R Markdown Output

Last run on: 2020-12-25 06:28:07

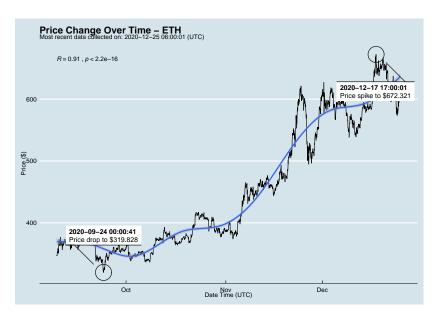
2020-12-25 06:28:07

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	23604.240	2020-12-25	06:00:01
##	1	ETHUSD	ETH	609.549	2020-12-25	06:00:01
##	2	BTCUSD	BTC	23608.820	2020-12-25	05:00:01
##	3	ETHUSD	ETH	609.330	2020-12-25	05:00:01
##	4	BTCUSD	BTC	23479.090	2020-12-25	04:00:01
##						
##	5697	BTCUSD	BTC	11972.900	2020-08-10	06:03:50
##	5698	BTCUSD	BTC	11985.890	2020-08-10	05:03:48
##	5699	BTCUSD	BTC	11997.470	2020-08-10	04:32:55
##	5700	BTCUSD	BTC	10686.880		NaT
##	5701	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']] price_percenti ## symbol ask_1_price BTC 23604.240 upper 50th percentile of price ## 0

609.549 lower 50th percentile of price ## 1 ETH BTC 23608.820 upper 50th percentile of price ## 2 ## 3 ETH 609.330 lower 50th percentile of price ## / DTC 22/70 000 unnam EO+h namaan+ila af nmia