferenciada)
# Obter os resíduos do modelo
residuos <- residuals(modelo)



# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)



# Verificar tamanho dos resíduos
n <- length(residuos)

# Definir tamanho da amostra
```

```
tam_amostra <- ifelse(n > 5000, 5000, n)

# Realizar amostragem
amostra <- sample(residuos, tam_amostra)

# Aplicar o teste de Shapiro-Wilk
resultado_teste <- shapiro.test(amostra)

# Imprimir resultado
print(resultado_teste)
```

```

Novamente, temos falha em rejeitar as hipóteses nulas dos testes de box-Ljung e Box-Pierce, além de rejeição da hipótese nula do teste de Shapiro-Wilk. Além disso, o auto-arima retornou como parâmetros ótimos (0,0,0), o que basicamente significa que o termo de ruído é o único relevante. Estes pontos inicialmente sugerem  que temos uma série aleatória.

Os gráficos de recorrência para o BTC, por outro lado adicionam uma informação relevante: O gráfico  da série mostra linhas negras diagonais sugerindo estados que mudam de forma similar. já o da série diferenciada apresenta  linhas vertical e  horizontais em clusters, indicando que existem estados que mudam lentamente, isso sugere que a série do Bitcoin, é governada por etapas relativamente homogêneas mas com transições abruptas entre elas, sugerindo que embora a série como um todo seja caótica, partes individuais da mesma talvez não o sejam.

## Análise de volatilidade

Neste trabalho estamos fundamentalmente interessados no comportamento da volatilidade de ativos financeiros, se podemos detectar pontos de mudança e também modelá-los. O conceito de volatilidade é bastante variado dentro da literatura financeira. em nosso caso entenderemos volatilidade como um desvio padrão rolante, de 21 dias, estes valores, no entanto, serão expostos na forma anualizada.
```

para obter a volatilidade, calcularemos inicialmente o preço relativo, isto é, seja S_t o preço no instante atual, S_{t-1} o preço no instante interior, obteremos:

$$\mu_t=ln(\frac{S_t}{S_{t-1}})$$

e a volatilidade σ será dada por:

$$\sigma=\sqrt{\frac{1}{n-1}\Sigma^n_{t=1}(\mu_t-\bar{\mu})}$$

Para cada uma das séries temporais, calcularemos a volatilidade rolante segundo o seguinte código:

```{r}
# Função division para calcular a divisão entre elementos consecutivos de um vetor
division <- function(x) c(NA, tail(x,n=-1) / head(x,n=-1))

# Função para calcular a volatilidade usando desvio padrão rolante de 21 dias
calcular_volatilidade <- function(dados, d=21) {
  # Calcula o logaritmo natural dos retornos calculados pela função division
  retornos <- log(division(dados))

  # Calcula o desvio padrão rolante de 21 dias dos retornos
  desvio_padrao_rolante <- rollapply(retornos, width = d, FUN = sd, align = "right", fill = NA)

  # Retorna a volatilidade
  return(desvio_padrao_rolante)
}

```

#volatilidade SELIC

Seguiremos na sequência com o cálculo da volatilidade da taxa SELIC e visualizaremos apenas os dados pós outubro de 1994, a fim de eliminar os efeitos de hiperinflação.

```{r}
df_SELIC$volatilidade=calcular_volatilidade(df_SELIC$SELIC)
```

```{r}
#filrar selic a partir de julho de 1994
df_SELIC_filtrados <- df_SELIC[df_SELIC$Data >= as.Date("1994-07-01"), ]
# Plotar o gráfico
ggplot(data = df_SELIC_filtrados, aes(x = Data, y = volatilidade)) +
  geom_line() +
  labs(x = "Data", y = "SELIC") +
  ggtitle("Gráfico da volatilidade da Taxa SELIC pós  julho de 1994")
```

```{r}
#filrar selic a partir de Outubro de 1994
df_SELIC_filtrados <- df_SELIC[df_SELIC$Data >= as.Date("1994-10-01"), ]
# Plotar o gráfico
ggplot(data = df_SELIC_filtrados, aes(x = Data, y = volatilidade)) +
  geom_line() +
  labs(x = "Data", y = "SELIC") +
  ggtitle("Gráfico da volatilidade da Taxa SELIC pós Outubro de 1994")
```

Podemos observar que a volatilidade da taxa SELIC se apresenta bastante elevada nos anos finais da década de 90, tornando-se menor nas duas primeiras décadas do século XXI e voltando a subir nos anos finais da década de 2010 e inicias de 2020 (mas com picos menores do que os dos anos 90). Seguiremos com a análise de aleatoriedade de box pierce e ljung box, junto com os grafícos de autocorrelação e teste de dickfulley a fim de avaliarmos o comportamento dos ativos.

```{r}
# Aplicar o teste aumentado de Dickey-Fuller na série original
adf_result <- adf.test(na.omit(df_SELIC_filtrados)$volatilidade)
cat("Resultado do teste de estacionariedade - Série original(ADF):\n")
print(adf_result)
```

```{r}
# Análise de Autocorrelação da cotação do EWZ diferenciado
acf(na.omit(df_SELIC_filtrados)$volatilidade, main="Autocorrelação da volatilidade da SELIC")
```

```{r}
# Converta a matriz de recorrência em um formato adequado para o ggplot2
recurrence_data <- recurrence_plot(na.omit(df_SELIC_filtrados)$volatilidade, dimensao=11, eps=0.16)

# Crie o gráfico de heatmap
ggplot(recurrence_data, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "black") +
  theme_minimal()
rm(recurrence_data)
```

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(df_SELIC_filtrados$volatilidade)
```

```
# Obter os resíduos do modelo
residuos <- residuals(modelo)


# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)


# Verificar tamanho dos resíduos
n <- length(residuos)

# Definir tamanho da amostra
tam_amostra <- ifelse(n > 5000, 5000, n)

# Realizar amostragem
amostra <- sample(residuos, tam_amostra)

# Aplicar o teste de Shapiro-Wilk
resultado_teste <- shapiro.test(amostra)

# Imprimir resultado
print(resultado_teste)

```

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(na.omit(df_SELIC_filtrados)$volatilidade)

#consideraremos 10 defasagens
k=10
```

```
# Obter os resíduos do modelo

residuos <- residuals(modelo)


# Obter os parâmetros p, q e d do modelo ajustado

summary(modelo)


# Realizar o teste de Box-Pierce

bp_test <- Box.test(residuos, lag = k, type = "Ljung-Box")


# Realizar o teste de Ljung-Box

lb_test <- Box.test(residuos, lag = k, type = "Box-Pierce")


#Realizar o Runs Test:

runs_test_result <- runs.test(Binarize_Factorize(na.omit(df_SELIC_filtrados)$volatilidade))



