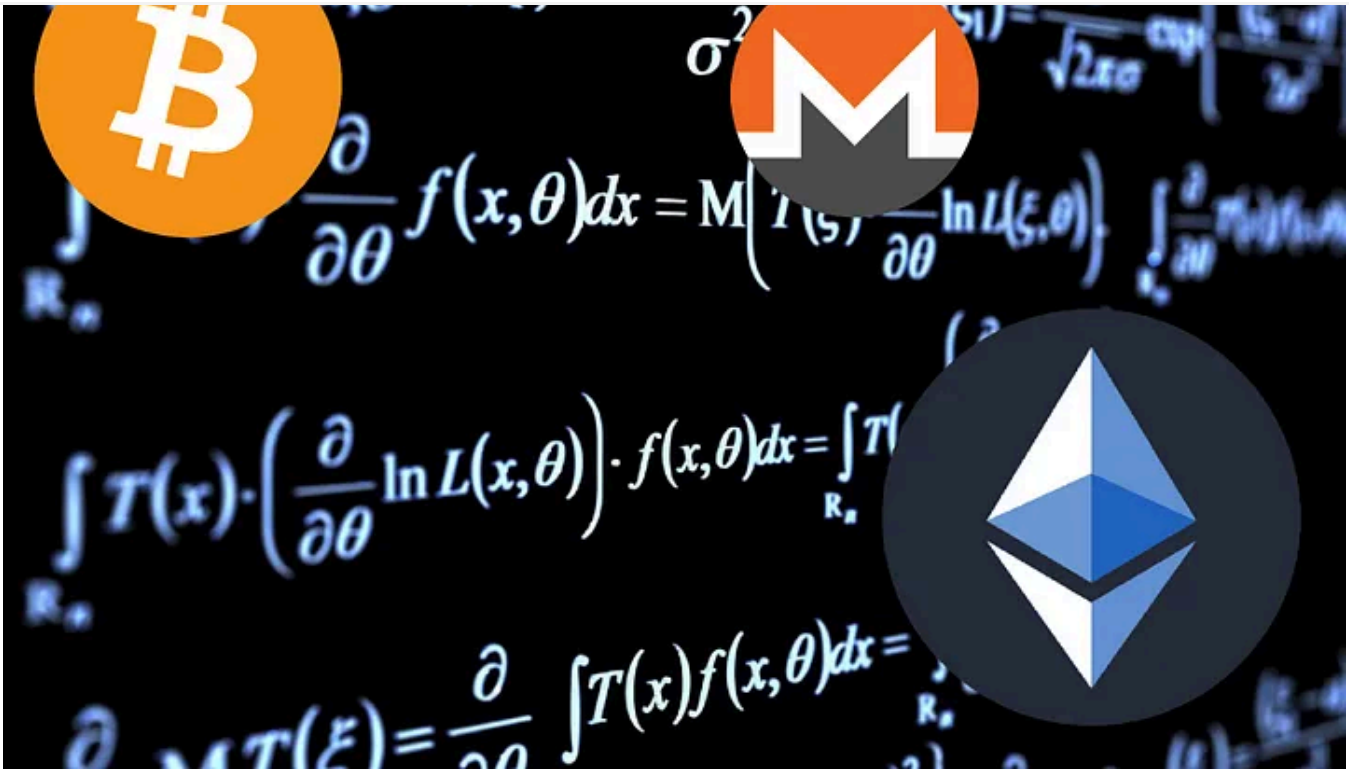


Advanced candlesticks for machine learning (ii): volume and dollar bars

In this article we will learn how to build volume and dollar bars and we will explore what advantages they offer in respect to traditional time-based candlesticks and tick-bars. Finally, we will analyze two of their statistical properties — autocorrelation and normality of returns — in a large dataset of 16 cryptocurrency trading pairs

Gerard Martínez · Follow
Published in Towards Data Science
9 min read · May 2, 2019

Introduction

In a previous post we learned how to build tick bars and assessed their particular ability to self-regulate the sampling rate based on a higher or

bars define activity as the number of assets traded in the exchange — for instance, number of Bitcoins exchanged. For dollar bars, activity is defined as fiat value exchanged — for instance, sample a bar every time 1000$ in assets are exchanged — which can be measured in dollars but also in Euro, Yens, etc.

Therefore, each bar type understands and synchronizes to market activity in a different way and this differential understanding brings its advantages and disadvantages. Let's dig into them.

Advantages and disadvantages of tick, volume and dollar bars

With the tick bars we found a way to scan the trade history of an exchange and sample more bars simply when more trades where executed in the exchange. While a strong correlation between number of trades placed and information arrival may exist, the correlation is not guaranteed. For instance, a well acquainted algorithm or trader may automatically place very small, repetitive orders to influence the sentiment of the market (by turning the trade history "green"), to hide the total amount of volume (also known as iceberg orders) or simply to disorient other trading bots by falsifying information arrival.

A potential solution to this scenario is to use volume bars instead. Volume bars do not care about the sequence or number of trades, they just care about the total volume of these trades. For them, information arrival is increased volume traded between the exchange users. This way, volume bars are able to bypass misleading interpretations of the number of trades being executed at the cost of loosing any information that could lie hidden in the actual sequence of trades.

Another interesting feature about volume bars, which may sound obvious but is important to notice, is that market volume information is intrinsically coded on the bar themselves: each volume bar is a bucket of a predefined volume. Again, this may sound obvious, but for long time, and still today, lots of researchers in the financial space are clue-less about how to include volume information in their predictive models. Volume bars yield volume information out-of-the-box.

Now, the problem with the volume bars is that volume exchanged may be very correlated with the actual value of the asset being exchanged. For instance, with

the revaluation of the asset. A way to correct for this fluctuation is by, instead of counting the number of assets exchanged (volume bars), counting the quantity of fiat value exchanged (dollar bars), which happens to be in dollars for the BTC-USD pair, but could also be in Euros for the ETH-EUR pair, etc.

Building volume and dollar bars

Now that we have seen the strengths and weaknesses of each bar type, let's look at how can we actually build them.

