



**UNIVERSIDADE ESTADUAL DA PARAÍBA
CAMPUS 1
CENTRO DE CIÊNCIA E TECNOLOGIA - CCT
DEPARTAMENTO ESTATÍSTICA
CURSO DE ESTATÍSTICA**

CLEVIA BENTO DE OLIVEIRA

**ANÁLISE DE REGRESSÃO E SÉRIES TEMPORAIS APLICADOS NO HISTÓRICO
DE PREÇOS DA CRIPTOMOEDA ETHEREUM**

**CAMPINA GRANDE - PB
2022**

CLEVIA BENTO DE OLIVEIRA

**ANÁLISE DE REGRESSÃO E SÉRIES TEMPORAIS APLICADOS NO HISTÓRICO
DE PREÇOS DA CRIPTOMOEDA ETHEREUM**

Trabalho apresentado a Ricardo Alves de
Olinda na disciplina de Econometria, curso de
Estatística da Universidade Estadual da
Paraíba, para conclusão da disciplina.

**CAMPINA GRANDE
2022**

SUMÁRIO

- 1. INTRODUÇÃO**
- 2. CRIPTOMOEDA ETHEREUM**
- 3. OBJETIVOS DA ANÁLISE**
- 4. METODOLOGIA**
- 5. DESCRIÇÃO DO BANCO DE DADOS**
- 6. ANÁLISE DE REGRESSÃO**
- 7. SÉRIES TEMPORAIS**
- 8. ANÁLISE INICIAL DO BANCO DE DADOS**
- 9. ANÁLISE UTILIZANDO REGRESSÃO SIMPLES**
- 10. ANÁLISE UTILIZANDO SÉRIES TEMPORAIS**
- 11. CONCLUSÃO**
- 12. REFERÊNCIAS**
- 13. ANEXOS**

1. INTRODUÇÃO

Após a quebra do tradicional banco americano, Lehman Brothers, em 2008 e a recusa do governo norte-americano de salvá-lo (ao colocar dinheiro público no Banco que era privado) as bolsas ao redor do mundo entraram em colapso, pois os investidores passaram a resgatar suas aplicações, diminuindo a liquidez no mercado, ou seja, a facilidade e à rapidez em comprar ou vender um ativo financeiro específico a um preço mais ou menos estável diminuiu consideravelmente.

Isso quer dizer que quem tinha dinheiro investido em bancos e em ações pediu para sacá-lo com medo de perdê-lo, e os bancos não tinham como cobrir tantos saques. Nessa época, um conjunto de pessoas fora da área de finanças, quase todos com percurso ligado à tecnologia, começaram a pensar numa forma alternativa de trocar bens e serviços, sem os intermediários tradicionais que são os bancos e as empresas financeiras.

O Bitcoin, criptomoeda mais reconhecida do mundo, foi lançada para o público em 2009 por Satoshi Nakamoto, seu criador. Pouco se sabe sobre a identidade de Nakamoto, porém o Blockchain, tecnologia por ele desenvolvida que de forma resumida é um sistema que permite rastrear o envio e recebimento de alguns tipos de informação pela internet. São pedaços de código gerados online que carregam informações conectadas – como blocos de dados que formam uma corrente, modernizou o sistema por trás das criptomoedas e possibilitou a ampliação de seu uso. O Bitcoin foi a primeira criptomoeda a utilizar a tecnologia do Blockchain, grande responsável pelo seu sucesso por conta da segurança trazida pelo sistema. A partir dela, criptomoedas alternativas surgiram adotando o Blockchain, que hoje é o sistema-base para a maioria das criptomoedas.

1.1 O que é uma criptomoeda?

As criptomoedas são moedas digitais descentralizadas, ou seja, que não são controladas por algum órgão ou país em específico. Elas são criadas em uma rede blockchain que é responsável por armazenar com segurança os mais diversos tipos de informações. Como, por exemplo, transações financeiras, registros e os dados de pessoas que participam dessas transações. Essas criptomoedas geradas no blockchain possuem um valor que, em alguns casos, pode ser convertido para outras moedas, como o dólar ou real, e, por isso, elas podem ser utilizadas como moeda de troca para compra de produtos e consumo de serviços.

Inicialmente as criptomoedas tinham a função de servir como pagamento para mineradores na blockchain. Atualmente, elas podem ser vistas como meio de troca, sendo utilizadas para a compra de produtos e consumo de serviços. Outra grande função das criptomoedas é sua utilização como alternativa de investimentos, já que os investidores começaram a enxergar que elas podem se tornar uma boa reserva de valor.

A partir daí surgem empresas que aceitam criptomoedas como pagamento, fundos de investimento em criptomoedas e muitas outras oportunidades para as moedas digitais crescerem no mercado.

Assim como em qualquer outro tipo de moeda, a cotação de criptomoedas pode variar de acordo com a sua demanda entre os investidores.

Assim, a lei da oferta e da procura, idealizada pelo economista britânico Adam Smith, funciona da seguinte forma: a estabilidade do mercado acontece ao ser traçada uma linha entre a procura dos compradores e a oferta dos vendedores. Desta forma, quanto maior for a busca por determinada moeda — ou bens, em geral — maior será o seu preço. O caso contrário segue a mesma regra: a menor procura gera um menor preço. Isso significa que as criptomoedas com cotação mais alta são influenciadas por uma grande procura. Entretanto, este é apenas um dos fatores que devem ser levados em consideração.

2. CRIPTOMOEDA ETHEREUM

Neste relatório será analisado um banco de dados da criptomoeda Ethereum, Na prática, a Ethereum não é uma criptomoeda, mas sim uma plataforma com base na tecnologia blockchain — a mesma utilizada pelo Bitcoin. Dentro da Ethereum, os investidores podem negociar contratos inteligentes e descentralizados da economia tradicional. Essas transações são pagas com uma criptomoeda própria: o Ether (ETH). Como há uma relação muito forte entre o Ether e a rede Ethereum, de um modo geral eles são usados como sinônimos.

A Ethereum foi idealizada pelo russo Vitalik Buterin. Em 2013, ele compartilhou a sua ideia e logo ganhou apoio de investidores para que ela fosse colocada em prática. Ele já sabia da existência do Bitcoin, mas queria evoluir a tecnologia. É por isso que pensou em lançar uma plataforma, em vez de simplesmente uma criptomoeda, a Ethereum não permite apenas a negociação financeira, mas principalmente a inclusão de contratos inteligentes. Esse modelo é capaz de facilitar a interação de tudo que, de alguma maneira, possa ser programado de forma digital.

3. OBJETIVOS DA ANÁLISE

- Verificar se o volume de transações é influenciado pelo valor de fechamento mensal
- Analisar a variação do Ethereum ao longo do tempo.
- Aplicar um modelo de previsão

4. METODOLOGIA

Para realização desta análise o banco de dados foi transformado para dados mensais e analisado no software R Studio (script em anexo).

5. DESCRIÇÃO DO BANCO DE DADOS

Este conjunto de dados fornece o histórico dos preços diários do Ethereum. Os dados começam a partir de 07 de agosto de 2015 e vão até 7 de dezembro de 2021. Com valores em USD (dólar americano). Dados retirados da plataforma Kaggle.

Variáveis:

Date	Data (dia especificado)
Open	Preço de abertura (preço da primeira transação por dia de negociação), variável quantitativa contínua.
High	Preço máximo (por dia de negociação), variável quantitativa contínua.
Low	Preço mínimo (por dia de negociação), variável quantitativa contínua
Close	Preço de fechamento (o preço da última transação para o dia de negociação), variável quantitativa contínua
Adj Close	Preço de fechamento ajustado (leva em consideração ações corporativas, como desdobramentos de ações, dividendos e garantias - ou seja, qualquer coisa que possa afetar o preço após o fechamento do dia), variável quantitativa contínua.
Volume	Volume de negociação (número de unidades monetárias trocadas por dia de negociação), variável quantitativa discreta.

6. ANÁLISE DE REGRESSÃO

Análise de regressão é um método estatístico que permite examinar a relação entre duas ou mais variáveis. Deste modo, identifica quais têm maior impacto diante de um tema de interesse.

