

Análise Descritiva

Informações iniciais:

- O Database escolhido contém os dados do Bitcoin desde sua criação (2014).

- Para nossa previsão, usaremos as seguintes datas de filtragem: Data de Início – 25/05/2019;

Data Final: 25/05/2021;

Data de previsão (Até) – 25/05/2022.

- Para analisar os dados, precisaremos filtrar a tabela entre as datas escolhidas, modificar os dados para os formatos desejados, identificar os tipos de variáveis e calcular as estatísticas básicas para uma análise mais precisa (Média, moda, mediana, etc...).

Instalando bibliotecas:

```
install.packages("ggplot2")
install.packages("pacman")
install.packages("dplyr")
library(ggplot2)
```

Setup do script em lotes:

```
rm(list = ls())
library(readxl)
library(ggplot2)
library(dplyr)
library(pacman)
```

Definindo diretório e carregando o Database:

```
setwd("D:/Backup Vitin/Tudo/Facul/7o Semestre/Projeto UC/Análise Descritiva")
dados <- read.csv("D:/Backup Vitin/Tudo/Facul/7o Semestre/Projeto UC/Análise Descritiva/BTC.csv")
View(dados)
```

Filtrando datas:

```
dados_filtrado <- slice(dados, 1712:2443)
```

Listando variável target e calculando média:

```
dados_filtrado$Close
mean(dados$Close)
```

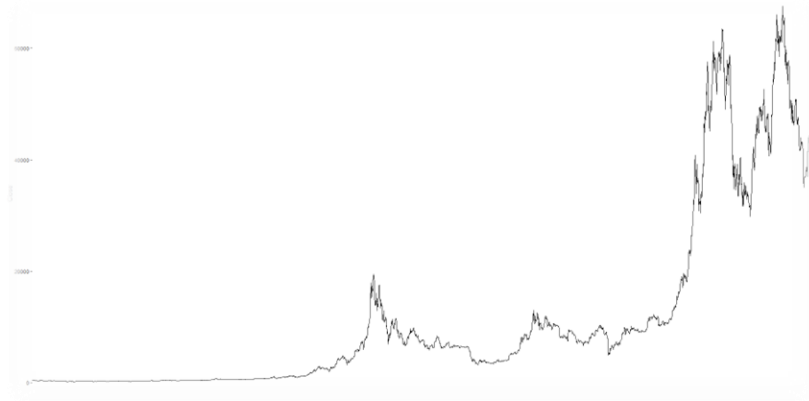
```
> mean(dados$Close)
[1] 11682.89
```

Calculando média das colunas restantes:

```
> mean(dados_filtrado$open)
[1] 17879.65
> mean(dados_filtrado$High)
[1] 18386.18
> mean(dados_filtrado$Low)
[1] 17325.64
> mean(dados_filtrado$Adj.Close)
[1] 17920.49
> mean(dados_filtrado$volume)
[1] 35786438275
```

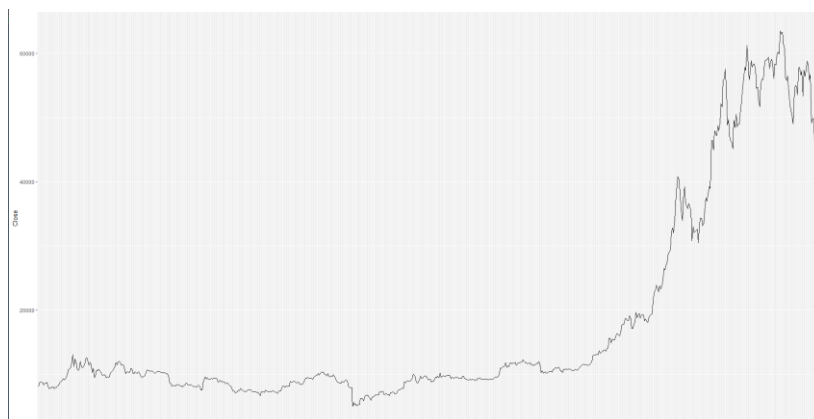
Testando gráfico de preço sem datas filtradas:

```
ggplot(data = dados)+
  geom_line(mapping=aes(x=Date, y=Close, group=1))
```



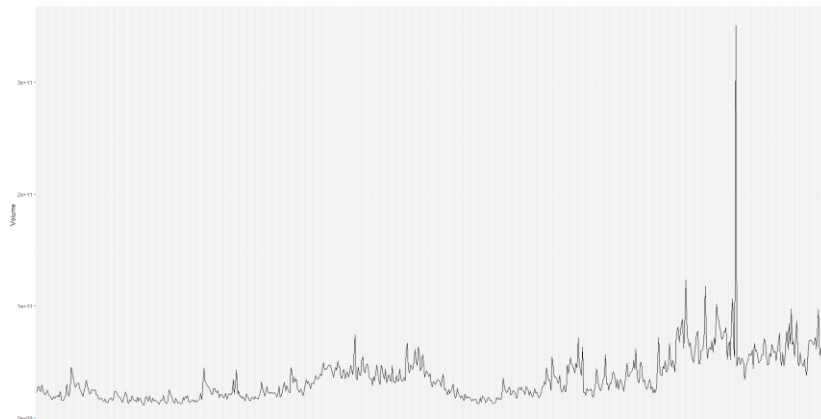
Testando gráfico de preço com datas filtradas:

```
ggplot(data = dados_filtrado)+
  geom_line(mapping=aes(x=Date, y=Close, group=1))
```



Testando gráfico de volume:

```
ggplot(data = dados_filtrado)+  
  geom_line(mapping=aes(x=Date, y=Volume, group=1))
```



Calculando estatísticas da variável target:

```
summary(dados_filtrado$Close)
```

```
> summary(dados_filtrado$Close)  
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   
  4971   8787   10242   17920   18089   63503
```

Calculando estatísticas das colunas restantes:

```
> summary(dados_filtrado$open)  
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   
  5003   8780   10232   17880   17876   63524  
> summary(dados_filtrado$High)  
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   
  5332   8913   10398   18386   18434   64863  
> summary(dados_filtrado$Low)  
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   
  4107   8620   9996   17326   17574   62209  
> summary(dados_filtrado$Adj.Close)  
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   
  4971   8787   10242   17920   18089   63503  
> summary(dados_filtrado$volume)  
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   
 1.145e+10 2.022e+10 3.012e+10 3.579e+10 4.633e+10 3.510e+11
```

Calculando variância da variável target:

```
var(dados_filtrado$Close)
```

```
> var(dados_filtrado$Close)  
[1] 258348590
```

Calculando variância das colunas restantes:

```
> var(dados_filtrado$Open)
[1] 257942702
> var(dados_filtrado$High)
[1] 275175303
> var(dados_filtrado$Low)
[1] 238003562
> var(dados_filtrado$Adj.Close)
[1] 258348590
> var(dados_filtrado$Volume)
[1] 5.051454e+20
```

Calculando desvio padrão da variável target:

```
sd(dados_filtrado$Close)
```

```
> sd(dados_filtrado$Close)
[1] 16073.23
```

Calculando desvio padrão das colunas restantes:

```
> sd(dados_filtrado$Open)
[1] 16060.59
> sd(dados_filtrado$High)
[1] 16588.41
> sd(dados_filtrado$Low)
[1] 15427.36
> sd(dados_filtrado$Adj.Close)
[1] 16073.23
> sd(dados_filtrado$Volume)
[1] 22475439779
```

Gráfico de dispersão:

