

## R Markdown Output

Last run on: 2021-03-01 06:30:20

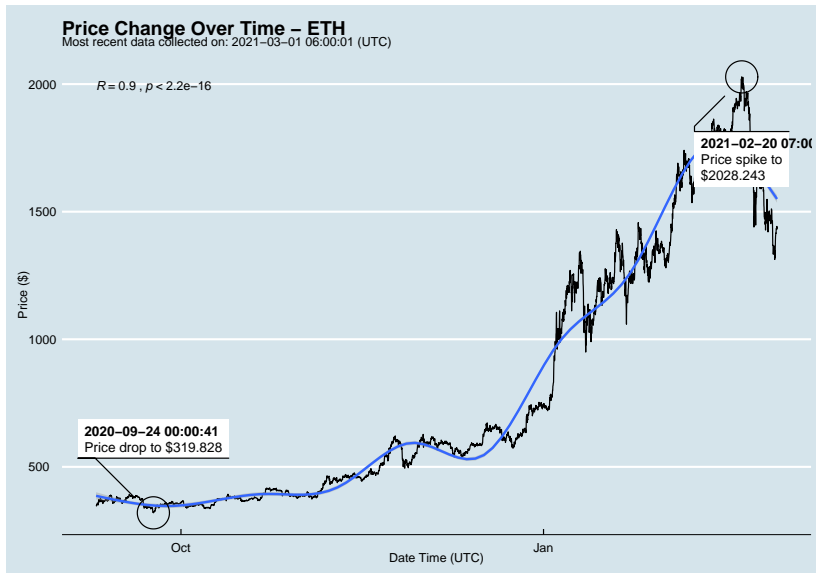
2021-03-01 06:30:20

# 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    46179.370 2021-03-01 06:00:01
## 1      ETHUSD   ETH     1433.548 2021-03-01 06:00:01
## 2      ETHUSD   ETH     1442.465 2021-03-01 05:00:01
## 3      BTCUSD   BTC    46462.770 2021-03-01 05:00:00
## 4      ETHUSD   ETH     1430.300 2021-03-01 04:00:01
## ...      ...    ...           ...           ...
## 8865 BTCUSD   BTC    11972.900 2020-08-10 06:03:50
## 8866 BTCUSD   BTC    11985.890 2020-08-10 05:03:48
## 8867 BTCUSD   BTC    11997.470 2020-08-10 04:32:55
## 8868 BTCUSD   BTC     10686.880                NaT
## 8869 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	0	BTC	46179.370	upper 50th percentile of price
## 1	1	ETH	1433.548	lower 50th percentile of price
## 2	2	ETH	1442.465	lower 50th percentile of price
## 3	3	BTC	46462.770	upper 50th percentile of price
## 4	4	ETH	1430.300	lower 50th percentile of price