# Exibir os resultados dos testes

print(bp_test)

print(lb_test)

print(runs_test_result)
```

Observamos no teste de autocorrelação que existem lags relevantes por toda a extensão, havendo um decréscimo da relevância do lag 1 ao 20, com posterior crescimento do 20 ao 30, o teste de dick-fulley rejeitou a hipótese nula de raiz unitária e os de box-ljung e box-pierce a de aleatoriedade. Além disso, o gráfico de recorrência mostra a presença de linhas e padrões ocorrendo de forma periodica. Todos esses fatores sugerem a presença de uma componente sazonal, como ocorre na SELIC ou, de change-points que possam alterar o processo gerador da série. Por outro lado, os parametros ideais do auto-arima foram (0,0,0) esse fato indica que um modelo ARIMA provavelmente não será capaz de modelar adequadamente a volatilidade da SELIC.

faremos agora um peridograma da volatilidade:

```{r}
```

```
peridograma <- spec.pgram(na.omit(df_SELIC_filtrados)$volatilidade)

plot(peridograma, main = "Peridograma da volatilidade da SELIC", xlab = "Frequência
(ciclos/dia)", ylab = "Densidade Espectral",xlim=c(0, 0.1))
```

```
Notam-se no peridograma a presença de ciclos de baixa frequência, indicando a necessidade
de tratamento para sazonalidade.
```

```{r}
# Calcula o peridograma

peridograma <- spec.pgram(df_SELIC_filtrados$volatilidade)

# Calcula a frequência correspondente ao período

frequencia <- peridograma$freq

# Calcula o período (em dias)

periodo <- 1 / frequencia

# Plota o peridograma com o eixo x em período

plot(periodo, peridograma$spec, type = "l", main = "Peridograma da Série Temporal",
    xlab = "Período (dias)", ylab = "Densidade Espectral", xlim = c(0, 8000))
```

```
Analise de sazonalidade:
```

```{r}
# Testar sazonalidade anual (365 dias)

data<- df_SELIC_filtrados[, c("Data", "volatilidade")]

annual_result <- apply_friedman_test(data, 400,"volatilidade")

print("Resultado do teste de Friedman para sazonalidade anual:")

print(annual_result)
```

```
# Testar sazonalidade de 5 anos (1825 dias)
data<- df_SELIC_filtrados[, c("Data", "volatilidade")]
five_year_result <- apply_friedman_test(data, 1440,"volatilidade")
print("Resultado do teste de Friedman para sazonalidade de 5 anos:")
print(five_year_result)

# Testar sazonalidade longa  (2430 dias)
data<- df_SELIC_filtrados[, c("Data", "volatilidade",)]
ten_year_result <- apply_friedman_test(data, 3300,"volatilidade")
print("Resultado do teste de Friedman para sazonalidade de 10 anos:")
print(ten_year_result)
```

Volatilidade Ouro

```{r}
df_Ouro$volatilidade=calcular_volatilidade(df_Ouro$Ouro_USD)
```

```{r}
# Plotar o gráfico
ggplot(data = df_Ouro, aes(x = Data, y = volatilidade)) +
  geom_line() +
  labs(x = "Data", y = "SELIC") +
  ggtitle("Gráfico da Volatilidade da cotação do Ouro")
```

```{r}
# Aplicar o teste aumentado de Dickey-Fuller na série original
```

```
adf_result <- adf.test(na.omit(df_Ouro)$volatilidade)
cat("Resultado do teste de estacionariedade - Série original(ADF):\n")
print(adf_result)
```

```{r}
# Análise de Autocorrelação da cotação da volatilidade do ouro
acf(na.omit(df_Ouro)$volatilidade, main="Autocorrelação da volatilidade da cotação do ouro")
```

```{r}
# Converta a matriz de recorrência em um formato adequado para o ggplot2
recurrence_data <- recurrence_plot(na.omit(df_Ouro)$volatilidade, dimensao=11, eps=0.16)

# Crie o gráfico de heatmap
ggplot(recurrence_data, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "black") +
  theme_minimal()
rm(recurrence_data)
```

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(na.omit(df_Ouro)$volatilidade)

#consideraremos 20 defasagens
k=20

# Obter os parâmetros p, q e d do modelo ajustado
```

```
summary(modelo)

# Obter os resíduos do modelo
residuos <- residuals(modelo)

# Realizar o teste de Box-Pierce
bp_test <- Box.test(residuos, lag = k, type = "Ljung-Box")

# Realizar o teste de Ljung-Box
lb_test <- Box.test(residuos, lag = k, type = "Box-Pierce")

#Realizar o Runs Test:
runs_test_result <- runs.test(Binarize_Factorize(na.omit(df_Ouro)$volatilidade))


# Exibir os resultados dos testes
print(bp_test)
print(lb_test)
print(runs_test_result)
```

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(na.omit(df_Ouro)$volatilidade)
# Obter os resíduos do modelo
residuos <- residuals(modelo)


# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)
```

```
# Verificar tamanho dos resíduos

n <- length(residuos)


# Definir tamanho da amostra

tam_amostra <- ifelse(n > 5000, 5000, n)


# Realizar amostragem

amostra <- sample(residuos, tam_amostra)


# Aplicar o teste de Shapiro-Wilk

resultado_teste <- shapiro.test(amostra)


# Imprimir resultado

print(resultado_teste)
```

A série histórica do Ouro é estacionária e não aleatória, mas de residuos de modelo ARIMA não normalmente distribuidos. no entanto, o gráfico de recorrência mostra padrões de linhas horizontais e verticais periódicos, indicando uma possível componente sazonal. faremos um peridograma para avaliar esta possibilidade:

```{r}
peridograma <- spec.pgram(na.omit(df_Ouro)$volatilidade)

plot(peridograma, main = "Peridograma da volatilidade do Ouro", xlab = "Frequência (ciclos/dia)", ylab = "Densidade Espectral",xlim=c(0, 0.1))
```

```{r}
# Calcula o peridograma


```

```
peridograma <- spec.pgram(na.omit(df_Ouro)$volatilidade)

# Calcula a frequência correspondente ao período
frequencia <- peridograma$freq

# Calcula o período (em dias)
periodo <- 1 / frequencia

# Plota o peridograma com o eixo x em período
plot(periodo, peridograma$spec, type = "l", main = "Peridograma da Série Temporal do ouro",
    xlab = "Período (dias)", ylab = "Densidade Espectral", xlim = c(0, 8000))
```

Novamente, são aparentes aqui ciclos de baixa frequência, exigindo o uso de alguma estratégia para tratamento de sazonalidade.

Volatilidade EWZ

```{r}
df_EWZ$volatilidade=calcular_volatilidade(df_EWZ$Close)
```

```{r}
# Plotar o gráfico
ggplot(data = na.omit(df_EWZ), aes(x = Data, y = volatilidade)) +
  geom_line() +
  labs(x = "Data", y = "EWZ") +
  ggtitle("Gráfico da Volatilidade da cotação do EWZ")
```

```

```{r}
# Aplicar o teste aumentado de Dickey-Fuller na série original
adf_result <- adf.test(na.omit(df_EWZ)$volatilidade)
cat("Resultado do teste de estacionariedade - Série original(ADF):\n")
print(adf_result)
```

```{r}
# Análise de Autocorrelação da cotação da volatilidade do ouro
acf(na.omit(df_EWZ)$volatilidade, main="Autocorrelação da volatilidade da cotação do EWZ")
```

```{r}
# Converta a matriz de recorrência em um formato adequado para o ggplot2
recurrence_data <- recurrence_plot(na.omit(df_EWZ)$volatilidade, dimensao=11, eps=0.16)

# Crie o gráfico de heatmap
ggplot(recurrence_data, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "black") +
  theme_minimal()
rm(recurrence_data)
```

