Cryptocurrency Price Forecasting

DS 5110

December 13th,2021

Authors:

Bala Sirisha Sripathi Panditharadhyula
Amulya Gupta Vangapalli
Isha Hemanth Arora

Cryptocurrency Price Forecasting

Authors: Bala Sirisha Sripathi Panditharadhyula , Amulya Gupta Vangapalli, Isha Hemanth Arora

Summary:

Background of the Project:

Cryptocurrencies are digital tokens that, in the future, might easily replace traditional currency. The ease with which they may be accessed is one of the reasons for their rapid popularity. These coins are available to almost everyone and are accepted as payment in the same way that traditional currency is.

In the future, the blockchain technology on which most of these currencies are built, as well as their decentralized systems, could be used to create more safe and secure organizational environments. After the invention of Bitcoin as well as other blockchain-based peer-to-peer payment systems, the cryptocurrency market has rapidly gained popularity.

Consequently, the volatility of the various cryptocurrency prices attracts substantial attention from both investors and researchers and hence forecasting cryptocurrency prices is crucial for investors.

Description of Dataset:

The dataset that we used in this project includes one CSV file:

Data.csv The dataset includes data for approximately 5 years with around 2071 types of cryptocurrencies

	slug	symbol	name	date	ranknow	open	high	low	close	volume	market	close_ratio	spread
0	bitcoin	BTC	Bitcoin	2013-04-28	1	135.30	135.98	132.10	134.21	0.0	1.488567e+09	0.5438	3.88
1	bitcoin	BTC	Bitcoin	2013-04-29	1	134.44	147.49	134.00	144.54	0.0	1.603769e+09	0.7813	13.49
2	bitcoin	BTC	Bitcoin	2013-04-30	1	144.00	146.93	134.05	139.00	0.0	1.542813e+09	0.3843	12.88
3	bitcoin	BTC	Bitcoin	2013-05-01	1	139.00	139.89	107.72	116.99	0.0	1.298955e+09	0.2882	32.17
4	bitcoin	BTC	Bitcoin	2013-05-02	1	116.38	125.60	92.28	105.21	0.0	1.168517e+09	0.3881	33.32

It consists of about 942 thousand observations. We have data focused on every date and type of cryptocurrency. Other attributes are opening bid, highest bid, lowest bid, number of transactions, market capitalization.

Project Goals:

In today's emerging world, cryptocurrency has evolved and grown to great extents and understanding such complex trends is extremely difficult and unpredictable. Therefore, analysis of cryptocurrency with precise predictions is important to keep up with the developing industry. In this project we aim to forecast the closing price of cryptocurrencies.

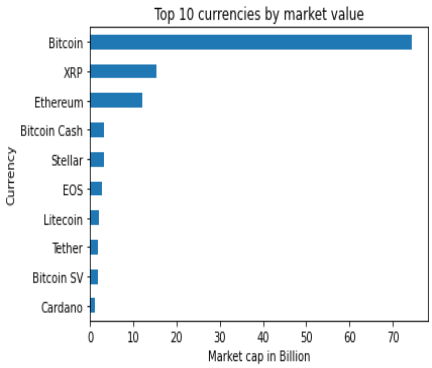
Description of methods and results:

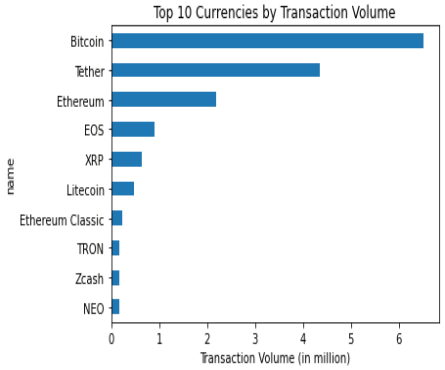
Time-series analysis can predict the future ups and downs in the price of Bitcoin. Our methods for this purpose include developing time-series forecasting models using LSTM (Long Short-Term Memory); and classical models using Decision Trees, Random Forest and XGBoost and determine if our dataset works best by a time-series approach or a regression approach.
The parameters on the basis of which we have evaluated these models are Root Mean Square Error (RMSE) and Mean Absolute Error (MAE).