Here's a fast Python implementation to build volume bars:

```python
import numpy as np

# expects a numpy array with trades
# each trade is composed of: [time, price, quantity]
def generate_volumebars(trades, frequency=10):
    times = trades[:,0]
    prices = trades[:,1]
    volumes = trades[:,2]
    ans = np.zeros(shape=(len(prices), 6))
    candle_counter = 0
    vol = 0
    lasti = 0
    for i in range(len(prices)):
        vol += volumes[i]
        if vol >= frequency:
            ans[candle_counter][0] = times[i]                       # time
            ans[candle_counter][1] = prices[lasti]                  # open
            ans[candle_counter][2] = np.max(prices[lasti:i+1])      # high
            ans[candle_counter][3] = np.min(prices[lasti:i+1])      # low
            ans[candle_counter][4] = prices[i]                      # close
            ans[candle_counter][5] = np.sum(volumes[lasti:i+1])     # volume
            candle_counter += 1
            lasti = i+1
            vol = 0
    return ans[:candle_counter]
```

volumebar_generator.py hosted with ❤️ by GitHub view raw

Snippet 1. Volume bar implementation

```python

# expects a numpy array with trades
# each trade is composed of: [time, price, quantity]
def generate_dollarbars(trades, frequency=1000):
    times = trades[:,0]
    prices = trades[:,1]
    volumes = trades[:,2]
    ans = np.zeros(shape=(len(prices), 6))
    candle_counter = 0
    dollars = 0
    lasti = 0
    for i in range(len(prices)):
        dollars += volumes[i]*prices[i]
        if dollars >= frequency:
            ans[candle_counter][0] = times[i]                       # time
            ans[candle_counter][1] = prices[lasti]                  # open
            ans[candle_counter][2] = np.max(prices[lasti:i+1])      # high
            ans[candle_counter][3] = np.min(prices[lasti:i+1])      # low
            ans[candle_counter][4] = prices[i]                      # close
            ans[candle_counter][5] = np.sum(volumes[lasti:i+1])     # volume
            candle_counter += 1
            lasti = i+1
            dollars = 0
    return ans[:candle_counter]
```

dollarbar_generator.py hosted with ❤️ by GitHub view raw

Snippet 2. Dollar bar implementation

Finally, here's how they look compared to traditional time-based candlesticks:

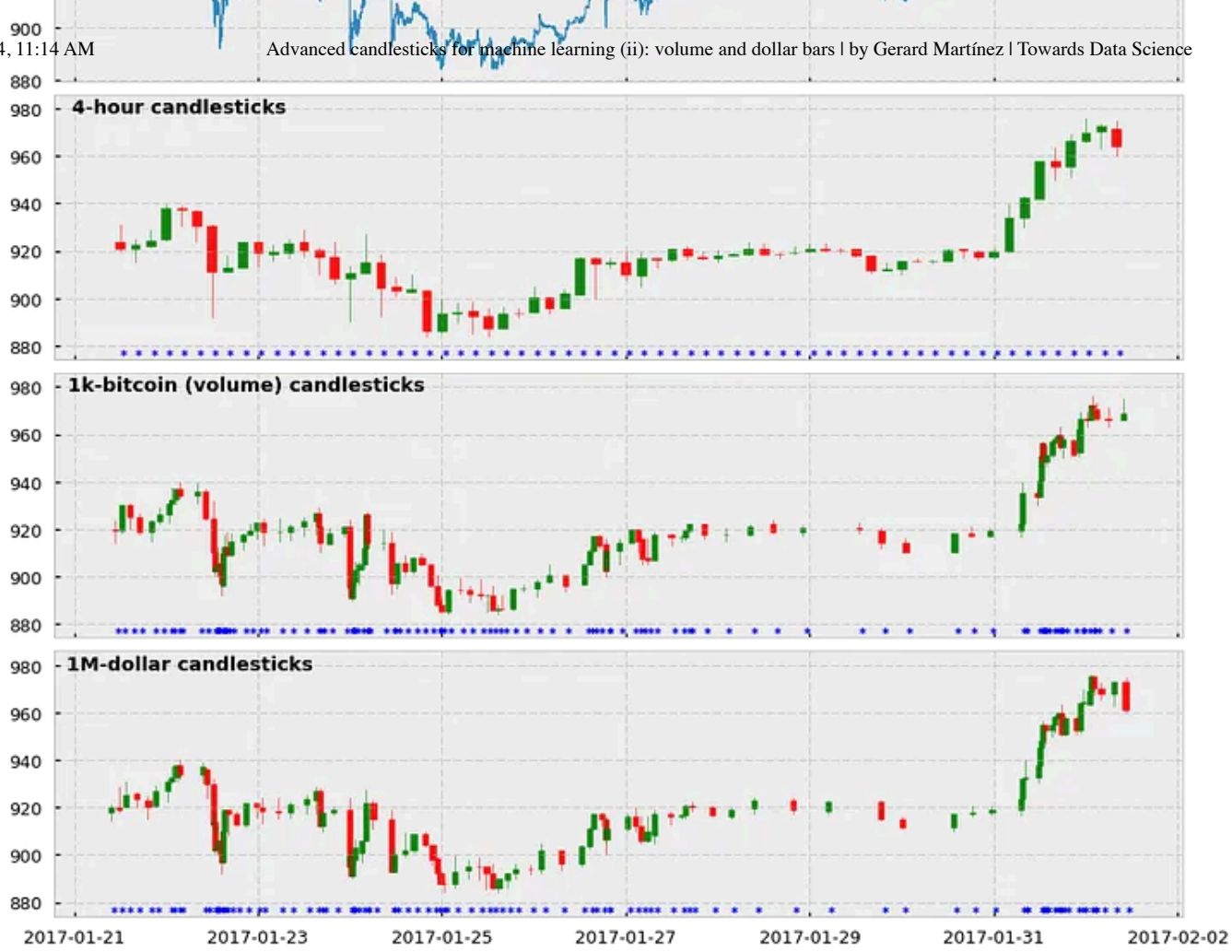


Figure 1. Trade, time-based, volume and dollar bars depiction. Asterisks are plotted every time a candle is sampled

Volume and dollar-based candles are similar to the tick bars in the sense that, while is true that they look chaotic and overlapped when compared to the harmonic time-based ones, they do their job well at sampling whenever there is a change in market activity.

Statistical analysis of volume bars

Let's now look at their statistical properties. We'll be looking at serial correlation of returns by performing the Pearson auto-correlation test and the Durbin-Watson test. Finally, we will also look at the normality of results by performing the Jarque-Bera and the Shapiro-Wilk tests. Refer to the old article about tick bars to learn more about these statistical tests.

sizes for each cryptocurrency was by calculating the mean volume exchanged per day and then by dividing the daily mean volume by the same ratios as 5min, 15min, 30min, 1h, 4h, 12h correspond to 1d, and rounded to the nearest 10. Here are the automatically chosen volumes per cryptocurrency pair:

Table 1. CryptoDatum.io volume bar sizes

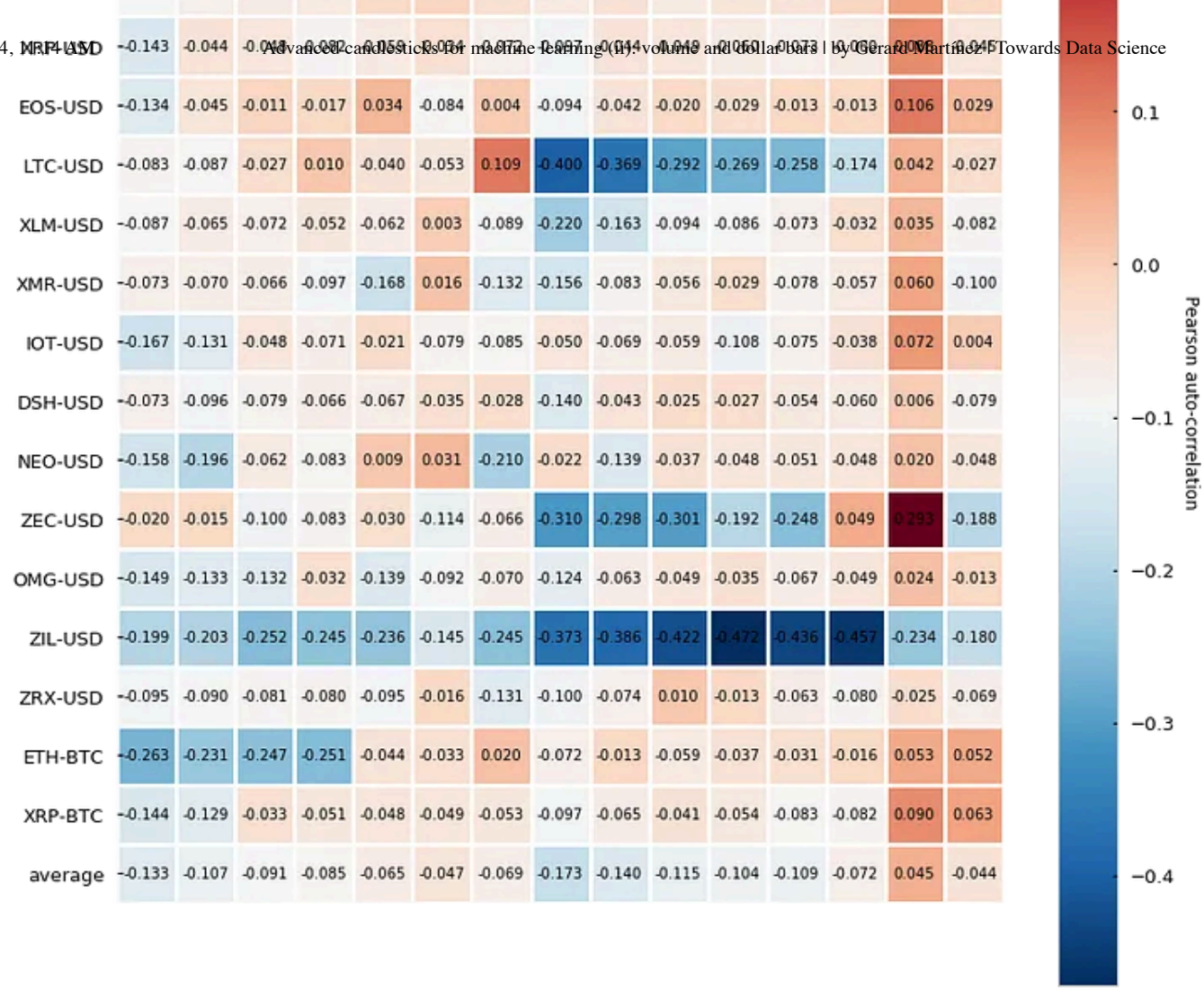


Figure 2. Pearson auto-correlation for volume bars

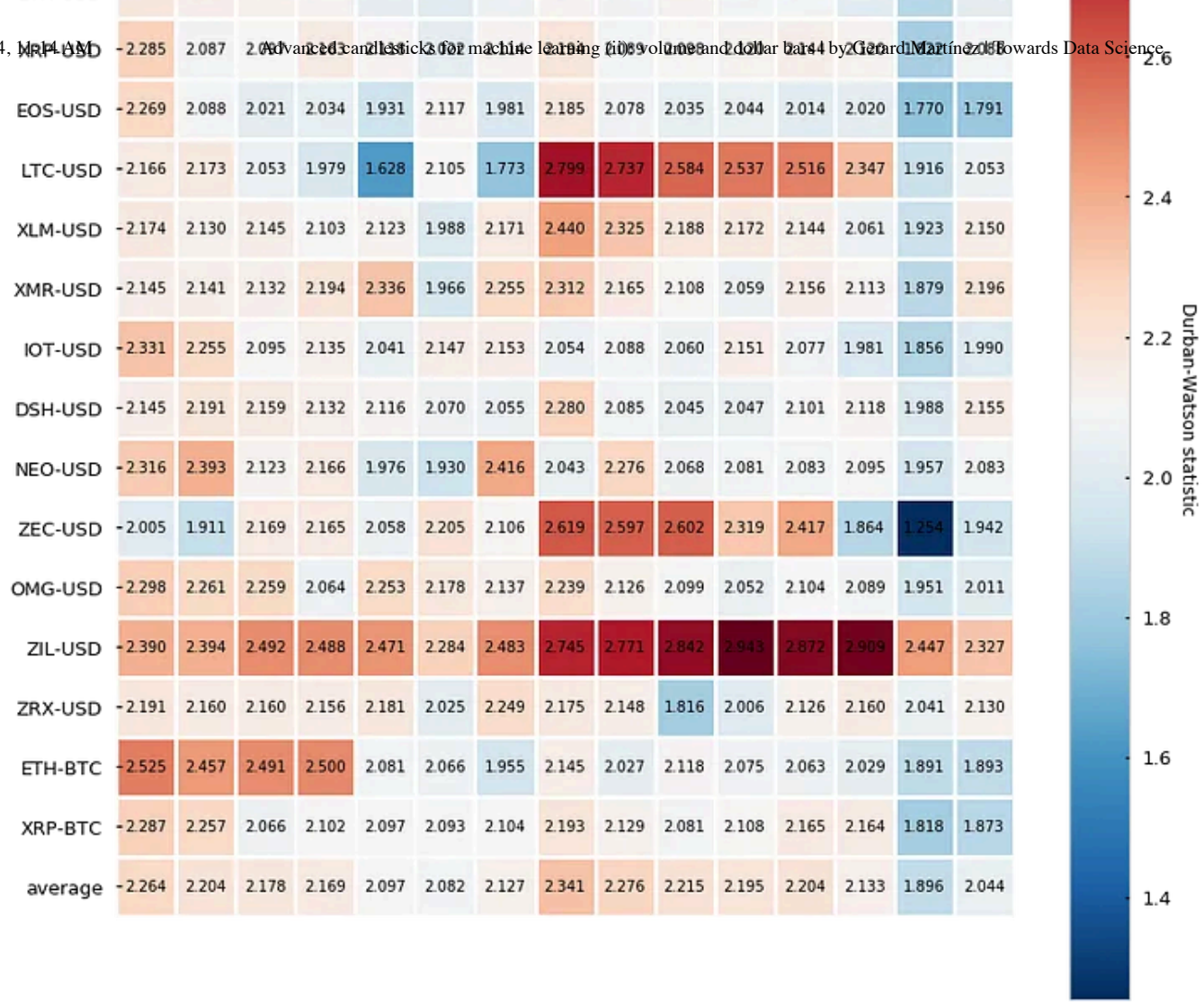


Figure 3. Durban-Watson statistic for volume bars

Results are in line with what we saw with tick bars. Volume bars show slightly less auto-correlation in comparison to time-based candlesticks (Figure 2 and 3) and the null hypothesis of normality is rejected in most cases in both normality tests (Figure 4) so we have certainty that returns do not follow a Gaussian distribution.