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(na.omit(df_EWZ)$volatilidade)
```
```

```
#consideraremos 20 defasagens
k=20

# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)

# Obter os resíduos do modelo
residuos <- residuals(modelo)

# Realizar o teste de Box-Pierce
bp_test <- Box.test(residuos, lag = k, type = "Ljung-Box")

# Realizar o teste de Ljung-Box
lb_test <- Box.test(residuos, lag = k, type = "Box-Pierce")

#Realizar o Runs Test:
runs_test_result <- runs.test(Binarize_Factorize(na.omit(df_EWZ)$volatilidade))


# Exibir os resultados dos testes
print(bp_test)
print(lb_test)
print(runs_test_result)
```


```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(na.omit(df_EWZ)$volatilidade)
# Obter os resíduos do modelo
residuos <- residuals(modelo)
```

```
# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)

# Verificar tamanho dos resíduos
n <- length(residuos)

# Definir tamanho da amostra
tam_amostra <- ifelse(n > 5000, 5000, n)

# Realizar amostragem
amostra <- sample(residuos, tam_amostra)

# Aplicar o teste de Shapiro-Wilk
resultado_teste <- shapiro.test(amostra)

# Imprimir resultado
print(resultado_teste)
```

A Análise da série de volatilidade do EWZ indica uma série estacionária, com estados sem repetição (correspondentes aos dos picos nas crises de 2008 e e na de 2020, após o covid-19), não aleatória, mas de resíduos não normais ao ser analisadas por um ARIMA. Periodicidades não parecem estar envolvidas aqui. Isso sugere que a volatilidade do EWZ é estocástica, mas não linear.

Volatilidade BTC

```{r}
df_BTC$volatilidade=calcular_volatilidade(df_BTC$Close)
```

```

```{r}
# Plotar o gráfico
ggplot(data = na.omit(df_BTC), aes(x = Data, y = volatilidade)) +
  geom_line() +
  labs(x = "Data", y = "BTC") +
  ggtitle("Gráfico da Volatilidade da cotação do BTC")
```

```{r}
# Aplicar o teste aumentado de Dickey-Fuller na série original
adf_result <- adf.test(na.omit(df_BTC)$volatilidade)
cat("Resultado do teste de estacionariedade - Série original(ADF):\n")
print(adf_result)
```

```{r}
# Análise de Autocorrelação da cotação da volatilidade do ouro
acf(na.omit(df_BTC)$volatilidade, main="Autocorrelação da volatilidade da cotação do BTC")
```

```{r}
# Converta a matriz de recorrência em um formato adequado para o ggplot2
recurrence_data <- recurrence_plot(na.omit(df_BTC)$volatilidade, dimensao=11, eps=0.16)

# Crie o gráfico de heatmap
ggplot(recurrence_data, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "black") +
  theme_minimal()
```
```

```
rm(recurrence_data)
```

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(na.omit(df_BTC)$volatilidade)

#consideraremos 20 defasagens
k=20

# Obter os parâmetros p, q e d do modelo ajustado
summary(modelo)

# Obter os resíduos do modelo
residuos <- residuals(modelo)

# Realizar o teste de Box-Pierce
bp_test <- Box.test(residuos, lag = k, type = "Ljung-Box")

# Realizar o teste de Ljung-Box
lb_test <- Box.test(residuos, lag = k, type = "Box-Pierce")

#Realizar o Runs Test:
runs_test_result <- runs.test(Binarize_Factorize(na.omit(df_BTC)$volatilidade))



# Exibir os resultados dos testes
print(bp_test)
```

```
print(lb_test)
print(runs_test_result)
```

```{r}
# Ajustar o modelo ARIMA automático à série temporal
modelo <- auto.arima(na.omit(df_BTC)$volatilidade)
# Obter os resíduos do modelo
residuos <- residuals(modelo)

# Obter os parâmetros p, q e d do modelo ajustado
p <- modelo$arma[1]
q <- modelo$arma[2]
d <- modelo$arma[3]

# Imprimir os valores dos parâmetros do auto arima
print("Parametros do Auto ARIMA")
print(paste("Parâmetro p:", p))
print(paste("Parâmetro q:", q))
print(paste("Parâmetro d:", d))

# Verificar tamanho dos resíduos
n <- length(residuos)

# Definir tamanho da amostra
tam_amostra <- ifelse(n > 5000, 5000, n)

# Realizar amostragem
```

```
amostra <- sample(residuos, tam_amostra)

# Aplicar o teste de Shapiro-Wilk
resultado_teste <- shapiro.test(amostra)

# Imprimir resultado
print(resultado_teste)
```

```
```

A volatilidade do BTC se mostra estacionária, como pode ser observado pelo teste ADF, porém os testes de box-ljung e box-pierce indicam termos uma série aleatória. O gráfico de recorrência, aparece ser bastante homogeneo, exceto por um estado não repetido exibido pelo cruzamento de uma linha horizontal e vertical branca. além disso, os resíduos de um modelo ARIMA são não-normais. todos estes fatos evidenciam que a volatilidade do BTC exibe uma componente de ruído significativa, sendo provavelmente não modelável no universo univariado.

Análise Multivariada

Junção de todos os dataframes, mantendo apenas as datas comuns:

```{r}
#renomear colunas nos dataframes
df_EWZ <- df_EWZ %>%
  rename(EWZ = Close)
df_EWZ <- df_EWZ %>%
  rename(volatilidade_EWZ = volatilidade)
df_EWZ <- df_EWZ %>%
  rename(diferenciada_EWZ = diferenciada)

df_BTC <- df_BTC %>%
```

```
  rename(BTC = Close)

df_BTC <- df_BTC %>%

  rename(volatilidade_BTC = volatilidade)

df_BTC <- df_BTC %>%

  rename(diferenciada_BTC = diferenciada)


df_Ouro <- df_Ouro %>%

  rename(volatilidade_Ouro = volatilidade)

df_Ouro <- df_Ouro %>%

  rename(diferenciada_Ouro = diferenciada)


df_SELIC <- df_SELIC %>%

  rename(volatilidade_SELIC = volatilidade)
```

```{r}

#junção propriamente dita

df <- merge(df_SELIC, df_EWZ[, c("Data", "EWZ","volatilidade_EWZ","diferenciada_EWZ")], by = "Data", all.x = TRUE)

df <- merge(df, df_BTC[, c("Data", "BTC","volatilidade_BTC","diferenciada_BTC")], by = "Data", all.x = TRUE)

df <- merge(df, df_Ouro, by = "Data", all.x = TRUE)

df <- df[complete.cases(df), ]

```

```{r}

# Função para normalizar uma série

normalize <- function(x) {

  (x - min(x)) / (max(x) - min(x))


```

```
}
```

```{r}
# Normalizar as séries, exceto a coluna "Data"
df_normalized <- df %>%
  mutate_if(~ !("Date" %in% class(.x)), normalize)
```