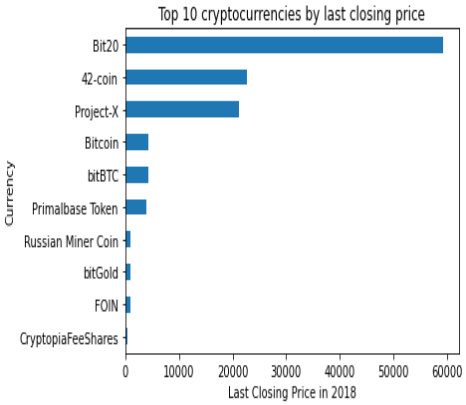
Methods:

(a) Exploratory Data Analysis:

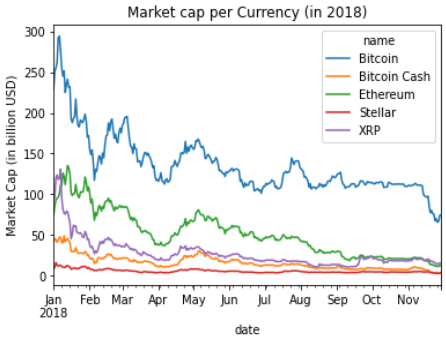
Exploratory Data Analysis is an important step in any project to analyze and understand the data and the problem better.

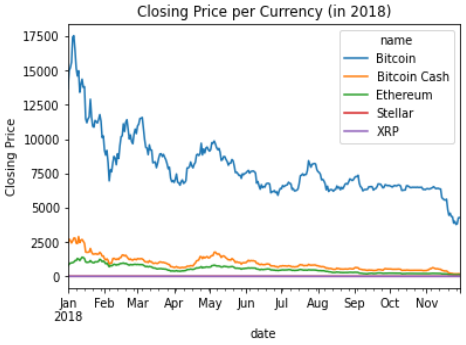


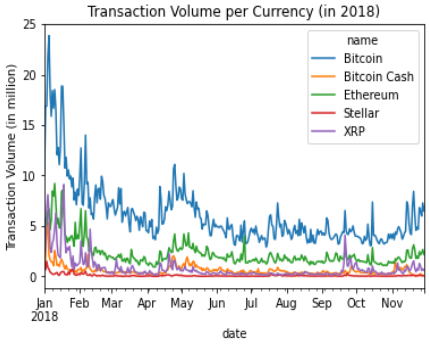




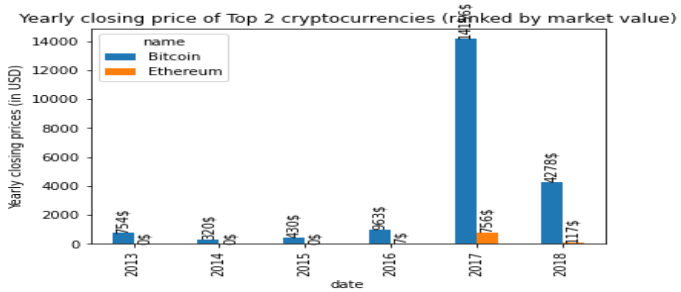
From the above graphs it can be inferred that Bitcoin has been the top currency as per market value, transaction volume and closing price. Ethereum has considerably high market value and transaction volume. For all other cryptocurrencies, the trend varies.