A análise de regressão é útil para uma organização, pois permite determinar o grau em que as variáveis independentes influenciam as variáveis dependentes. Além disso, permite explicar um fenômeno e prever coisas sobre o futuro, assim como também pode obter informações comerciais valiosas e acionáveis.

Modelo de regressão linear simples

A regressão linear simples é a técnica mais utilizada. É uma forma que nos permite modelar uma relação entre dois conjuntos de variáveis. O resultado é uma equação que pode ser utilizada para fazer projeções ou estimativas dos dados.

Modelo de regressão linear múltipla

A regressão linear múltipla é uma técnica estatística responsável pela análise de situações envolvendo mais de uma variável. Esse método nos permite identificar quais são as variáveis independentes que podem explicar uma variável dependente, comprovar as causas e prever os valores aproximados.

7. SÉRIES TEMPORAIS

Uma série temporal é uma coleção de observações feitas sequencialmente ao longo do tempo. Em modelos de regressão linear com dados cross-section (Dados de corte transversal) a ordem das observações é irrelevante para a análise, em séries temporais a ordem dos dados é fundamental. Uma característica muito importante deste tipo de dados é que as observações vizinhas são dependentes e o interesse é analisar e modelar essa dependência.

Existem duas formas de estudar séries temporais. Uma análise da série temporal é um método para tentar entender a série temporal, de forma a entender a estrutura que gerou a série. Uma previsão a partir da série temporal procura construir um modelo matemático a partir do qual seja possível prever valores futuros da série.

8. ANÁLISE INICIAL DO BANCO DE DADOS

Após a remoção de dados faltantes (NA's) e especificação das variáveis, foi possível extrair as seguintes informações descritas nas tabelas 1 e 2

Tabela 1. 6 primeiras linhas do banco de dados com os as informações diárias de transações de Ethereum em USD

date	open	high	low	close	adj_close	volume
2015-08-07	2.83	3.54	2.52	2.77	2.77	164329
2015-08-08	2.79	2.80	0.715	0.753	0.753	674188
2015-08-09	0.706	0.880	0.629	0.702	0.702	532170
2015-08-10	0.714	0.730	0.637	0.708	0.708	405283
2015-08-11	0.708	1.13	0.663	1.07	1.07	1463100
2015-08-12	1.06	1.29	0.884	1.22	1.22	2150620

Nesta tabela observamos o preço de abertura (open) da moeda no dia, o maior valor (high) que a moeda atingiu, o menor valor (low), o preço de fechamento (close), o valor ajustado de fechamento (adj close) e o volume das transações.

Tabela 2. Variações da moeda Ethereum de 2015 a 2021

high	low	max_volume	min_volume
14892	0.421	84482912776	102128

Nesta tabela temos o maior valor (em dólar) registrado do Ethereum no intervalo de 2015 a 2021, o menor valor, o maior volume de transações e o menor valor de transações.

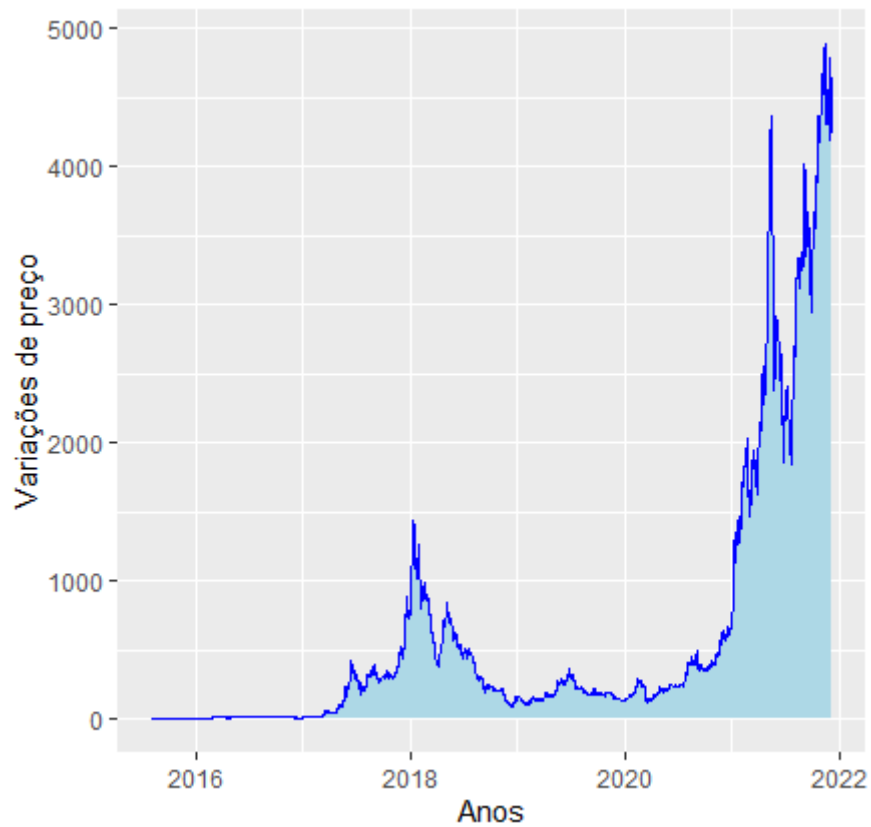


Gráfico 1: Variação diária do preço do Ethereum ao longo dos anos

Podemos observar que a partir de 2021 o Ethereum teve um salto em seu valor de mercado.

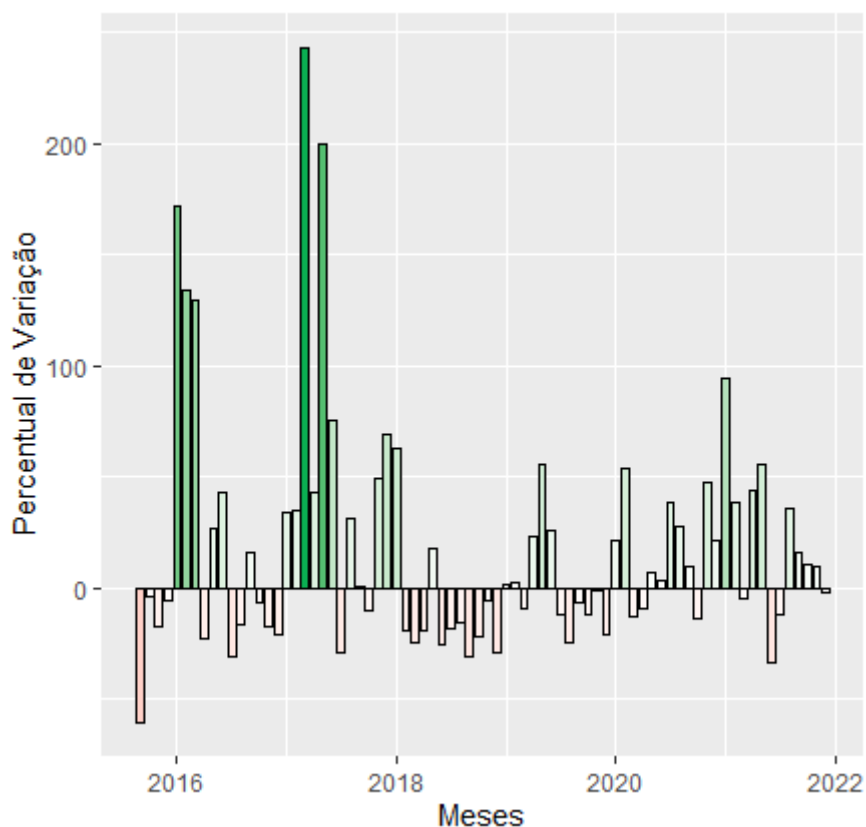


Gráfico 2: Percentual de variação da moeda por meses no intervalo de 2015 a 2021

Este gráfico mostra a porcentagem de subida e descida do Ethereum. Vemos que entre os anos de 2016 e 2018 o Ethereum apresentou uma grande variação positiva em seu valor.

Para obter melhores resultados na análise foi feita uma conversão dos dados diários para dados mensais.