```{r}
# Plotar as séries no gráfico de linhas
ggplot(data = df_normalized, aes(x = Data)) +
  geom_line(aes(y = EWZ, color = "EWZ")) +
  geom_line(aes(y = BTC, color = "BTC")) +
  geom_line(aes(y = Ouro_USD, color = "Ouro_USD")) +
  geom_line(aes(y = SELIC, color = "SELIC")) +
  labs(x = "Data", y = "Valores", color = "Séries") +
  ggtitle("Gráfico de Linhas das Séries Temporais") +
  theme_minimal()

```

Não existe aparência visível de nenhum relacionamento entre as séries, seguiremos com uma análise de cointegração entre as variáveis:

```{r}
# Realizar o teste de Engle-Granger com as colunas selecionadas
resultado_cointegracao <- ca.jo(df_normalized[, c("EWZ", "BTC", "Ouro_USD","SELIC")], type = "eigen", ecdet = "trend")
```

```
# Exibir os resultados do teste

summary(resultado_cointegracao)
```

O teste de Engle-Granger, rejeita a hipótese nula de ausência de cointegração a 99% de significância (estatistica de teste: 45.08, valor critico para 99%: 36.65), mas falha em estabelecer mesmo uma única relação (estatística de teste:18.05 valor crítico para 90%: 23.11).

Observamos pelo autovetor de maior autovalor (o primeiro) que aparentemente o EWZ e o Bitcoin tem alguma relação de cointegração, aplicaremos uma transformação linear multiplicando o autovetor pelas séries e plotando a mesma. Esperamos que o resultado seja uma série aparentemente estacionária, caso existam relações de cointegração

```{r}
serie_composta=df_normalized[,"EWZ"]*1+df_normalized[,"BTC"]*-20.218477973+df_normalized[,"Ouro_USD"]*0.053630538+df_normalized[,"SELIC"]*-3.068467778

plot(df_normalized[,"Data"],serie_composta)
```

```{r}
# Teste de estacionariedade - Teste de Dickey-Fuller aumentado (ADF)

adf_result <- adf.test(serie_composta)

cat("Resultado do teste de estacionariedade (ADF):\n")

print(adf_result)
```

O gráfico acima mostra uma série aparentemente estacionária até o início de 2020 (quando ocorreu a pandemia da covid-19), a partir dai, o resultado é claramente distinto, indicando uma mudança no comportamento das séries, possívelmente um change-point. O Teste aumentado de Dick-Fulley não rejeitou a hipótese nula, indicando que a série não é estacionária, provavelmente por conta da mudança de relação. Vamos verificar agora como as volatilidades se comportam:

```{r}
# Plotar as séries no gráfico de linhas
ggplot(data = df_normalized, aes(x = Data)) +
  geom_line(aes(y = volatilidade_EWZ, color = "volatilidade_EWZ")) +
  geom_line(aes(y = volatilidade_BTC, color = "volatilidade_BTC")) +
  geom_line(aes(y = volatilidade_Ouro, color = "volatilidade_Ouro")) +
  geom_line(aes(y = volatilidade_SELIC, color = "volatilidade_SELIC")) +
  labs(x = "Data", y = "Valores", color = "Séries") +
  ggtitle("Gráfico da volatilidade das séries analisadas") +
  theme_minimal()

```

As séries de volatilidade parecem ter algum comortamento próximo, mas visualmente é dificil de identificá-lo, seguiremos com o teste de Engle-Granger:

```{r}
# Realizar o teste de Engle-Granger com as colunas selecionadas
resultado_cointegracao <- ca.jo(df_normalized[, c("volatilidade_EWZ", "volatilidade_BTC", "volatilidade_Ouro","volatilidade_SELIC")], type = "eigen", ecdet = "trend")

# Exibir os resultados do teste
summary(resultado_cointegracao)
```

```{r}
serie_composta=df_normalized[,"volatilidade_EWZ"]*1+df_normalized[,"volatilidade_BTC"]*5.0704963848+df_normalized[,"volatilidade_Ouro"]*-0.5810324492
+df_normalized[,"volatilidade_SELIC"]*-9.1365092714
plot(df_normalized[,"Data"],serie_composta)
```

```{r}

# Teste de estacionariedade - Teste de Dickey-Fuller aumentado (ADF)

adf_result <- adf.test(serie_composta)

cat("Resultado do teste de estacionariedade (ADF):\n")

print(adf_result)

```

O resultado dos testes de cointegração das variáveis indica um relacionamento muito mais próximo entre as mesmas, graficamente, também vemos uma série com aparência estacionária, exceto pelo período da Pandemia aonde houve um maior desvio, mas com convergência ao final da série. o Teste aumentado de dick-fulley rejeitou a hipótese nula, indicando que as volatilidades são cointegradas entre si.

6.1.2 Modelagem

---

title: "Modelagem V6"

author: "Pedro Guilherme Frade Moro"

date: '2023-11-13'

output:

  pdf_document: default

  html_document: default

```{r setup, include=FALSE, warning=FALSE}

knitr::opts_chunk$set(echo = TRUE)

```

```{r}

library(rugarch)

library(ggplot2)

library(pracma)
```

```
library(dplyr)
library(Rlibeemd)
library(Metrics)
library(knitr)
```

Vector Autorregression

```{r}
library(vars)

efetua_resultados_var <- function(df_input, window, columns) {
  # Inicialize um dataframe para armazenar as previsões
  predictions <- data.frame()
  df<-df_input[,columns]
  # Loop sobre as observações na série temporal
  for(i in seq(window + 1, nrow(df))) {

    # Selecione as observações na janela de treinamento
    train_data <- df[(i-window):(i-1), ]

    # Selecione a ordem do modelo VAR usando o critério AIC
    var_order <- VARselect(train_data, lag.max = 10, type = "const")$selection["AIC(n)"]

    # Ajuste o modelo VAR da ordem selecionada
    var_model <- VAR(train_data, p = var_order, type = "const")

    # Faça a previsão para a próxima observação
    pred <- predict(var_model, n.ahead = 1)$fcst
```

```
  # Extraia as previsões para cada variável
  pred_values=c()
  for(column in colunas_var){
    pred_values <- cbind(pred_values, pred[[column]][1])
    }

  # Adicione as previsões ao dataframe de previsões
  predictions <- rbind(predictions, pred_values)
 }

 # Nomeie as colunas do dataframe de previsões
 colnames(predictions) <- paste(colnames(df), "_predicted", sep = "")

 # Crie um dataframe vazio com window linhas
 #para tornar o dataframe de predições com o mesmo tamanho do original
 df_nan <- data.frame(matrix(ncol = ncol(predictions), nrow = window))
 colnames(df_nan) <- colnames(predictions)

 # Preencha o dataframe com NA
 df_nan[,] <- NA

 # Adicione as linhas ao início do dataframe 'predictions'
 predictions <- rbind(df_nan, predictions)
 # Concatene o dataframe original com as previsões
 df <- cbind(df, predictions)
 df$Data <-  df_input$Data
 # Retorne o dataframe com as previsões
 return(df)
}
```