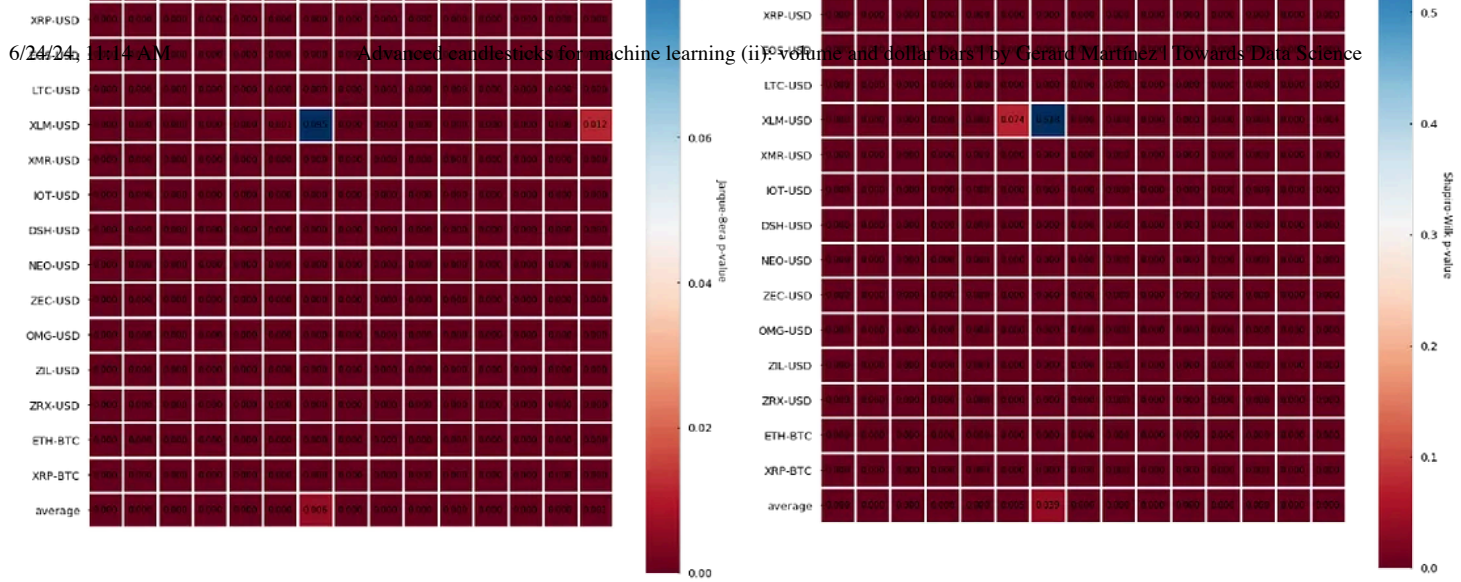


Figure 4. Jarque-Bera and Spahiro-Wilk tests p-value for volume bars

Statistical analysis of dollar bars

We'll repeat the previous tests for autocorrelation and normality of returns. In the case the dollar bars, the dollar bar sizes (expressed in $) are defined as in the next table:

Table 2. CryptoDatum.io dollar bar sizes ($)

Let’s look now at the normality and autocorrelation tests results:

Figure 5. Pearson auto-correlation for dollar bars

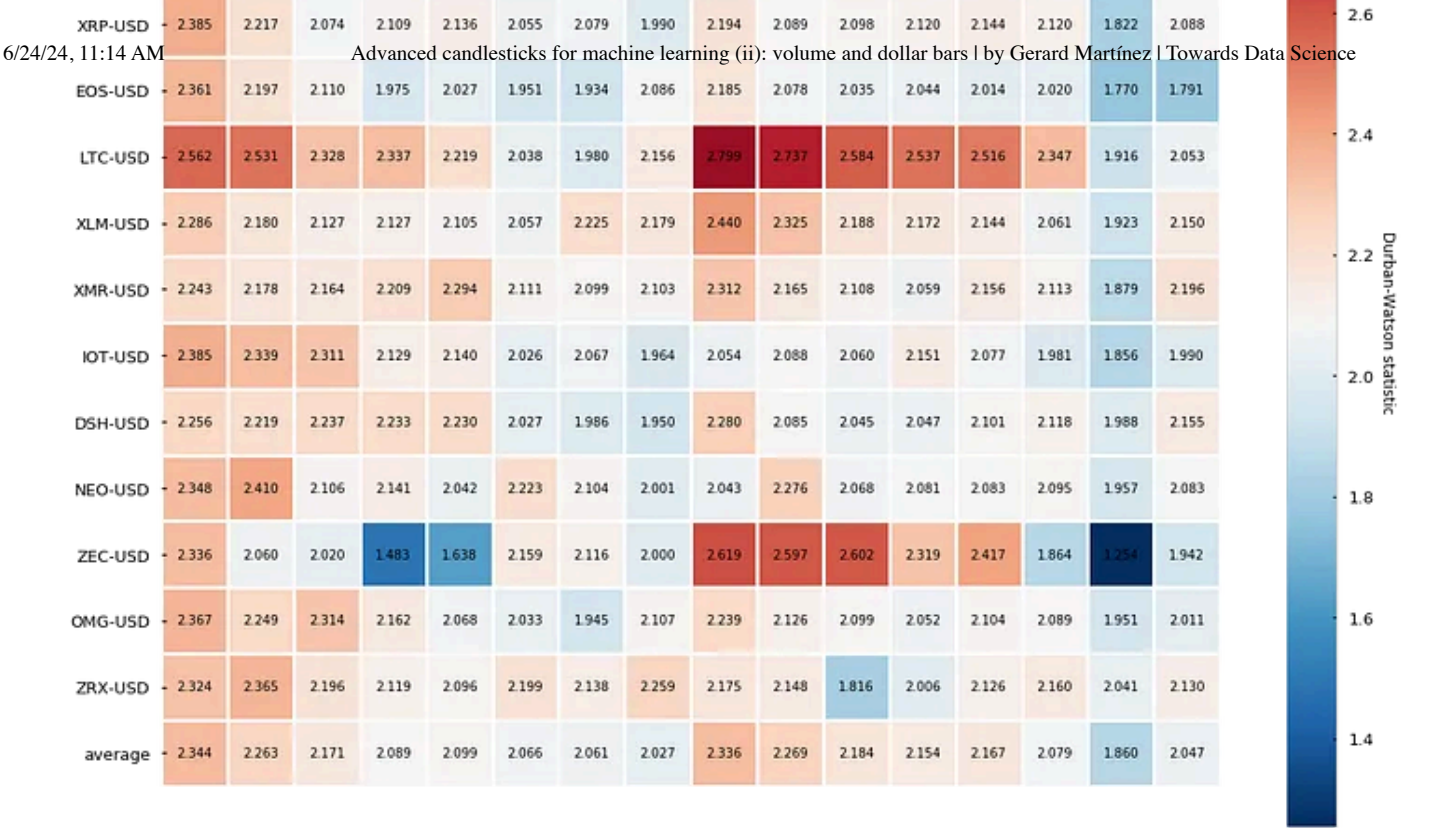


Figure 6. Durbin-Watson test results for dollar bars

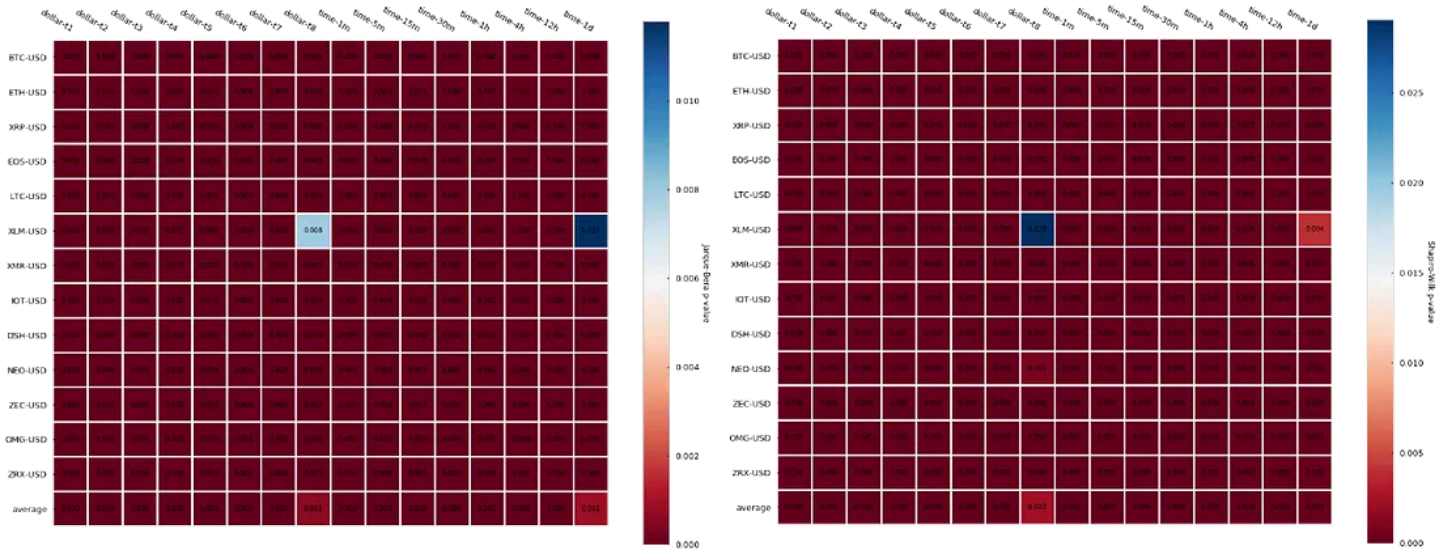


Figure 7. Normality tests for dollar bars

The results, contrary to what we have seen for both tick and volume bars is that dollar bars in fact show little improvement in terms of autocorrelation when compared to time-based candlesticks. We can observe a slightly lower

We have said that tick, volume and dollar bars sampling adapt to the market activity and allow us to sample more in high activity periods and sample less in low activity period. Ultimately, this property should be reflected in the amount of price change inside a single candle. The idea is that time-based candlesticks sample at fix time intervals regardless of the market activity while the alternative bars get synchronized with market activity. Therefore, seems intuitive that time-based candlesticks should have both candles with little price change (periods of oversampling) and candles with high price change (periods of undersampling), while alternative bars should be more equilibrated.

However, what do we understand as market activity? Can a market be very active and the price move sideways? The answer is yes, while generally there's a correlation between high market activity and big price change this relationship is not guaranteed. For instance, there are cases in which high FUD (fear, uncertainty and doubt) can provoke a big spike in volume but little price change. Also wash trading is definitely not uncommon in cryptocurrency markets. These moves involve self trading (the buyer and the seller are the same person), which again provokes huge spikes of volume with little price change.

In order clear this matter out, let's look at the distributions of intra-candle price variation for each of the type of bars. Intra-candle price variation has been calculated as:

> *intra-candle variation = (high-low)/high*

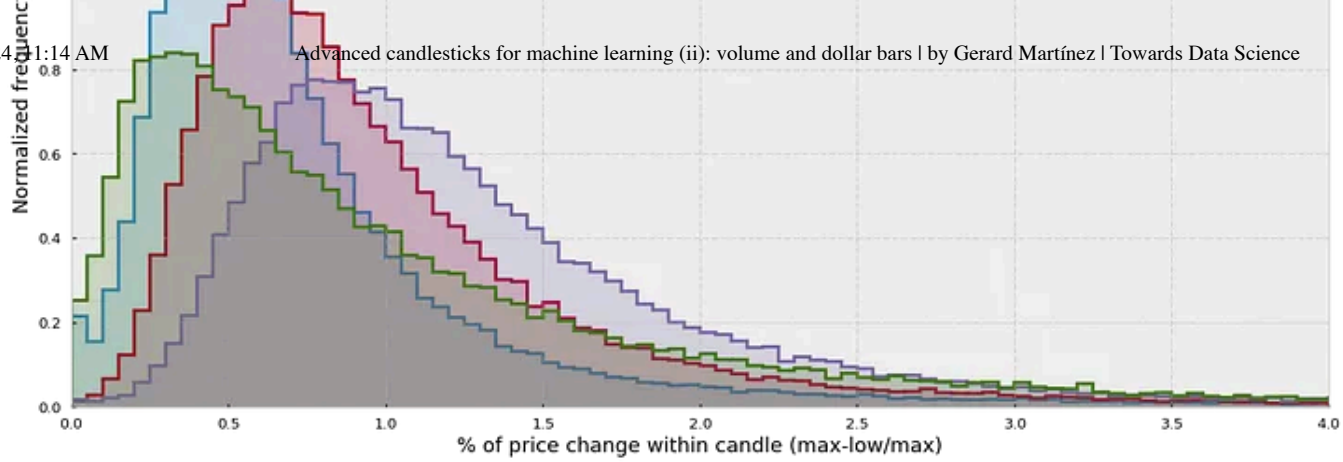
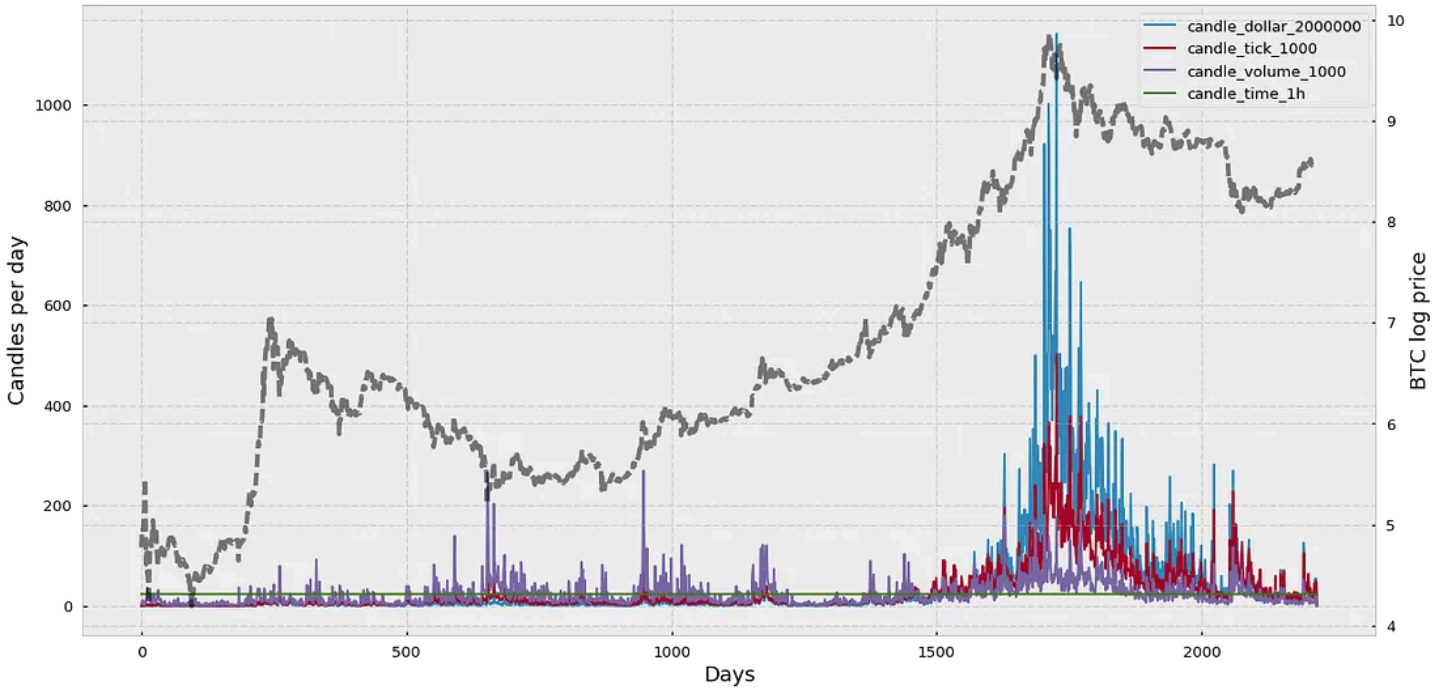