Tabela 3: 6 primeiras linhas do banco de dados com os as informações de transações mensais de Ethereum em USD e o percentual de variação da moeda

meses	high	low	volume_max	volume_min	adj_close	percentual_mudanca
2015-08-01	3.54	0.629	4637030	164329	1.32	NA
2015-09-01	1.39	0.557	1942600	200541	0.989	-60.6
2015-10-01	1.35	0.421	2429200	102128	0.661	-3.42
2015-11-01	1.11	0.726	3060340	372173	0.931	-17.7
2015-12-01	1.05	0.770	1376280	164565	0.886	-5.52
2016-01-01	2.84	0.930	13939200	206062	0.1.49	171.0

Na tabela acima temos:

meses: refere-se aos meses de cada ano, começando sempre do dia primeiro de cada mês.

high: o valor máximo em que a moeda atingiu no mês.

low: o valor mínimo em que a moeda atingiu no mês.

volume_max: valor máximo de transações mensais.

volume_min: volume mínimo de transações mensais.

adj_close: média mensal dos valores de fechamento da moeda.

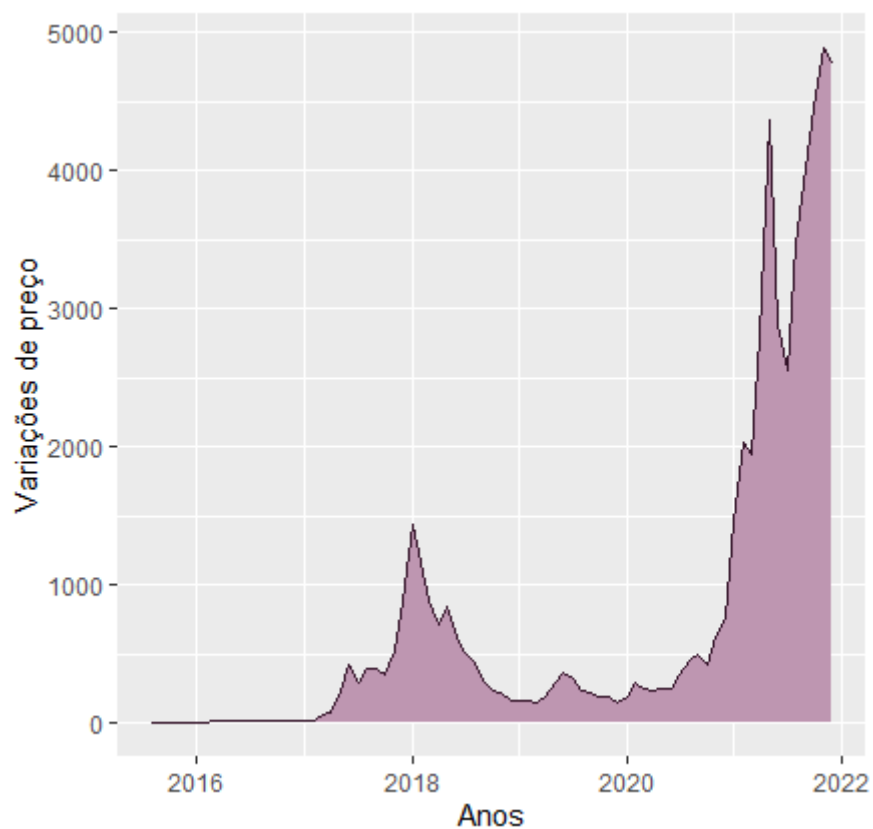


Gráfico 3: Variação mensal no preço do Ethereum ao longo dos anos

Comparando esse gráfico com o Gráfico 1, vemos que as são mais suaves, ou seja, os dados possuem menos variações.

9. ANÁLISE UTILIZANDO REGRESSÃO LINEAR SIMPLES

Nesta análise vamos verificar se o volume de transações de Ethereum é influenciado pelo valor de fechamento mensal

Variável dependente: **volume_max**

Variável independente: **adj_close**

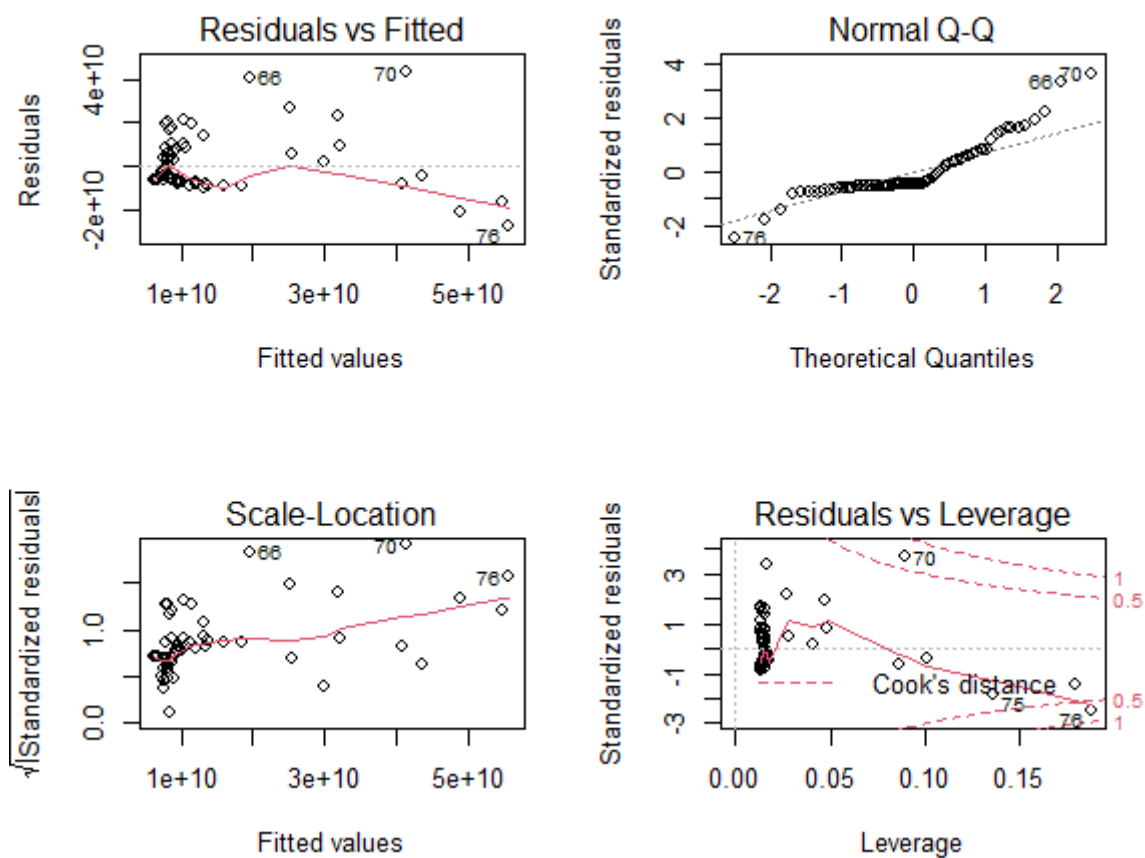


Gráfico 4: Visualização do comportamento dos resíduos do modelo criado (volume_max ~ adj_close)

O primeiro gráfico, Residuals vs Fitted, indica que os resíduos não seguem um comportamento linear. No segundo gráfico Normal Q-Q observamos que os resíduos estão próximos da linha da distribuição normal. Para verificarmos melhor se os resíduos atendem os pressupostos para adequação do modelo vamos utilizar os testes a seguir.

9.1 Aplicação dos testes.

Anova

Inicialmente vamos aplicar a análise de variância (ANOVA) para investigarmos se a interação dessas variáveis é significativa. Considerando 0,05 de significância.

Hipóteses:

H0: o valor de fechamento do Ethereum não influencia no volume de transações.

H1: o valor de fechamento do Ethereum possui influência no volume de transações.

Tabela 4: análise de variância

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
adj_close	1	1,04E+26	1,04E+26	68.626	0,000000003435
Residuals	75	1,14E+26	1,52E+24		

Vemos que pela análise de variância o valor P é menor que 0,05, nós rejeitamos H0. Concluindo assim que o valor de fechamento do Ethereum possui influência no volume de transações.