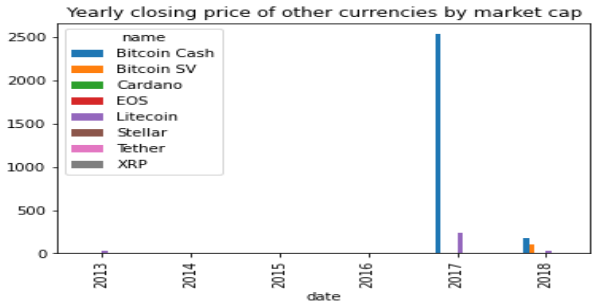


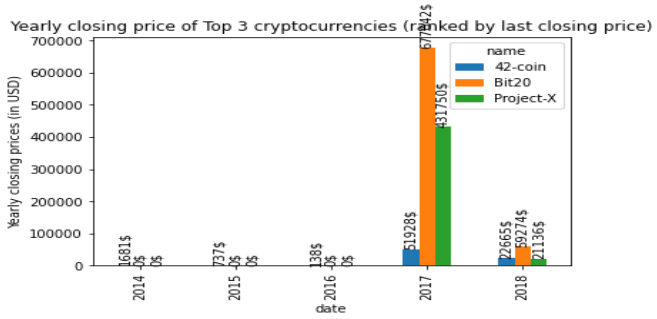


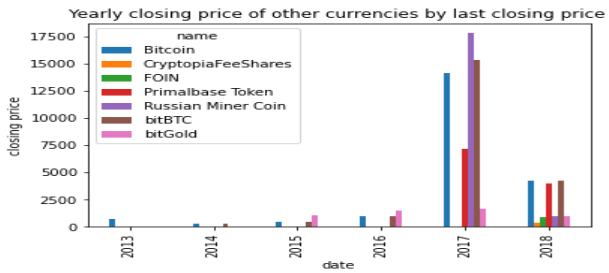


From the above graphs (as plotted for the last year of data we have), it can be inferred that in the month of January for Bitcoin, the market cap, closing price and transaction volume was the highest. This trend seems to be declining when we move towards the end of the year. For Stellar and XRP, the trend seems to be almost constant throughout the year. For all the cryptocurrencies the highest point is in the month of January while the least in November for market cap and closing price. Ethereum has a decreasing market cap, constant closing price and a fluctuating transaction volume. Inspite of its fluctuating trend as per transactional volume, Ethereum manages to stay in the top 3 cryptocurrencies by volume.

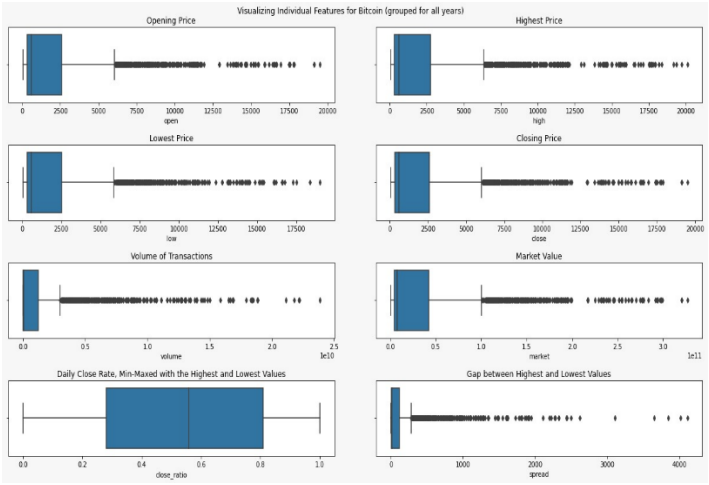








The closing price of bitcoin and Ethereum were extremely low from the years 2013 – 2016 and hit a peak in the year 2017 for bitcoin and Ethereum. From the years 2013- 1016, the popularity of cryptocurrency was not dominant and hence less closing prices are observed. Even though Bitcoin hit a high closing price in 2017, it dropped by around by 50% in 2018.
From the year 2017 all the other cryptocurrencies came into picture and were circulated in the market with a decent market cap.
When sorted by last closing price, the top 3 currencies were found to be 42coin, bit20 and project which had their highest closing price in the year 2017. They reduced drastically in the next year. Closing price of all the other currencies also were high in 2017 and reduced in 2018.
From all these graphs it can be inferred that cryptocurrencies were not prevalent till 2017.

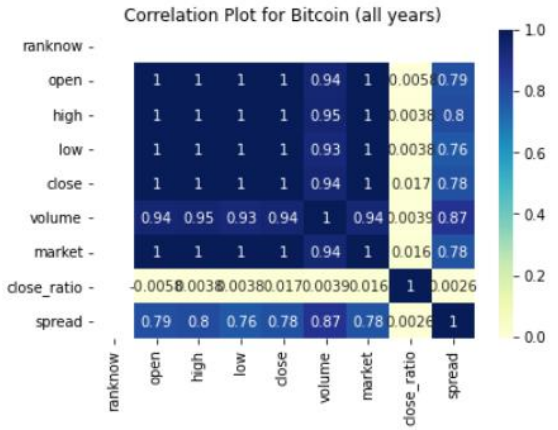


From the graphs above it can be said that except for the year 2014, the high price of bitcoin has always increased along the months for each year. For the low price, an increasing trend can be observed in all years except 2013 and 2014. The highest price of bitcoin was in the year 2017 while the lowest was in 2013.
From the boxplots, we can observe that there are no pressing outliers in any of the data used as predictors and hence no outlier removal is needed.

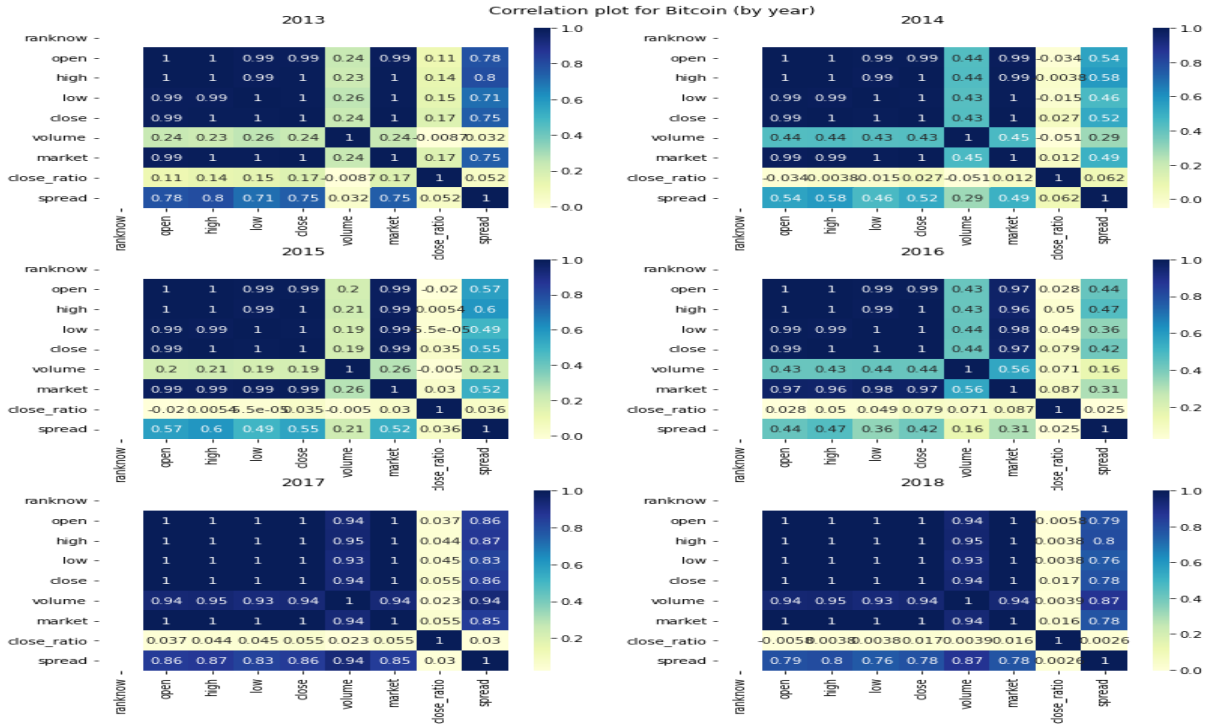
(b) Data pre-processing and Feature Engineering:

Data pre-processing involves cleaning the datasets and removing missing values, splitting the dataset into test and train. Feature Engineering is the next essential step in data preparation as it helps us to take into the account the most important features in the data and perform any necessary transformations to prepare the data for modeling.

The first step was to check for Null Values in the dataset. On checking, no null values in any of the columns were found. The next step was to check the datatypes of the features. A change was made to the date column and changed into the datetime datatype.



From our observations, the "close ratio" parameter didn't have any significant correlation with any other features, hence we decided to drop it in the initial stages itself. Next, we have plotted the correlation matrix for bitcoin for all the years from 2013 to 2018.

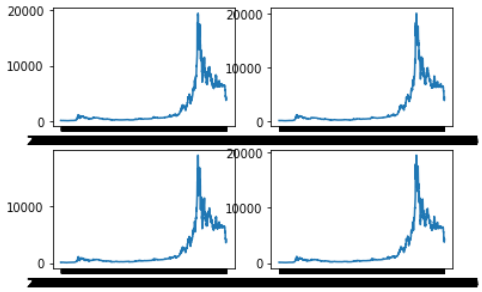


Volume had very less impact on closing price from 2013 to 2016, but in the years 2017 and 2018, it dramatically had a massive correlation with closing price. This can account as a potential supporter for the bitcoin's huge surge that occurred on Dec 7th 2017. *"A dramatic increase in bitcoin trading volume occurred at around 11 am EST (or 16:00 UTC), a period which coincides with bitcoin's skewed price movements on December 7"* states the cryptocurrency news.