```

GARCH

```{r}
library(fGarch)

# Função para realizar predições rolantes de volatilidade com GARCH
garch_rolling_prediction <- function(data, series_col = "serie", volatility_col = "volatilidade", train_window = 20, max_pq = 5) {
  # Criar um dataframe de trabalho
  df <- data.frame(data)

  # Número total de observações
  n <- nrow(df)

  # Inicializar vetor para armazenar as predições de volatilidade
  volatility_predictions <- rep(NA, n)

  for (i in seq(train_window, n)) {
    # Subconjunto dos dados para a janela de treinamento atual
    train_data <- df[(i - train_window + 1):i, ]

    # Inicializar o BIC para um valor alto
    best_bic <- 1e999
    best_model <- NULL

    # Loop sobre possíveis valores de p e q
    for (j in 1:max_pq) {
```
```

```
    for (k in 0:max_pq) {
      # Tente ajustar o modelo GARCH
      tryCatch({
        p <- j
        q <- k
        formula <- as.formula(paste("~ garch(", p, ",", q, ")", sep = ""))
        model <- garchFit(formula = formula, data = as.vector(train_data[[series_col]]), trace =
FALSE)

        # Verificar se o BIC deste modelo é melhor
        if (model@fit$ics[2] < best_bic) {
          best_bic <- model@fit$ics[2]
          best_model <- model
        }
      }, error = function(e) {
        message(paste("Falha ao ajustar o modelo GARCH(", p, ",", q, "). Prosseguindo para a
próxima combinação.", sep = ""))
      })
    }
  }

  # Tente realizar a predição para o próximo ponto
  tryCatch({
    forecast_result <- predict(best_model, n.ahead = 1)
    # Armazenar a predição de volatilidade
    volatility_predictions[i] <- forecast_result$standardDeviation
  }, error = function(e) {
    message("Falha ao prever a volatilidade. Usando a última volatilidade conhecida.")
    # Se a previsão falhar, use a última volatilidade conhecida
    last_known_volatility <- tail(train_data[[volatility_col]], 1)
    volatility_predictions[i] <- last_known_volatility
  })
```

```
  }

  # Adicionar as predições de volatilidade ao dataframe original
  df$Predictions <- volatility_predictions

  # Retornar o dataframe com as predições de volatilidade
  return(df)
}
```

SARIMAX

```{r}
library(forecast)

# Função para realizar predição rolante com SARIMAX
library(forecast)

# Função para realizar predição rolante com SARIMAX
sarimax_rolling_prediction <- function(data, endog_col, train_window = 20, seasonal_order = c(0, 1, 1, 12), exog_col = NULL, forward_pred=FALSE) {
  # Criar um dataframe de trabalho
  df <- data.frame(data)

  # Inicializar vetores para armazenar as predições e os resíduos
  predictions <- numeric(0)
  residuals <- numeric(0)

  # Número total de observações
  n <- nrow(df)
  #se forward pred, faz predições para a próxima janela desconhecida, se não, faz apenas para as próximas
```

```
 if (forward_pred){
  Indexes=seq(train_window, n)
 }
 else{
  Indexes=seq(train_window, n-1)
 }
 for (i in Indexes) {
  # Subconjunto dos dados para a janela de treinamento atual
  train_data <- df[(i - train_window + 1):i, ]

  # Verificar se a variável de controle está presente
  if (!is.null(exog_col)) {
   # Estimar a ordem do modelo ARIMA usando o critério AIC
   auto_arima_result <- auto.arima(train_data[[endog_col]], xreg = train_data[[exog_col]])

   # Ajustar o modelo SARIMAX com a ordem estimada
   model <- Arima(train_data[[endog_col]], order = auto_arima_result$arma[c(1, 6, 2)],
                  seasonal = seasonal_order, xreg = train_data[[exog_col]])
   # Realizar a predição para o próximo ponto
  forecast_result <- forecast(model, xreg = ifelse(!is.null(exog_col), df[[exog_col]][i],
NULL),h=1)
  } else {
   # Ajustar o modelo SARIMAX sem variável de controle
   model <- auto.arima(train_data[[endog_col]], seasonal = seasonal_order)
    # Realizar a predição para o próximo ponto
  forecast_result <- forecast(model,h=1)
  }



  # Armazenar a predição e o resíduo
```

```
    predictions <- c(predictions, forecast_result$mean)
    residuals <- c(residuals, forecast_result$residuals)
  }

  # Adicionar as predições ao dataframe original
  df$Predictions <- c(rep(NA, train_window), predictions)

  # Retornar o dataframe com as predições e os resíduos
  return(df)
}
```

```

Filtro de Kalman

```{r}
kalman_filter_single <- function(y, Q_init = 0.1, R_init = 1, x_init = 0, P_init = 1) {
  # Verificar se y é uma matriz e, se não for, convertê-la em uma matriz
  if (!is.matrix(y)) {
    y <- matrix(y, nrow = length(y), ncol = 1)
  }

  # Definir matrizes de transição de estado (A) e covariâncias do processo (Q)
  A <- matrix(c(1), nrow = 1, ncol = 1)  # Para uma série temporal univariada, A é simplesmente 1
  Q <- ifelse(is.matrix(Q_init), Q_init, matrix(c(Q_init), nrow = 1, ncol = 1))  # Covariância do ruído do processo
```
```

```
  # Inicializar matrizes de observação (H) e covariâncias do ruído de medição (R)
  H <- matrix(c(1), nrow = 1, ncol = 1)  # H é 1, pois estamos observando a volatilidade diretamente
  R <- ifelse(is.matrix(R_init), R_init, matrix(c(R_init), nrow = 1, ncol = 1))  # Covariância do ruído de medição

  # Inicializar vetor de estado inicial (x) e covariância do estado inicial (P)
  x <- ifelse(is.matrix(x_init), x_init, matrix(c(x_init), nrow = 1, ncol = 1))  # Valor inicial do estado
  P <- ifelse(is.matrix(P_init), P_init, matrix(c(P_init), nrow = 1, ncol = 1))  # Covariância inicial do estado

  # Etapa de previsão
  x_pred <- A %*% x
  P_pred <- A %*% P %*% t(A) + Q

  # Etapa de atualização
  K <- P_pred %*% t(H) %*% solve(H %*% P_pred %*% t(H) + R)
  x <- x_pred + K %*% (y[length(y)] - H %*% x_pred)
  P <- P_pred - K %*% H %*% P_pred

  # Estimar a volatilidade
  y_pred <- H %*% x

  return(list(y_pred = y_pred, Q = Q, R = R, x = x, P = P))
}

kalman_filter_rolling <- function(y, window = length(y), Q_init = 0.1, R_init = 1, x_init = 0, P_init = 1) {
  # Verificar se y é uma matriz e, se não for, convertê-la em uma matriz
  if (!is.matrix(y)) {
    y <- matrix(y, nrow = length(y), ncol = 1)
  }
```