Figure 8. Intra-candle price variation for all the history of the BTC-USD pair at the Bitfinex exchange (data source: CryptoDatum.io)

We can see that time-candlesticks distribution is more displaced towards the 0 and has a long tail to the right, while the other distributions effectively sample more in periods of high price variation.

Daily bar frequency

Before ending the article, I would like to look at the daily frequency of bars and how the price of the traded asset and the activity in the market affect the sampling rate. Let's see the results:



per day) and that the sampling rate of the alternative bars varies according to changes in the bitcoin price (shown in grey).

Interestingly, among the alternative bars, volume bars sampling frequency seems to remain the most stable across all BTC-USD history — only if we include the all-time-high of 2017, otherwise dollar or tick bars seem to be more stable. Another feature that surprises me is that dollar candles, which are supposed to be more stable because they sort of "correct" for the actual asset price, seem to get out of control during the all time high of 2017. I suspect that this is because dollar bars work well at correcting revaluations of the asset during more or less stable volumes. However, if you combine the fact that the asset price gets multiplied 10 times or more and the total volume, instead of decreasing due to the high cost of the asset, keeps increasing, then you encounter a situation when dollar bar sampling simply explodes.

What have we learned?

- Volume bars address the tick bars limitation regarding multiple low sized trades and iceberg orders by only focusing on the total amount of asset exchanged.
- Dollar bars measure the fiat value exchanged, which ideally corrects for the fluctuating value of an asset such as in cryptocurrencies.
- Both volume and dollar bars sample more when activity in the market increases and sample less when activity decreases.
- Volume bars display generally lower serial correlation than traditional time-based candlesticks.
- Dollar bars do not seem to yield lower serial correlation. However, keep in mind that the number of dollar bars seemed to explode during the all-time-high, which means that most dollar bars probably come from a very short period of time and therefore average autocorrelation may be higher simply because of the close "adjacency" of the bars.
- Both volume and dollar bar log returns do not follow a Gaussian distribution.

displaced towards the right.

- Volume bars sampling rate seems to be the most stable after fix-period time-based candlesticks.
- Dollar bars sampling rate seems to explode in bubbles with remarkably high volume.

Thanks for reading! In the next episode we will talk about one of the most interesting types of bars: imbalance bars. These bars are sampled based on the imbalance observed in the sequence of trades and are a good example of information-driven bars.

This project is part of our research at CryptoDatum.io, a cryptocurrency data API that aims to provide plug-and-play datasets to train machine learning algorithms. If you liked the data we showed in this article, get your free API key and play with it yourself at https://cryptodatum.io