Teste de normalidade de resíduos

O teste utilizado será o Teste de Shapiro-Wilk, A hipótese nula do teste é que os dados seguem, pelo menos aproximadamente, a Normal. Já a hipótese alternativa é de que os dados não seguem uma distribuição Normal. Se o p-valor for menor que 0.05 rejeitamos a hipótese nula.

Dessa forma temos as seguintes hipóteses:

H0: Os resíduos seguem uma distribuição Normal.

H1: Os resíduos não seguem uma distribuição Normal.

Ao realizar o teste foi obtido o seguinte resultado p-value 1.399e-07, vemos que o p-valor foi menor que 0,05 então rejeitamos a H0, concluindo assim que os resíduos não seguem uma distribuição Normal.

Teste para verificação de Outliers (valores discrepantes) e pontos influentes

Ao realizar o teste foi obtido um valor mínimo de resíduo de -2.450338 e o valor máximo de 3.663594. Para que os Outliers não sejam considerados pontos influentes eles devem estar entre -3 e 3 sendo assim concluímos que não há Outliers influentes.

Teste para Homocedasticidade (Breusch-Pagan):

Em estatística, o teste de Breusch-Pagan permite testar a hipótese de homocedasticidade do termo de erro de um modelo de regressão linear.

Ele procura determinar a natureza da variância do termo de erro: se a variância é constante, então temos homocedasticidade; por outro lado, se variar, temos heterocedasticidade.

H_0 = há homocedasticidade.

H_1 = não há homocedasticidade.

O valor de p foi igual a 5.24e-05 podemos concluir que há homocedasticidade entre os resíduos

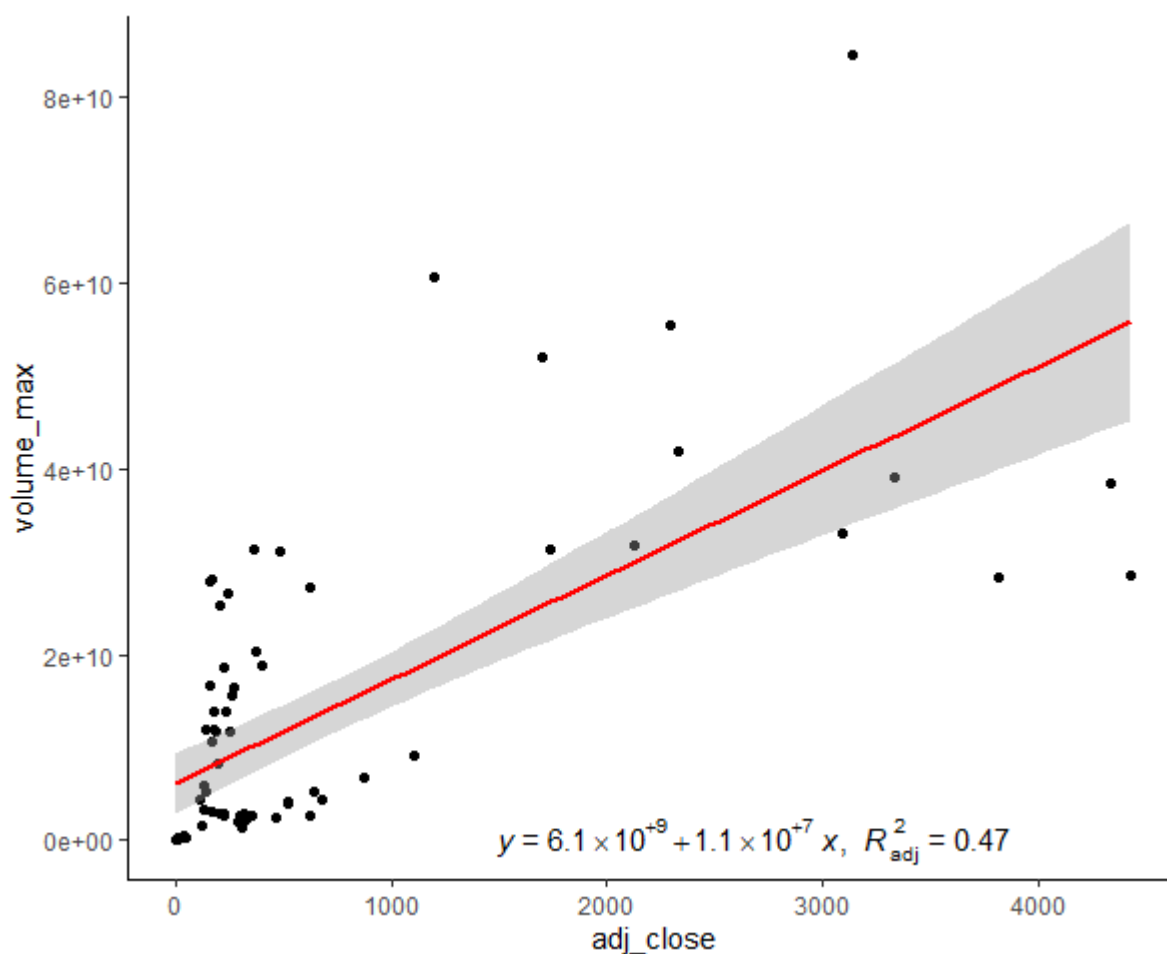


Gráfico 5: Visualização do modelo de regressão linear

Por não atender o pressuposto de normalidade nos resíduos e analisando graficamente a reta, vemos que ela não se ajusta bem aos dados.

10. ANÁLISE UTILIZANDO SÉRIES TEMPORAIS

Transformando os dados mensais em série temporal para verificar a variação do Ethereum ao longo do tempo.

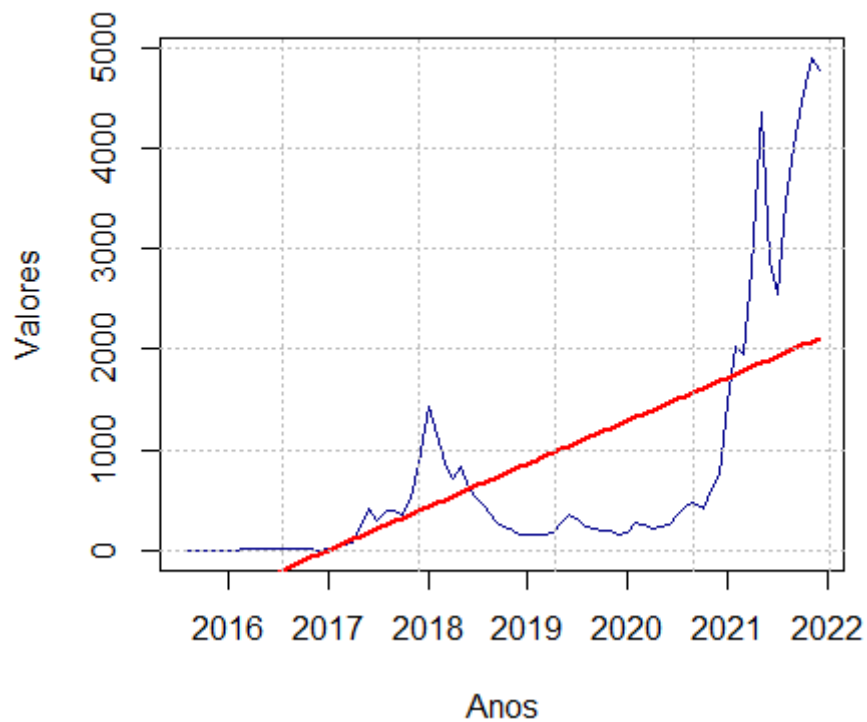


Gráfico 6: visualização da série temporal

Analisando o gráfico vemos que ele possui uma tendência positiva, ou seja, os dados possuem crescimento ao longo do tempo.

Decompondo a série utilizando um modelo aditivo temos a seguinte estrutura. Onde *data* são os valores observados, *trend* é a tendência e *seasonal* é a sazonalidade.

