

R Markdown Output

Last run on: 2021-07-10 06:11:40

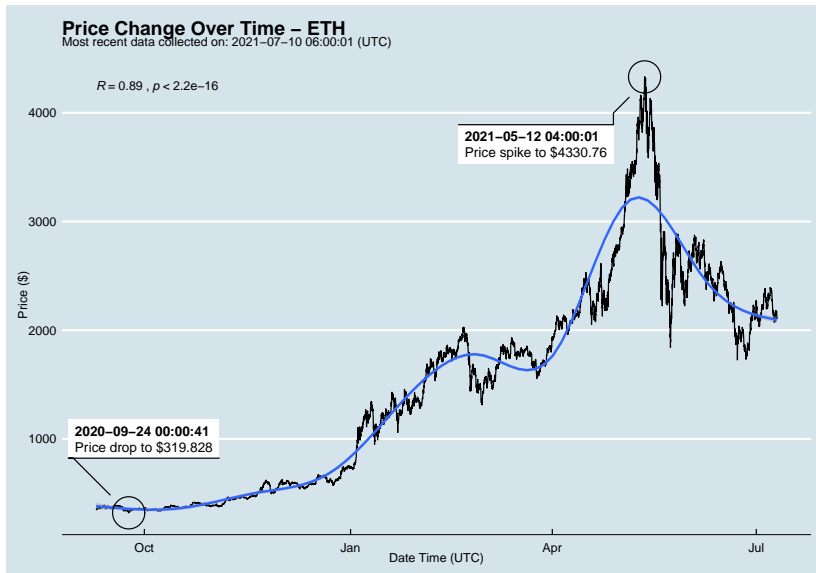
2021-07-10 06:11:40

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	2114.555	2021-07-10 06:00:01
## 1	BTCUSD	BTC	33653.860	2021-07-10 06:00:00
## 2	BTCUSD	BTC	33651.580	2021-07-10 05:00:01
## 3	ETHUSD	ETH	2107.891	2021-07-10 05:00:01
## 4	ETHUSD	ETH	2132.825	2021-07-10 04:00:01
##
## 15101	BTCUSD	BTC	11372.920	2020-08-12 00:03:55
## 15102	BTCUSD	BTC	11868.320	2020-08-11 03:03:52
## 15103	BTCUSD	BTC	11900.900	2020-08-11 02:03:53
## 15104	BTCUSD	BTC	11863.170	2020-08-11 01:03:53
## 15105	BTCUSD	BTC	11906.130	2020-08-11 00:03:49

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_price'],
                                                  50,
                                                  '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	0	ETH	2114.555	lower 50th percentile of price
## 1	1	BTC	33653.860	upper 50th percentile of price
## 2	2	BTC	33651.580	upper 50th percentile of price
## 3	3	ETH	2107.891	lower 50th percentile of price
## 4	4	ETH	2132.825	lower 50th percentile of price