Bitcoin | Algorithmic Trading | Cryptocurrency | Python | Towards Data Science

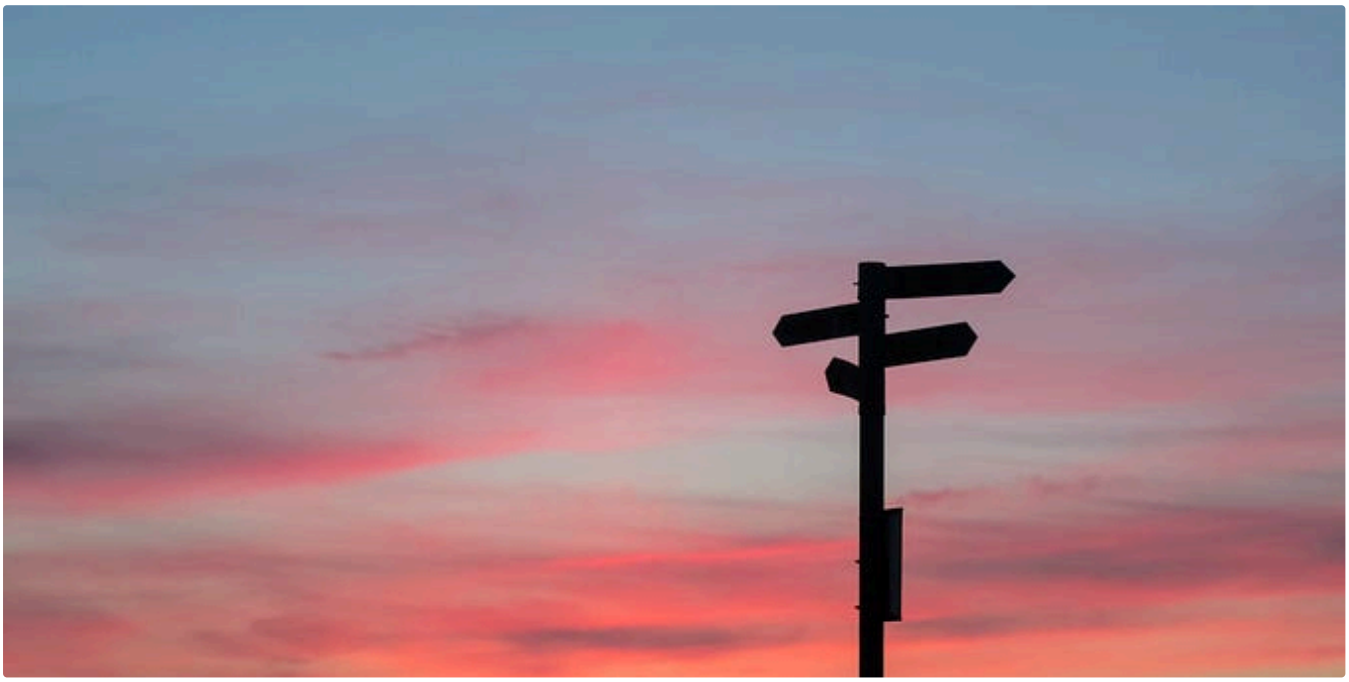
Follow

Written by Gerard Martínez

1.4K Followers · Writer for Towards Data Science

Trading strategy developer — Founder of CryptoDatum.io

More from Gerard Martínez and Towards Data Science



Gerard Martínez in Towards Data Science

Information-driven bars for financial machine learning: imbalance bars

In previous articles we talked about tick bars, volume bars and dollar bars, alternative types of bars which allow market...

8 min read · May 20, 2019

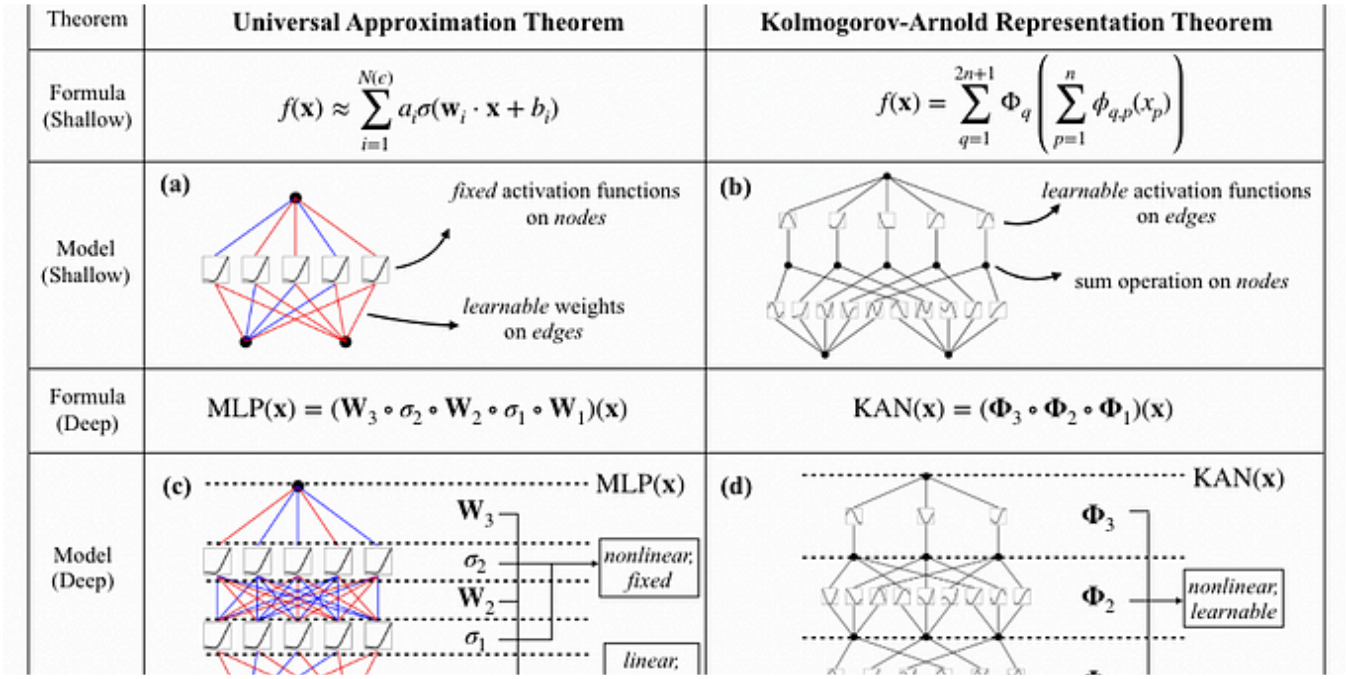
372 4

Torsten Walbaum in Towards Data Science

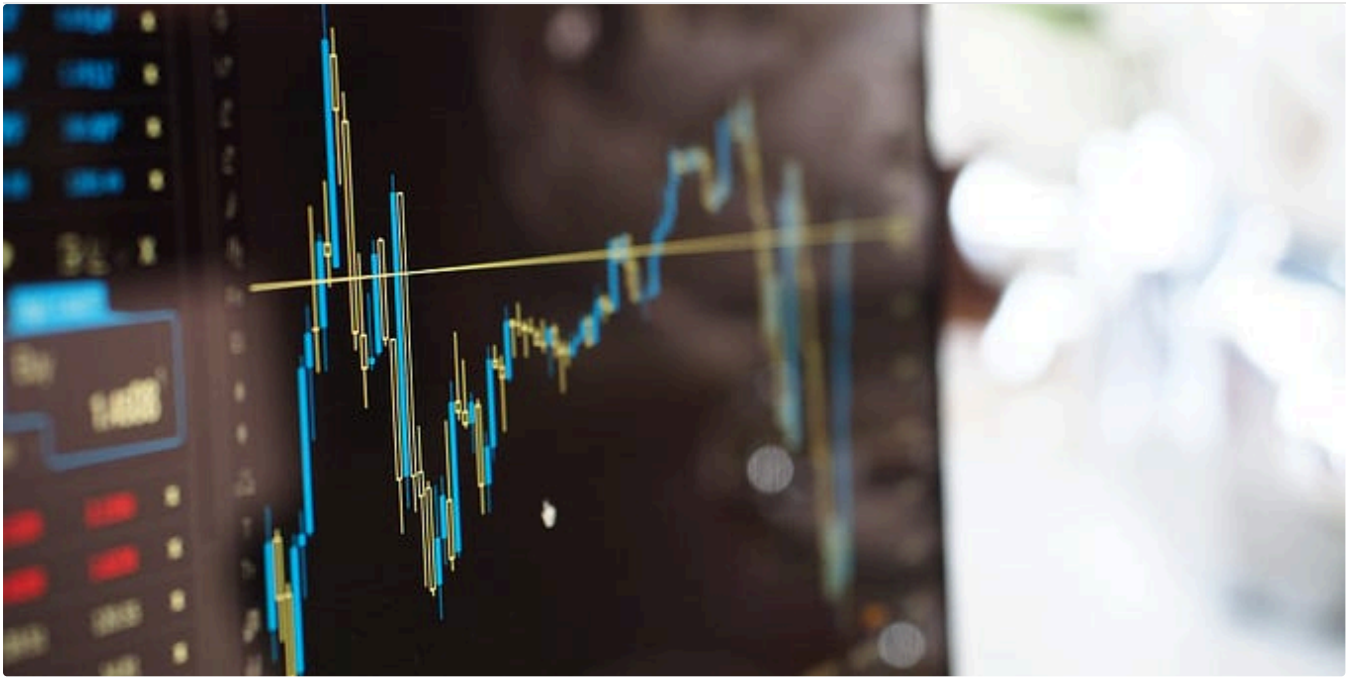
What 10 Years at Uber, Meta and Startups Taught Me About Data Analytics

Advice for Data Scientists and Managers

9 min read · May 30, 2024



Theo Wolf in Towards Data Science



PulsePointFX

Relative Strength Index Strategy using Python 🐍

Learn how to create a Relative Strength Index (RSI) trading strategy using Python programming language

6 min read · Jan 10, 2024

33

Anoob Paul

Pivot Points and Central Pivot Range(CPR) using python

Pivot points are price level that are used as part of our technical analysis to identify potential support and resistance. We can consider...