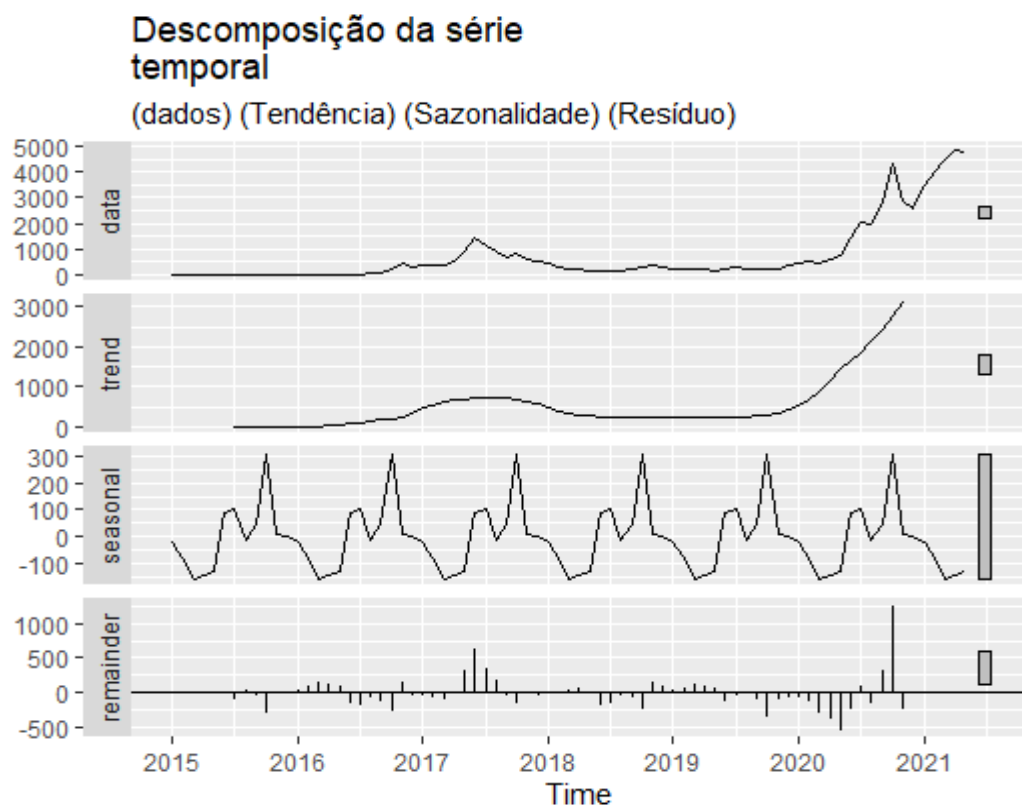


Gráfico 7: Decomposição da série

Verificando se a série é estacionária

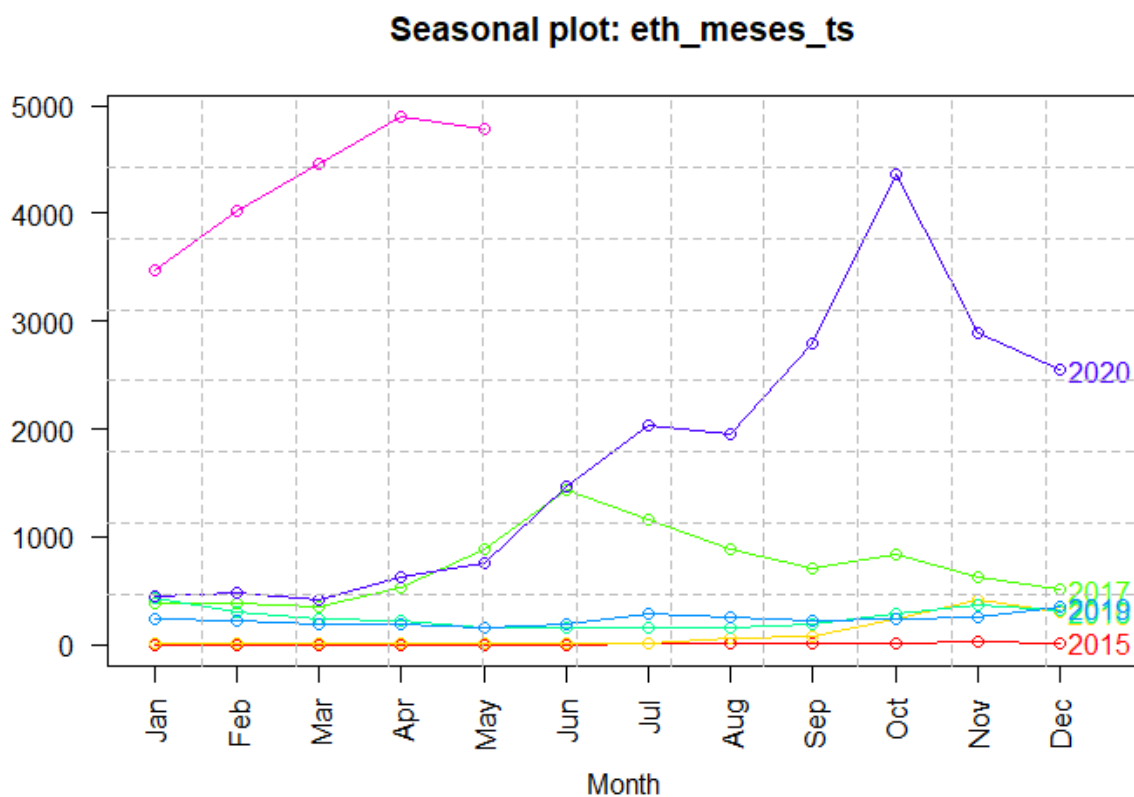


Gráfico 8: visualização da série do comportamento da série em cada ano

Cada linha do gráfico é referente aos anos da série. vemos a variação do Ethereum ao longo de cada ano.

Ajuste de Médias Móveis

A média móvel é um recurso utilizado para se identificar a tendência de um conjunto de dados dispostos em uma série de tempo. Dados e informações econômicas e financeiras variam segundo uma série de eventos ao longo do tempo.

Ao ajustar um modelo de Médias Móveis para os dados obtivemos o seguinte gráfico.

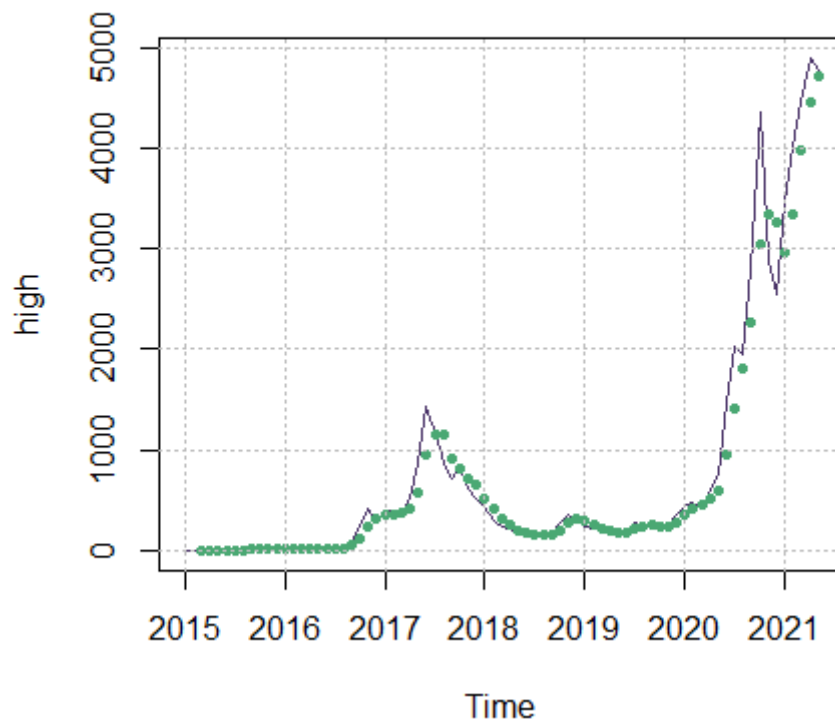


Gráfico 9: modelo de Médias móveis por trimestre

Neste gráfico foi ajustado um modelo de médias móveis por trimestre, vemos que o ajuste está próximo dos dados reais do gráfico.

Suavização Exponencial

Um método de suavização exponencial produz uma previsão para um período futuro. Ao se utilizar a técnica de projeção de tendência, previsões futuras poderão ser geradas. Ela é considerada precisa por levar em consideração a diferença entre as projeções reais e o que realmente aconteceu.

Aplicando a Suavização Exponencial de Holt Winters na série temos os seguintes resultados

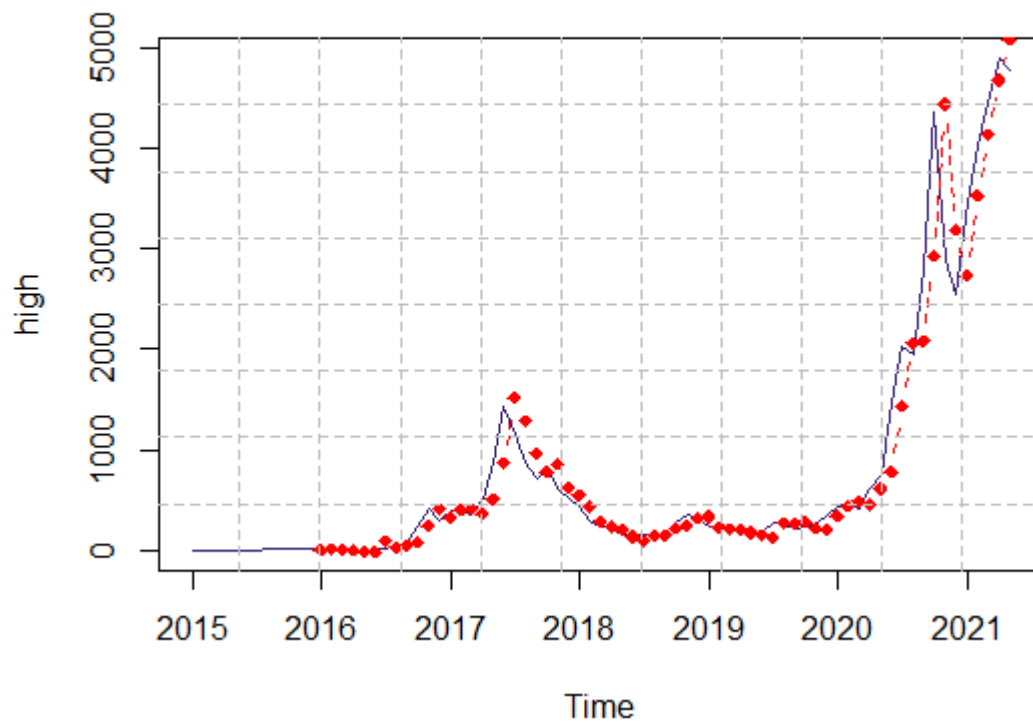


Gráfico 9: modelo ajustado aos dados

Os pontos vermelhos do gráfico se referem ao modelo ajustado aos dados.

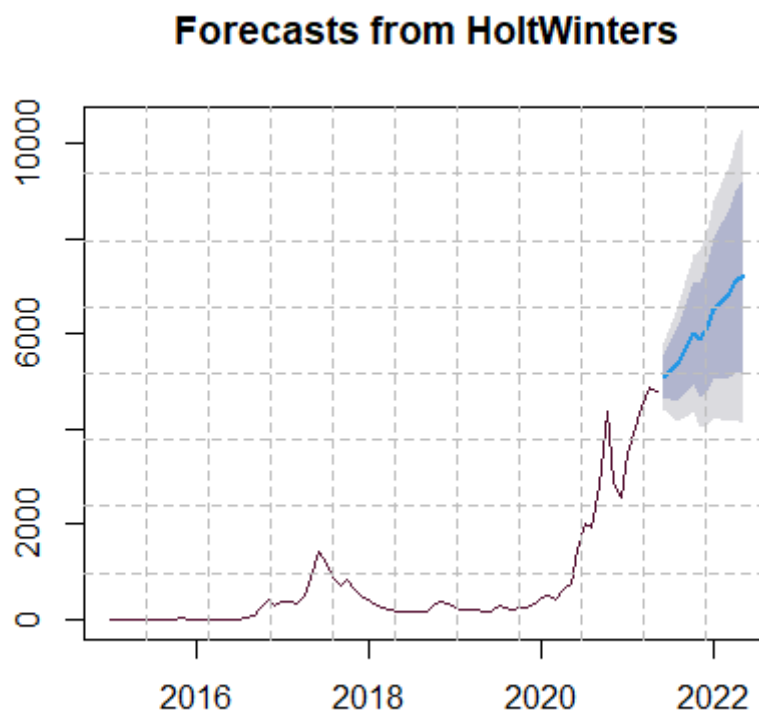


Gráfico 10: previsão da série para 2022

Neste gráfico vemos que a previsão do Ethereum para o ano de 2022 utilizando o modelo Holt Whinters é de crescimento.

- **Modelo Arima:**

coeficientes do modelo ARIMA (2,2,1)

	ar1	ar2	ma1
	-938	-4.574	-8.267
s.e.	1.121	1.077	753

Tabelas com os erros

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training Set	322.433	3.011.807	1.480.556	8.859.911	2.821.093	1.847.575	1.790.513

Forecasts from ARIMA(2,2,1)

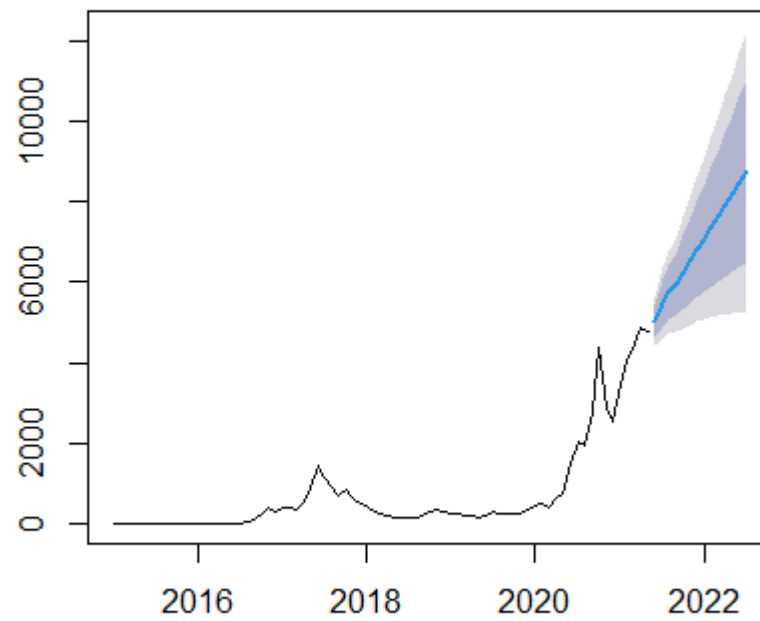


gráfico de previsão com o modelo Arima.

11. CONCLUSÃO

Utilizando regressão linear simples para verificar se o volume de transações é influenciado pelo valor de fechamento mensal vimos que de fato há essa influência.

Com a utilização dos modelos de séries temporais foi possível identificar o comportamento da série e ajustar um modelo de previsão para o ano de 2022, indicando crescimento na valorização do Ethereum.

12. REFERÊNCIAS

ANÁLISE de regressão simples x análise de regressão múltipla. Operdata,2020. Disponível em:

<<https://operdata.com.br/blog/analise-de-regressao-simples-x-analise-de-regressao-multipla/>> . Acesso em: 21 de março de 2022.

Montgomery, Douglas C., Cheryl L. Jennings e Murat Kulahci. *Introdução à análise e previsão de séries temporais* . John Wiley & Sons, 2015.

O QUE é uma criptomoeda? Tecnoblog, 2021. Disponível em:

<<https://tecnoblog.net/responde/o-que-e-uma-criptomoeda/>> . Acesso em: 21 de março de 2022.

ETHEREUM: como surgiu a segunda criptomoeda mais valiosa do mundo? InfoMoney.

