

R Markdown Output

Last run on: 2021-02-13 06:26:34

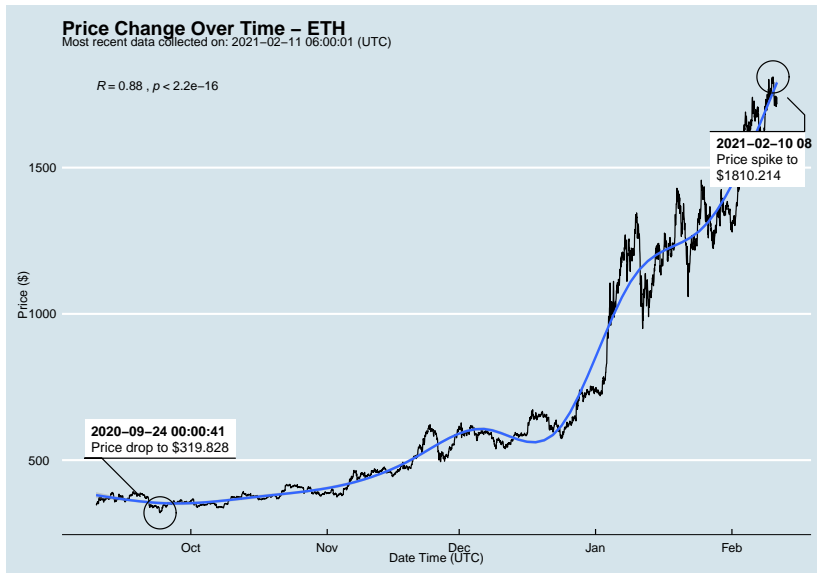
2021-02-13 06:26:34

Overview

This document has code embedded throughout. In the next section we will create a visualization using the already loaded dataset `eth_data`:

```
datatable(eth_data)
```

Price Chart - Ethereum



Python Code Example

```
import pandas as pd
# Create the Python object from R
df = r.cryptodata
# Show the new Python dataframe
df
```

```
##           pair symbol  ask_1_price      date_time_utc
## 0      BTCUSD   BTC    44640.980 2021-02-11 06:00:01
## 1      ETHUSD   ETH     1716.652 2021-02-11 06:00:01
## 2      ETHUSD   ETH     1723.927 2021-02-11 05:00:01
## 3      BTCUSD   BTC    44718.930 2021-02-11 05:00:00
## 4      ETHUSD   ETH     1740.290 2021-02-11 04:00:01
## ...      ...      ...           ...           ...
## 8001 BTCUSD   BTC    11972.900 2020-08-10 06:03:50
## 8002 BTCUSD   BTC    11985.890 2020-08-10 05:03:48
## 8003 BTCUSD   BTC    11997.470 2020-08-10 04:32:55
## 8004 BTCUSD   BTC     10686.880                NaT
## 8005 ETHUSD   ETH       357.844                NaT
```

One more Python example

The code below creates a new column `price_percentile` that specifies if the price for the row was in the upper or lower 50th percentile of prices (BTC should be upper and ETH lower):

```
import numpy as np
# Create a new column based on the ask_1_price value:
df['price_percentile'] = np.where(df['ask_1_price'] >
                                  np.percentile(df['ask_1_p
                                  'upper 50th percentile of price
                                  'lower 50th percentile of price

# Show modified dataframe:
df[['symbol', 'ask_1_price', 'price_percentile']]
```

##	symbol	ask_1_price	price_percentile
## 0	BTC	44640.980	upper 50th percentile of price
## 1	ETH	1716.652	lower 50th percentile of price
## 2	ETH	1723.927	lower 50th percentile of price
## 3	BTC	44718.930	upper 50th percentile of price
## 4	ETH	1740.290	lower 50th percentile of price