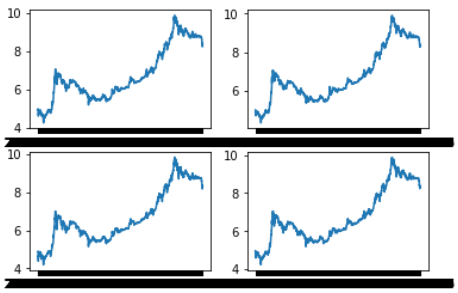
"Market" had a very less correlation with volume till 2016 but in 2017 and 2018, the market capitalization rose due to the huge increase in volume (as market cap is calculated by multiplying the price of bitcoin with total number of coins in circulation) which can be seen in the correlation plot as well. "Open", "High" and "Close" have high correlation throughout all the years. Removing them also seemed to reduce the accuracy and hence these parameters stayed intact.

In further steps, we have applied log transformations on the parameters as a part of our preprocessing.

Before Transformations



After Transformations



We can observe that before transformations were applied, the data was left skewed and hence transformations are applied.

Models:

1. LSTM: Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems. LSTMs feature gate mechanisms for regulating information and a cell state for preserving information about prior predictions over lengthy periods of time. As a result, LSTMs are useful for modeling sequence data and work well for time series forecasting. We used LSTMs to try to forecast the meter reading at the current timestamp based on the past 10 observations to see how well these models work. The data was divided into two groups: 80 percent train data and 20% test data. We got an RMSE of 24.80 and an MAE of 4.66, which were both below average.

2. Decision Tree Regressor: A decision tree regressor is a supervised machine learning model used to predict a target by learning decision rules from features. Decision tree builds regression model in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed.

3. Random Forest Regressor: A random forest is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

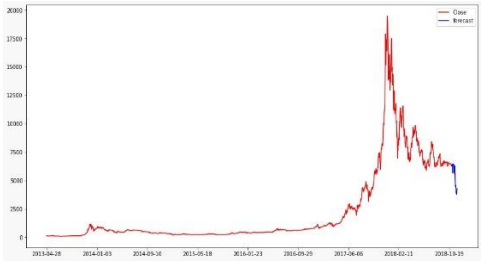
4. XGBoost regressor: eXtreme Gradient Boosting is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework.

Results:

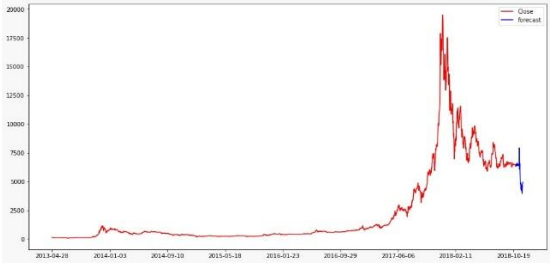
The following are the results before the transformations were applied.

Models	RMSE	MAE
Decision Tree	1648.522	565.522
Random Forest	1060.035	391.13
XGBoost	1392.609	707.74
LSTM	980.544	520.101

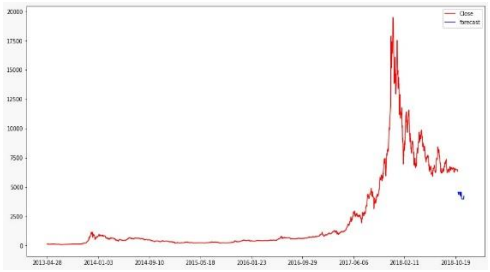
Decision Tree without transformations on features



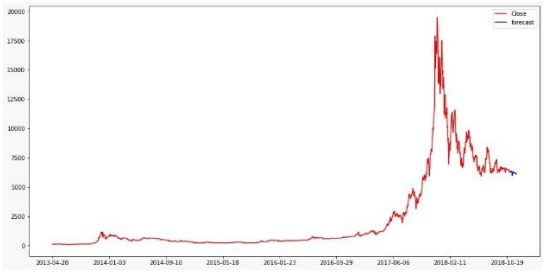
Random Forest without transformations on features



