R Markdown Output

Last run on: 2021-04-09 06:13:54

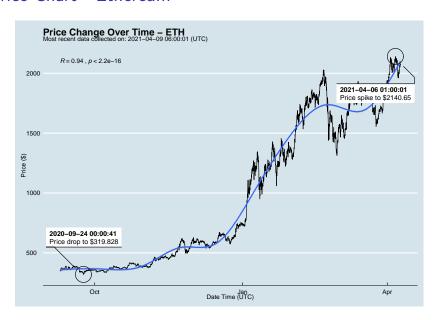
2021-04-09 06:13:54

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	$\mathtt{date_time_utc}$
##	0	BTCUSD	BTC	57960.010	2021-04-09 06:00:01
##	1	ETHUSD	ETH	2069.008	2021-04-09 06:00:01
##	2	BTCUSD	BTC	58190.010	2021-04-09 05:00:01
##	3	ETHUSD	ETH	2073.089	2021-04-09 05:00:01
##	4	BTCUSD	BTC	57982.410	2021-04-09 04:00:01
##					
##	10733	BTCUSD	BTC	11972.900	2020-08-10 06:03:50
##	10734	BTCUSD	BTC	11985.890	2020-08-10 05:03:48
##	10735	BTCUSD	BTC	11997.470	2020-08-10 04:32:55
##	10736	BTCUSD	BTC	10686.880	NaT
##	10737	ETHUSD	ETH	357.844	NaT

One more Python example

3

/

ETH

DTC

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
##
                 ask_1_price
                                               price_percent:
            BTC
                   57960.010
                               upper 50th percentile of price
## 0
                    2069.008
                               lower 50th percentile of price
## 1
            ETH
            BTC
                   58190.010
                               upper 50th percentile of price
## 2
```

lower 50th percentile of price

unnam EO+h namaan+ila af nwi

2073.089

E7000 /110