```
  # Inicializar o vetor de previsões
  predictions <- numeric(window)

  # Loop sobre a série temporal
  for (i in (window + 1):length(y)) {
    # Obter a janela atual da série temporal
    y_window <- y[(i - window):(i - 1), , drop = FALSE]

    # Aplicar o Filtro de Kalman à janela atual
    kalman_result <- kalman_filter_single(y_window, Q_init, R_init, x_init, P_init)

    # Adicionar a previsão ao vetor de previsões
    predictions <- c(predictions, kalman_result$y_pred)

    # Atualizar Q, R, x e P para a próxima iteração
    Q_init <- kalman_result$Q
    R_init <- kalman_result$R
    x_init <- kalman_result$x
    P_init <- kalman_result$P
  }

  return(predictions)
}

```

```{r}
# Instale e carregue as bibliotecas necessárias
```

```
library(tseriesChaos)

library(infotheo)

library(rEDM)


simplex_takens <- function(df, column_name, embedding_dimension, time_delay,
window=20) {

  # Extraia a série temporal da coluna especificada

  values <- df[[column_name]]


  # Crie um dataframe para armazenar as previsões

  predictions <- data.frame()


  #validação de window: garante que o tamanho mínimo do dataset respeita as dimensões
necessárias

  window=max(window,embedding_dimension*time_delay)


  # Loop sobre a série temporal

  for (i in window:(length(values))) {

    # Crie um vetor de atraso

    #delay_vector <- embed(values[i:(i + embedding_dimension * time_delay - 1)],
embedding_dimension)


    # Converta o vetor de atraso em um data.frame

   # delay_df <- as.data.frame(delay_vector)


    # Use o Simplex Projection para fazer a previsão

    #simplex_result <- simplex(delay_vector, E = embedding_dimension, tp = time_delay, lib =
c(1, nrow(delay_vector)), pred = c(nrow(delay_vector) + 1, nrow(delay_vector) + 1))


    simplex_result <- simplex(values, lib =c(1, i), pred= c(i+1, i+2),norm = 2,E =
1:embedding_dimension,tau = time_delay, tp = 1)

    #https://ha0ye.github.io/rEDM/reference/simplex.html
```

```
  # Adicione a previsão ao dataframe de previsões
  predictions <- rbind(predictions, simplex_result$Predictions)
 }

 # Adicione a coluna de previsões ao dataframe original
 df$predictions <- predictions

 # Retorne o dataframe com a nova coluna de previsões
 return(df)
}
```

R quadrado:

```{r}
r_squared <- function(volatilidade_teste, volatilidade_pred) {
r_squared <- 1 - sum((volatilidade_teste - volatilidade_pred)^2) / sum((volatilidade_teste - mean(volatilidade_teste))^2)
return(r_squared)
}
```

Modelagem Volatilidade SELIC

```{r}
df=df_SELIC_filtrados
```

```
df <- na.omit(df)
```

predição Sarima

```{r}
resultados_sarimax <- sarimax_rolling_prediction(df, endog_col = "volatilidade", train_window = 20,
                                        seasonal_order = c(0, 1, 1, 12))
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_sarimax)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Predição Takens e Simplex

```{r}
#resultados_takens <- simplex_takens(df, "volatilidade",10,7)
#embeding dimensions=10
#time delay=7
#ambos retirados do código de false nearnest neighbors
```

```{r}
# Criar o gráfico de dispersão
### resultados_plot<- na.omit(resultados_takens)
###  ggplot() +
###    # Adicionar dados reais
###    geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
  # Adicionar previsões
###    geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
  # Adicionar rótulos aos eixos e título
###    labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
  # Personalizar cores e tema do gráfico
###    theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
```

cat("R-Quadrado:", r_quadrado, "\n")

```

```{r}

r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)

cat("MAE:", r_quadrado, "\n")

```

Predição GARCH:

```{r}

#last_100_rows <- tail(df, 1000)

#resultados_garch <- garch_rolling_prediction(last_100_rows, series_col = #"SELIC", volatility_col = "volatilidade", train_window = 40)

```

```{r}

resultados_garch <- garch_rolling_prediction(df, series_col = "SELIC", volatility_col = "volatilidade", train_window = 40)

```

```{r}

# Criar o gráfico de dispersão

resultados_plot<- na.omit(resultados_garch)

 ggplot() +

  # Adicionar dados reais

  geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +

  # Adicionar previsões

  geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +

```
  # Adicionar rótulos aos eixos e título
  labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
  # Personalizar cores e tema do gráfico
  theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Filtro de Kalman

```{r}
#last_100_rows <- tail(df, 100)
#last_100_rows$Predicitions = kalman_filter(last_100_rows$volatilidade)
```

```{r}
resultados_Kalman=df
resultados_Kalman$Predictions = kalman_filter_rolling(df$volatilidade, window =22)
```

```{r}
# Criar o gráfico de dispersão
```

```
resultados_plot<- na.omit(resultados_Kalman)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Modelagem Volatilidade Ouro

```{r}
df=df_Ouro
df <- na.omit(df)
```

predição Sarima

```{r}
resultados_sarimax <- sarimax_rolling_prediction(df, endog_col = "volatilidade_Ouro",
train_window = 20,
                                                 seasonal_order = c(0, 1, 1, 12))
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_sarimax)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade_Ouro), color = "blue", size
= 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "volatilidade_Ouro", title = "Comparação entre Dados Reais e Previsões")
+
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

```

Predição GARCH:

```{r}
resultados_garch <- garch_rolling_prediction(df, series_col = "Ouro_USD", volatility_col = "volatilidade_Ouro", train_window = 20)

```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_garch)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade_Ouro), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
```
```

```
cat("MAE:", r_quadrado, "\n")
```

Filtro de Kalman

```{r}
resultados_Kalman=df
resultados_Kalman$Predictions = kalman_filter(df$volatilidade_Ouro)
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_Kalman)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade_Ouro), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
```

r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)

cat("MAE:", r_quadrado, "\n")

```

# Modelagem Volatilidade BTC

```{r}

df=df_BTC

df <- na.omit(df)

```

predição Sarima

```{r}

resultados_sarimax <- sarimax_rolling_prediction(df, endog_col = "volatilidade_BTC",
train_window = 20,

                                    seasonal_order = c(0, 1, 1, 12))

```

```{r}

# Criar o gráfico de dispersão

resultados_plot<- na.omit(resultados_sarimax)

 ggplot() +

  # Adicionar dados reais

  geom_point(data = resultados_plot, aes(x = Data, y = volatilidade_BTC), color = "blue", size =
2) +

  # Adicionar previsões

  geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +

  # Adicionar rótulos aos eixos e título

  labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +

  # Personalizar cores e tema do gráfico

  theme_minimal()

```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Predição GARCH:

```{r}
resultados_garch <- garch_rolling_prediction(df, series_col = "BTC", volatility_col = "volatilidade_BTC", train_window = 20)
```

```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_garch)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade_BTC), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
```