Disponível em:

<[https://www.infomoney.com.br/guias/o-que-e-ethereum/#:~:text=Como%20e%20quando%20surgiu,%E2%80%9D%2C%20escreveu%20em%20seu%20blog./](https://www.infomoney.com.br/guias/o-que-e-ethereum/#:~:text=Como%20e%20quando%20surgiu,%E2%80%9D%2C%20escreveu%20em%20seu%20blog./>)> . Acesso em: 21 de março de 2022.

13. ANEXO

```
rm(list=ls())
```

```
require(ggplot2)
require(ggthemes)
library(readr)
library(tidyverse)
library(TTR)
library(forecast)
library(lubridate)
library(janitor)
library(dplyr)
library(TTR)
library(forecast)
library(timeSeries)
require(Kendall)
library(fpp2)
require(tseries)
```

```
pacman::p_load(dplyr, car, rstatix, lmtest, ggpubr,
               QuantPsyc, psych, scatterplot3d)
# library(writexl) #exportar o banco de dados limpo
```

```
#Importação e visualização dos dados
```

```
ETH_USD <- read_csv("Econometria/ETH-USD.csv")
#View(ETH_USD)
#View(head(ETH_USD))
summary(ETH_USD)
```

```
#limpeza dos dados
```

```
ETH_USD <- clean_names(ETH_USD)
summary(ETH_USD)
```

```
#remoção de linhas e colunas vazias
```

```
remove_empty(ETH_USD)
```

```
#verificando se há valores duplicados
```

```
get_dupes(ETH_USD)
```

```
#convertendo as colunas para o tipo certo de dados
```

```
ETH_USD$open = as.numeric(ETH_USD$open)
ETH_USD$high = as.numeric(ETH_USD$high)
ETH_USD$low = as.numeric(ETH_USD$low)
ETH_USD$close = as.numeric(ETH_USD$close)
ETH_USD$adj_close = as.numeric(ETH_USD$adj_close)
ETH_USD$volume = as.numeric(ETH_USD$volume)
ETH_USD$date = as.Date(ETH_USD$date)
summary(ETH_USD)
```

```
#remoção dos NA
```

```
ETH_USD = na.omit(ETH_USD)
summary(ETH_USD)
```

```
#analise exploratória
```

```
head(ETH_USD)
str(ETH_USD)
glimpse(ETH_USD)
View(ETH_USD)
```

```
# valores de maximo e minimo que o Ethereum alcançou em um dia e volume de transações
```

```
ETH_USD %>%
  summarise(
    high = max(high),
    low = min(low),
    max_volume = max(volume),
    min_volume = min(volume),
    adj_close= mean(adj_close)
  )
```

```
#transformação dos dados para visualização
```

```
eth_meses = ETH_USD %>%
```

```
  group_by(meses = floor_date(date,"month")) %>%
  summarize(high = max(high),
            low = min(low),
            volume_max = max(volume),
            volume_min = min(volume), adj_close= mean(adj_close)) %>%
```

```
mutate(percentual_mudanca = (high - lag(high)) / lag(high) * 100)
```

```
View(eth_meses)
```

```
head(eth_meses)
```

```
#grafico
```

```
eth_meses %>%  
  ggplot()+  
  geom_area(aes(x = meses, y = high), fill = "#be96b1", color = "#331228")+  
  labs(x = "Anos", y = "Variações de preço")
```

```
#percentual de mudança por mês
```

```
eth_meses %>%  
  ggplot()+  
  geom_col(aes(x = meses, y = percentual_mudanca, fill = percentual_mudanca), color =  
"black")+  
  labs(x = "Meses", y = "Percentual de Variação")+  
  scale_fill_gradient2(low = "#e53030", midpoint = 0, high = "#0aaf52")+  
  theme(legend.position = "none")
```

```
#nesta analise vamos verificar se o preço de alta e baixa possui relação com o volume  
maximo de
```

```
#transações do Ethereum por mês
```

```
#variavel dependente: volume
```

```
#variavel independente: high e low
```

```
#mod <- lm(volume ~ low + high, ETH_USD)
```

```
#trabalhando com os dados mensais
```

```
#mod1 <- lm(volume_max ~ low + high, eth_meses)
```

```
#analise gráfica
```

```
#par(mfrow=c(2,2))
```

```
#plot(mod1)
```

```
#par(mfrow=c(1,1))
```

```
#anova(mod1)
```

```
#aplicação dos testes
```

```
## Normalidade dos resíduos:  
#shapiro.test(mod1$residuals)
```

```
## Outliers nos resíduos:  
#summary(rstandard(mod1))
```

```
## Independência dos resíduos (Durbin-Watson):  
#durbinWatsonTest(mod1)
```

```
## Homocedasticidade (Breusch-Pagan):  
#bptest(mod1)
```

```
## Ausência de Multicolinearidade
```

```
#ETH_USDxx<-cbind(ETH_USD[3],ETH_USD[4],ETH_USD[7] )  
#head(ETH_USDxx)  
#pairs.panels(ETH_USDxx)
```

```
#eth_mesesxx<-cbind(eth_meses[3],eth_meses[4],eth_meses[2] )  
#head(eth_mesesxx)  
#pairs.panels(eth_mesesxx)
```

```
#pairs.panels(eth_meses)
```

```
### Multicolinearidade:  $r > 0.9$  (ou 0.8)
```

```
#vif(mod1)
```

```
### Multicolinearidade:  $VIF > 10$ 
```

```
##### Criação de um segundo modelo
```

```
## verificar se o volume de transações é influenciado pelo valor de fechamento mensal
```

```
mod2 <- lm( volume_max~adj_close,eth_meses)
```

```
par(mfrow=c(2,2))
```

```
plot(mod2)
```

```
par(mfrow=c(1,1))
```

```
anova(mod2)
```

```
#####
```

```
## Normalidade dos resíduos:
```

```
> shapiro.test(mod2$residuals)
```

Shapiro-Wilk normality test

```
data: mod2$residuals
```

```
W = 0.84249, p-value = 1.399e-07
```

```
> anova(mod2)
```

Analysis of Variance Table

Response: volume_max

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
adj_close	1	1.0446e+22	1.0446e+22	68.626	3.435e-12 ***
Residuals	75	1.1416e+22	1.5221e+20		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> ## Outliers nos resíduos:
```

```
> summary(rstandard(mod2))
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-2.450338	-0.507753	-0.490047	-0.004075	0.463148	3.663594

```
> ## Independência dos resíduos (Durbin-Watson):
```

```
> durbinWatsonTest(mod2)
```

lag	Autocorrelation	D-W Statistic	p-value
1	0.7827493	0.4078129	0

Alternative hypothesis: rho != 0

```
> ## Homocedasticidade (Breusch-Pagan):
```

```
> bptest(mod2)
```

studentized Breusch-Pagan test

```
data: mod2
```

```
BP = 16.359, df = 1, p-value = 5.24e-05
```

```
> # Passo 4: Análise do modelo
```

```
> summary(mod2)
```

Call:

```
lm(formula = volume_max ~ adj_close, data = eth_meses)
```

Residuals:

Min	1Q	Median	3Q	Max
-2.724e+10	-6.209e+09	-5.994e+09	5.634e+09	4.313e+10

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	6.101e+09	1.641e+09	3.719	0.000384 ***
adj_close	1.122e+07	1.355e+06	8.284	3.43e-12 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.234e+10 on 75 degrees of freedom

