R Markdown Output

Last run on: 2021-03-23 06:28:21

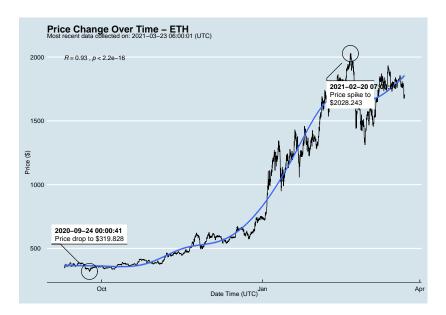
2021-03-23 06:28:21

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	ETHUSD	ETH	1676.748	2021-03-23	06:00:01
##	1	BTCUSD	BTC	53528.000	2021-03-23	06:00:00
##	2	ETHUSD	ETH	1693.591	2021-03-23	05:00:01
##	3	BTCUSD	BTC	54433.970	2021-03-23	05:00:00
##	4	ETHUSD	ETH	1701.902	2021-03-23	04:00:01
##						
##	9917	BTCUSD	BTC	11972.900	2020-08-10	06:03:50
##	9918	BTCUSD	BTC	11985.890	2020-08-10	05:03:48
##	9919	BTCUSD	BTC	11997.470	2020-08-10	04:32:55
##	9920	BTCUSD	BTC	10686.880		NaT
##	9921	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_] 'upper 50th percentile of price 'lower 50th percentile of price # Show modified dataframe:

```
df[['symbol', 'ask_1_price', 'price_percentile']]
        symbol
                                             price_percenti
##
                ask_1_price
           ETH
                   1676.748
                             lower 50th percentile of price
## 0
```

53528.000 upper 50th percentile of price ## 1 BTC ETH 1693.591 lower 50th percentile of price ## 2 ## 3 BTC 54433.970 upper 50th percentile of price ## / ETI 1701 000 larram EO+h namaan+ila af nmia