```
  theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade_BTC,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade_BTC,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Filtro de Kalman

```{r}
resultados_Kalman=df
resultados_Kalman$Predictions = kalman_filter(df$volatilidade_BTC)
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_Kalman)
 ggplot() +
  # Adicionar dados reais
  geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
  # Adicionar previsões
  geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
  # Adicionar rótulos aos eixos e título
  labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
  # Personalizar cores e tema do gráfico
```

```
  theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Modelagem volatilidade EWZ

```{r}
df=df_EWZ
df <- na.omit(df)
```

predição Sarima

```{r}
resultados_sarimax <- sarimax_rolling_prediction(df, endog_col = "volatilidade_EWZ",
train_window = 20,
                                  seasonal_order = c(0, 1, 1, 12))
```

```{r}
# Criar o gráfico de dispersão
```

```
resultados_plot<- na.omit(resultados_sarimax)

 ggplot() +

   # Adicionar dados reais

   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade_EWZ), color = "blue", size
= 2) +

   # Adicionar previsões

   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +

   # Adicionar rótulos aos eixos e título

   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +

   # Personalizar cores e tema do gráfico

   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)

cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)

cat("MAE:", r_quadrado, "\n")
```

Predição GARCH:

```{r}
resultados_garch <- garch_rolling_prediction(df, series_col = "EWZ", volatility_col =
"volatilidade_EWZ", train_window = 100)

```

```{r}
```

```
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_garch)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade_EWZ), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade_EWZ,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade_EWZ,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Filtro de Kalman

```{r}
resultados_Kalman=df
resultados_Kalman$Predictions = kalman_filter(df$volatilidade_EWZ)
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_Kalman)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade_EWZ), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Modelagem Multivariada - com SELIC

```{r}
colunas_var<-c("volatilidade_EWZ", "volatilidade_BTC", "volatilidade_Ouro","volatilidade_SELIC")
resultados_var <-efetua_resultados_var(df_normalized,22,colunas_var)
```

```{r}

for(coluna in colunas_var){

  # Criar o gráfico de dispersão

  resultados_plot<- na.omit(resultados_var)

  p <- ggplot() +

    # Adicionar dados reais

    geom_point(data = resultados_plot, aes_string(x = "Data", y = coluna), color = "blue", size = 2) +

    # Adicionar previsões

    geom_point(data = resultados_plot, aes_string(x = "Data", y = paste0(coluna, "_predicted")), color = "red", size = 2) +

    # Adicionar rótulos aos eixos e título

    labs(x = "Data", y = "Volatilidade", title = paste("Comparação entre Dados Reais e Previsões para ",coluna)) +

    # Personalizar cores e tema do gráfico

    theme_minimal()


  # Imprimir o gráfico

  print(p)

}


```


```{r}

for(coluna in colunas_var){

cat("Resultados para:", coluna, "\n")

r_quadrado <- r_squared(resultados_plot[[coluna]],resultados_plot[[paste0(coluna,"_predicted")]])

cat("R-Quadrado:", r_quadrado, "\n")

r_quadrado <- mae(resultados_plot[[coluna]],resultados_plot[[paste0(coluna, "_predicted")]])

cat("MAE:", r_quadrado, "\n\n")


```

```
}
```

##Modelagem Multivariada - Sem SELIC:

```{r}
colunas_var<-c("volatilidade_EWZ", "volatilidade_BTC", "volatilidade_Ouro")
resultados_var <-efetua_resultados_var(df_normalized,22,colunas_var)
```

```{r}
for(coluna in colunas_var){
  # Criar o gráfico de dispersão
  resultados_plot<- na.omit(resultados_var)
  p <- ggplot() +
    # Adicionar dados reais
    geom_point(data = resultados_plot, aes_string(x = "Data", y = coluna), color = "blue", size = 2) +
    # Adicionar previsões
    geom_point(data = resultados_plot, aes_string(x = "Data", y = paste0(coluna, "_predicted")), color = "red", size = 2) +
    # Adicionar rótulos aos eixos e título
    labs(x = "Data", y = "Volatilidade", title = paste("Comparação entre Dados Reais e Previsões para ",coluna)) +
    # Personalizar cores e tema do gráfico
    theme_minimal()

  # Imprimir o gráfico
  print(p)
}
```

```{r}
for(coluna in colunas_var){
cat("Resultados para:", coluna, "\n")
r_quadrado <-
r_squared(resultados_plot[[coluna]],resultados_plot[[paste0(coluna,"_predicted")]])
cat("R-Quadrado:", r_quadrado, "\n")
r_quadrado <- mae(resultados_plot[[coluna]],resultados_plot[[paste0(coluna, "_predicted")]])
cat("MAE:", r_quadrado, "\n\n")
}
```

Modelagem com EMD

Modelagem Volatilidade SELIC

```{r}
df=df_SELIC_filtrados
df <- na.omit(df)
```

```{r}
# 1 - Decomposição nas respectivas IMFs
imfs <- Rlibeemd::emd(df$volatilidade)
imfs<-data.frame(imfs)
predicted_imfs <- imfs
```

```{r}

# 2 - Aplicar predição Sarima em cada série
suppressWarnings({
for (i in 1:length(imfs)) {
 if(i==length(imfs)){
  coluna="Residual"
 }
 else{
 coluna=paste("IMF", i, sep = ".")
 }
  resultados_sarimax <- sarimax_rolling_prediction(imfs, endog_col = coluna, train_window =
20,
                                        seasonal_order = c(0, 1, 1, 12))
  predicted_imfs[,coluna] <- resultados_sarimax$Predictions
  cat(coluna, "Concluida \n")
  }
})
```

```{r}
# 3 - Recompor todas as predições
predicted_data <- rowSums(predicted_imfs)
resultados_sarimax$Predictions <-predicted_data
resultados_sarimax$volatilidade <- df$volatilidade
resultados_sarimax$Data <- df$Data
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_sarimax)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Filtro de Kalman

```{r}
resultados_Kalman=imfs
# 2 - Aplicar existing_predict_function em cada série
suppressWarnings({
```

```
for (i in 1:length(imfs)) {
  if(i==length(imfs)){
    coluna="Residual"
  }
  else{
  coluna=paste("IMF", i, sep = ".")
  }
    resultados_Kalman$Predictions = kalman_filter(imfs[,coluna])

    predicted_imfs[,coluna] <- resultados_Kalman$Predictions
    cat(coluna, "Concluida \n")
    }
})
```

```{r}
# 3 - Recompor todas as predições
predicted_data <- rowSums(predicted_imfs)
resultados_Kalman$Predictions <-predicted_data
resultados_Kalman$volatilidade <- df$volatilidade
resultados_Kalman$Data <- df$Data
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_Kalman)
  ggplot() +
    # Adicionar dados reais
    geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
    # Adicionar previsões
    geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
```

```
  # Adicionar rótulos aos eixos e título
  labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
  # Personalizar cores e tema do gráfico
  theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Modelagem Volatilidade Ouro

```{r}
df=df_Ouro
df <- na.omit(df)
```

predição Sarima

```{r}
# 1 - Decomposição nas respectivas IMFs
imfs <- Rlibeemd::emd(df$volatilidade_Ouro)
imfs<-data.frame(imfs)
```