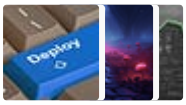
3 min read · Feb 28, 2024

21

Lists

Coding & Development

11 stories · 664 saves



Predictive Modeling w/ Python

20 stories · 1316 saves

Practical Guides to Machine Learning

10 stories · 1580 saves

ChatGPT

21 stories · 688 saves

Cristian Velasquez

Automating 61 Candlestick Trading Patterns in Python

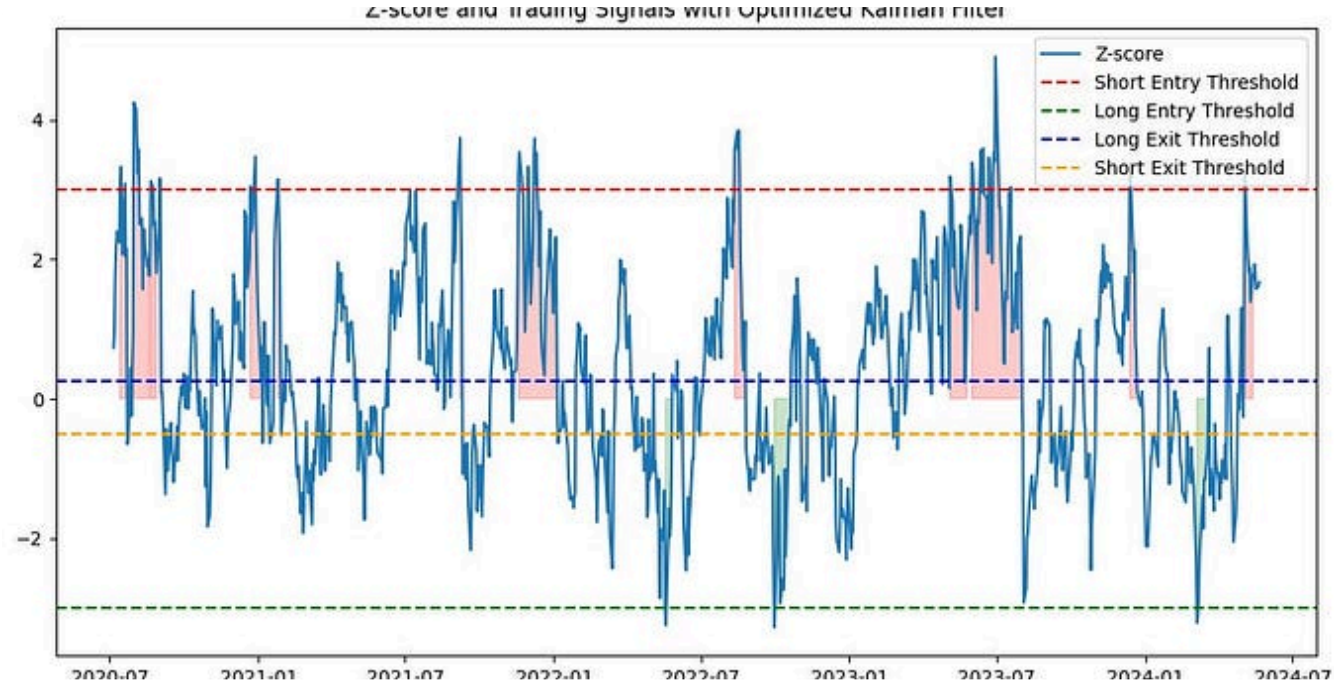
Towards Real-Time Automated Pattern Recognition using TA-Lib for Precision Pattern Scanning with Historical Accuracy Measures

· 28 min read · Feb 5, 2024

Ntale Geofrey

11 min read · Mar 21, 2024

193 5



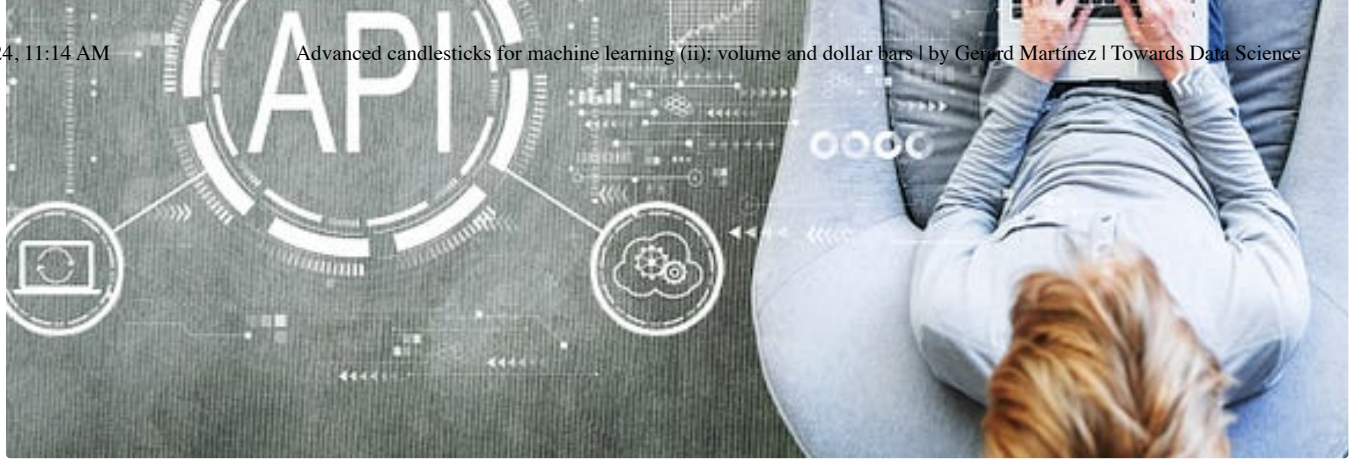
Serdar İlarslan

Implementing a Kalman Filter-Based Trading Strategy

Financial markets are inherently noisy and unpredictable, making it challenging for traders and investors to identify and capitalize on...

7 min read · May 25, 2024

163 2



EODHD APIs

Advanced Trading Strategies: Maximizing Profits with VWAP, TWAP, and PoV Using Python

This article explores advanced trading strategies—VWAP, TWAP, and PoV—demonstrating their practical application using Python and EODHD...

7 min read · Jun 13, 2024

151 1