Multiple R-squared: 0.4778, Adjusted R-squared: 0.4708

F-statistic: 68.63 on 1 and 75 DF, p-value: 3.435e-12

```
ggplot(data = eth_meses, mapping = aes(x = adj_close, y = volume_max)) +  
  geom_point() +  
  geom_smooth(method = "lm", col = "red") +  
  stat_regline_equation(aes(label = paste(..eq.label.., ..adj.rr.label..,  
                                          sep = "*plain(\"\",\"\")~~~")),  
                        label.x = 1500, label.y = 2800) +  
  theme_classic()
```

```
#####
```

```
#para mod 2
```

```
#transformando dos dados em serie temporal para verificar a variação do Ethereum ao  
#longo do tempo
```

```
#transformando os dados de fechamento em série temporal
```

```
ETH_USD2 <- ETH_USD[1609:2311, c(1,2,3,4,5,6,7)]  
head(ETH_USD2)  
eth_serie <- ts(ETH_USD2[,6], start=2020, freq=365)
```

```
#linha de tendência
```

```
N <- nrow(eth_serie)
Tempo <- 1:N
X.lm <- lm(eth_serie ~ Tempo)
X.pred <- predict(X.lm) ## Previsão da série observada
M <- seq(as.Date("2020/1/1"),as.Date("2021/12/3"),"days")
plot(M, eth_serie, type="l", col="darkblue", xlab = "Anos",ylab = "Valores", main = "Gráfico
7: Variação do Ethereum
    ao longo dos anos")
lines(M, X.pred, col= "red", lwd=2,lty=2)
```

```
# aplicando modelo multiplicativo para decompor a serie
```

```
ETH_USD_serie2<- ts(ETH_USD[,6],freq=365)
eth_adt <- decompose(ETH_USD_serie2, type = "mult")
plot(eth_adt, col= "darkblue")
```

```
# utilizando a Suavização Exponencial de Holt Winters
#pois a serie possui tendencia de sazonalidade aditiva
```

```
modelo_eth <- HoltWinters(ETH_USD_serie2)
```

```
#valores estimados da serie
```

```
modelo_eth$fitted
```

```
#gráfico
```

```
plot(ETH_USD_serie2)
lines(modelo_eth$fitted[,1],pch = 18, col = "red", type = "l",
      lty = 2, lwd = 1)
```

```
#####
```

```
#serie temporal por meses
```

```
eth_meses_ts<- ts(eth_meses[,2], start = 2015, freq= 12)
plot(eth_meses_ts)
```

```
N <- nrow(eth_meses_ts)
Tempo <- 1:N
X.lm <- lm(eth_meses_ts ~ Tempo)
```

```

X.pred <- predict(X.lm) ## Previsão da série observada
M <- seq(as.Date("2015/8/1"),as.Date("2021/12/1"),"month")
plot(M, eth_meses_ts, type="l",lwd=1, col="darkblue", xlab = "Anos",ylab = "Valores")
lines(M, X.pred, col= "red", lwd=2,lty=1)
grid(nx = NULL, ny = NULL,
      lty = 3,
      lwd = 1,
      col = "gray")

```

#decomposição da serie

```

eth_adt <- decompose(eth_meses_ts)
plot(eth_adt, col= "#422a60", lwd=1)
grid(nx = NULL, ny = NULL,
      lty = 2,
      lwd = 1,
      col = "gray")

```

```

#####
###

```

ver se a serie é estacionária

```

seasonplot(eth_meses_ts,col=rainbow(7),year.labels = TRUE,
            las=2)
grid(nx = NULL, ny = NULL,
      lty = 2,
      lwd = 1,
      col = "gray")

```

#Teste de Mann-Kendall para verificar tendência na série temporal, sob a
#hipótese:

#H0:"As observações da série são independentes e identicamente distribuídas"
#H1:"As observações da série possuem tendência monotônica no tempo"

```

MannKendall(eth_meses_ts)
tau = 0.589, 2-sided pvalue =< 2.22e-16
> adf.test(eth_meses_ts)

```

Augmented Dickey-Fuller Test

```

data: eth_meses_ts
Dickey-Fuller = 0.83671, Lag order = 4, p-value = 0.99
alternative hypothesis: stationary

```


Warning message:

In adf.test(eth_meses_ts) : p-value greater than printed p-value

```
> ethdif = diff(eth_meses_ts)
```

```
> ndiffs(ethdif)
```

```
[1] 1
```

```
# analise grafica de autocorrelacao
```

```
par(mfrow=c(2,2),mar=c(4,4,4,1)+.1)
```

```
plot(eth_meses_ts,ylab="Ethereum")
```

```
acf(eth_meses_ts,main="Série não-estacionária")
```

```
plot(ethdif,ylab="Ethereum")
```

```
acf(ethdif,main="Série Estacionária")
```

```
par(mfrow=c(1,1))
```

```
# decomposição
```

```
autoplot(decompose(ethdif,"additive")) + labs(title = "Descomposição da série  
temporal", subtitle = "(dados) (Tendência) (Sazonalidade) (Resíduo)")
```

```
autoplot(decompose(eth_meses_ts,"additive")) + labs(title = "Descomposição da série  
temporal", subtitle = "(dados) (Tendência) (Sazonalidade) (Resíduo)")
```

```
# Teste para Sazonalidade
```

```
fit1 <- ets(ethdif)
```

```
fit2 <- ets(ethdif,model="ANN")
```

```
deviance <- 2*c(logLik(fit1) - logLik(fit2));deviance
```

```
df <- attributes(logLik(fit1))$df - attributes(logLik(fit2))$df ;df
```

```
#P value
```

```
1-pchisq(deviance,df)
```

```
plot(forecast(fit1))
```

```
#H0:"Existe um componente Sazonal na Série"
```

```
#H1:"Não Existe um componente Sazonal na Série"
```

```
ggseasonplot(eth_meses_ts, polar = T)
```

```
modelo1 = auto.arima(eth_meses_ts, seasonal = FALSE, stationary = FALSE)
```

```
summary(modelo1)
```

```
checkresiduals(modelo1)
```

```
#####
```

```
Meses = c("J","F","M","A","M","J","J","A","S","O","N","D")
```

```
# Gráficos de Series de Tiempo
```

```
plot.ts(eth_meses_ts, xlab="Anos",ylab="Índice de valores",axes=FALSE)
```

```
axis(1,c(2015,2016,2017, 2018, 2019,2020,2021),las=3)
```

```
axis(2,c(0,500,1000,1500,2000,2500,3000,3500,4000, 4500, 5000))
```

```
points(eth_meses_ts,cex=0.8, font=4, col=1:4)
```

```
#####  
#####
```

```
#media móvel por trimestre
```

```
eth_M3<- SMA(eth_meses_ts, n=3)
```

```
plot(eth_meses_ts, col= "#493267")
```

```
points(eth_M3, pch=20, col= "#4aa873")
```

```
grid(nx = NULL, ny = NULL,
```

```
lty = 3,
```

```
lwd = 1,
```

```
col = "gray")
```

```
#####3
```

```
# utilizando a Suavização Exponencial de Holt Winters
```

```
#pois a serie possui tendencia de sazonalidade aditiva
```

```
modelo_ethM <- HoltWinters(eth_meses_ts)
```

```
a<- forecast(modelo_ethM, h=24)
```

```
plot(a)
```

```
#valores estimados da serie
```

```
head(modelo_ethM$fitted)
```

```
#gráfico
```

```
plot(eth_meses_ts, col= "#351d74")  
lines(modelo_ethM$fitted[,1],pch = 18, col = "red", type = "b",  
      lty = 2, lwd = 1)  
grid(nx = 11, ny = 8,  
      lty = 2,  
      lwd = 1,  
      col = "gray")
```

```
#previsão  
a<- forecast(modelo_ethM, h=12)  
plot(a, col= "#520c2e")  
grid(nx = 11, ny = 8,  
      lty = 2,  
      lwd = 1,  
      col = "gray")
```

```
#####  
#####  
#####  
#####
```

```
#modelo de médias móveis
```

```
modelo1 = auto.arima(eth_meses_ts, seasonal = FALSE, stationary = FALSE)
```

```
summary(modelo1)
```

```
checkresiduals(modelo1)
```

```
Meses = c("J","F","M","A","M","J","J","A","S","O","N","D")
```

```
# Gráficos de Series de Tiempo  
plot.ts(eth_meses_ts, xlab="Anos",ylab="Índice de preços do consumidor",axes=FALSE)
```

```
axis(1,c(2015,2016,2017, 2018, 2019,2020,2021,2022),las=2)  
axis(2,c(0,500,1000,2000,3000,4000,5000))
```

```
points(eth_meses_ts, pch=Meses, cex=0.8, font=4, col=1:4)
```

```
bb<- forecast(modelo1, h=14)
plot(bb)
```

```
#####  
####
```