```
predicted_imfs <- imfs
```

```{r}

# 2 - Aplicar existing_predict_function em cada série
suppressWarnings({
for (i in 1:length(imfs)) {
 if(i==length(imfs)){
  coluna="Residual"
 }
 else{
 coluna=paste("IMF", i, sep = ".")
 }
  resultados_sarimax <- sarimax_rolling_prediction(imfs, endog_col = coluna, train_window = 20,
                                        seasonal_order = c(0, 1, 1, 12))
  predicted_imfs[,coluna] <- resultados_sarimax$Predictions
  cat(coluna, "Concluida \n")
  }
})
```

```{r}
# 3 - Recompor todas as predições
predicted_data <- rowSums(predicted_imfs)
resultados_sarimax$Predictions <-predicted_data
resultados_sarimax$volatilidade <- df$volatilidade_Ouro
resultados_sarimax$Data <- df$Data
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_sarimax)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Filtro de Kalman

```{r}
```

```
resultados_Kalman=imfs
# 2 - Aplicar existing_predict_function em cada série
suppressWarnings({
for (i in 1:length(imfs)) {
 if(i==length(imfs)){
  coluna="Residual"
 }
 else{
 coluna=paste("IMF", i, sep = ".")
 }
  resultados_Kalman$Predictions = kalman_filter(imfs[,coluna])

  predicted_imfs[,coluna] <- resultados_Kalman$Predictions
  cat(coluna, "Concluida \n")
  }
})
```

```{r}
# 3 - Recompor todas as predições
predicted_data <- rowSums(predicted_imfs)
resultados_Kalman$Predictions <-predicted_data
resultados_Kalman$volatilidade <- df$volatilidade_Ouro
resultados_Kalman$Data <- df$Data
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_Kalman)
 ggplot() +
  # Adicionar dados reais
```

```
  geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
  # Adicionar previsões
  geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
  # Adicionar rótulos aos eixos e título
  labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
  # Personalizar cores e tema do gráfico
  theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Modelagem Volatilidade BTC

```{r}
df=df_BTC
df <- na.omit(df)
```

predição Sarima

```{r}
# 1 - Decomposição nas respectivas IMFs
imfs <- Rlibeemd::emd(df$volatilidade_BTC)
imfs<-data.frame(imfs)
predicted_imfs <- imfs
```

```{r}
# 2 - Aplicar existing_predict_function em cada série
suppressWarnings({
for (i in 1:length(imfs)) {
  if(i==length(imfs)){
    coluna="Residual"
  }
  else{
  coluna=paste("IMF", i, sep = ".")
  }
   resultados_sarimax <- sarimax_rolling_prediction(imfs, endog_col = coluna, train_window = 20,
                                        seasonal_order = c(0, 1, 1, 12))
   predicted_imfs[,coluna] <- resultados_sarimax$Predictions
   cat(coluna, "Concluida \n")
   }
})
```

```{r}
# 3 - Recompor todas as predições
predicted_data <- rowSums(predicted_imfs)
```

```
resultados_sarimax$Predictions <-predicted_data
resultados_sarimax$volatilidade <- df$volatilidade_BTC
resultados_sarimax$Data <- df$Data
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_sarimax)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Filtro de Kalman

```{r}
resultados_Kalman=imfs
# 2 - Aplicar existing_predict_function em cada série
suppressWarnings({
for (i in 1:length(imfs)) {
  if(i==length(imfs)){
    coluna="Residual"
  }
  else{
  coluna=paste("IMF", i, sep = ".")
  }
    resultados_Kalman$Predictions = kalman_filter(imfs[,coluna])

    predicted_imfs[,coluna] <- resultados_Kalman$Predictions
    cat(coluna, "Concluida \n")
    }
})
```

```{r}
# 3 - Recompor todas as predições
predicted_data <- rowSums(predicted_imfs)
resultados_Kalman$Predictions <-predicted_data
resultados_Kalman$volatilidade <- df$volatilidade_BTC
resultados_Kalman$Data <- df$Data
```

```{r}
# Criar o gráfico de dispersão
```

```
resultados_plot<- na.omit(resultados_Kalman)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Modelagem Volatilidade EWZ

```{r}
df=df_EWZ
df <- na.omit(df)
```

predição Sarima

```{r}
# 1 - Decomposição nas respectivas IMFs
imfs <- Rlibeemd::emd(df$volatilidade_EWZ)
imfs<-data.frame(imfs)
predicted_imfs <- imfs
```

```{r}

# 2 - Aplicar existing_predict_function em cada série
suppressWarnings({
for (i in 1:length(imfs)) {
  if(i==length(imfs)){
    coluna="Residual"
  }
  else{
  coluna=paste("IMF", i, sep = ".")
  }
    resultados_sarimax <- sarimax_rolling_prediction(imfs, endog_col = coluna, train_window =
20,
                                          seasonal_order = c(0, 1, 1, 12))
    predicted_imfs[,coluna] <- resultados_sarimax$Predictions
    cat(coluna, "Concluida \n")
    }
})
```

```

```{r}
# 3 - Recompor todas as predições
predicted_data <- rowSums(predicted_imfs)
resultados_sarimax$Predictions <-predicted_data
resultados_sarimax$volatilidade <- df$volatilidade_EWZ
resultados_sarimax$Data <- df$Data
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_sarimax)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```

Filtro de Kalman

```{r}
resultados_Kalman=imfs
# 2 - Aplicar existing_predict_function em cada série
suppressWarnings({
for (i in 1:length(imfs)) {
  if(i==length(imfs)){
    coluna="Residual"
  }
  else{
  coluna=paste("IMF", i, sep = ".")
  }
    resultados_Kalman$Predictions = kalman_filter(imfs[,coluna])

    predicted_imfs[,coluna] <- resultados_Kalman$Predictions
    cat(coluna, "Concluida \n")
    }
})
```

```{r}
# 3 - Recompor todas as predições
predicted_data <- rowSums(predicted_imfs)
resultados_Kalman$Predictions <-predicted_data
```

```
resultados_Kalman$volatilidade <- df$volatilidade_EWZ
resultados_Kalman$Data <- df$Data
```

```{r}
# Criar o gráfico de dispersão
resultados_plot<- na.omit(resultados_Kalman)
 ggplot() +
   # Adicionar dados reais
   geom_point(data = resultados_plot, aes(x = Data, y = volatilidade), color = "blue", size = 2) +
   # Adicionar previsões
   geom_point(data = resultados_plot, aes(x = Data, y = Predictions ), color = "red", size = 2) +
   # Adicionar rótulos aos eixos e título
   labs(x = "Data", y = "Volatilidade", title = "Comparação entre Dados Reais e Previsões") +
   # Personalizar cores e tema do gráfico
   theme_minimal()
```

```{r}
r_quadrado <- r_squared(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("R-Quadrado:", r_quadrado, "\n")
```

```{r}
r_quadrado <- mae(resultados_plot$volatilidade,resultados_plot$Predictions)
cat("MAE:", r_quadrado, "\n")
```