XGBoost without transformations on features



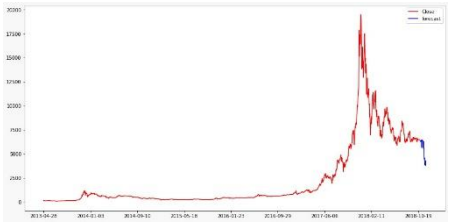
LSTM without transformations on features



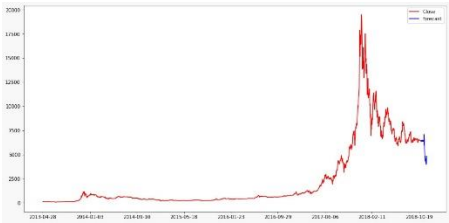
The following are the results after the transformations were applied.

Models	RMSE	MAE
Decision Tree	0.314	0.203
Random Forest	0.222	0.155
XGBoost	2.247	2.187
LSTM	0.325	0.223

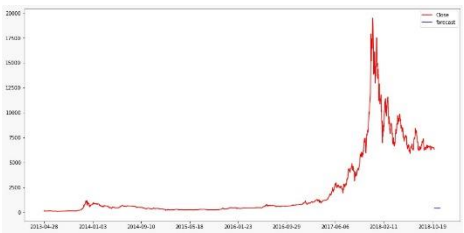
Decision Tree after transformations on features



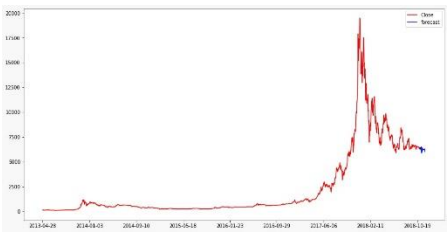
Random Forest after transformations on features



XGBoost after transformations on features



LSTM after transformations on features



Discussions:

It can easily be noticed that cryptocurrencies have become prevalent after the year 2017. In 2017, as Bitcoin was the one first introduced to the audience, it was the most common type and quickly gained immense popularity.
After the year 2017, we can see that a lot new cryptocurrencies were introduced into the market which removed the lime light of bitcoin and therefore resulted in the audience branching out to try and use varied cryptocurrencies.
This project aimed at predicting the closing price for the cryptocurrency - Bitcoin for the next 30 days. We thus believe that anyone who is interested in investing in the market would be benefitted.
In the future we intend to use more complex time series models like ARIMA and SARIMAX since we noticed a seasonal decomposition.
We hope that doing this would help us gain a better accuracy for predictions and thus extend our work for seasoned investors.
We also believe that these models can be extended to any other future cryptocurrencies which may come up.

Statement of Contribution:

Bala Sirisha Sripathi Panditharadhyula : EDA and Feature Engineering, LSTM
Amulya Gupta Vangapalli : EDA and Feature Engineering, XGBoost
Isha Hemanth Arora : EDA and Feature Engineering, Decision Tree and Random Forest

References:

1.https://www.geeksforgeeks.org/python-decision-tree-regression-using-sklearn/
2.https://towardsdatascience.com/understanding-gradient-boosting-machines-9be756fe76ab
3.https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-py thon-keras/
4.https://towardsdatascience.com/cryptocurrency-price-prediction-using-deep-learning-70cfca50dd3a
5.https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html
6.https://www.geeksforgeeks.org/random-forest-regression-in-python/
7. https://towardsdatascience.com/getting-started-with-xgboost-in-scikit-learn-f69f5f470a97

Appendix:

```
top10Currencies_lc = df.groupby('name')['close'].last().sort_values(ascending=False).head(10)
ax = top10Currencies_lc.sort_values().plot(kind='barh')
ax.set_xlabel("Last Closing Price in 2018");
ax.set_ylabel("Currency");
plt.title("Top 10 cryptocurrencies by closing price");
```

```
top10Currencies_mc = df.groupby('name')['market_billion'].last().sort_values(ascending=False).head(10)
ax = top10Currencies_mc.sort_values().plot(kind='barh')
ax.set_xlabel("Market cap in Billion");
ax.set_ylabel("Currency");
plt.title("Top 10 currencies by market value");
```

```
ax = data_top_5_currencies[data_top_5_currencies.date.dt.year >= 2018].groupby(['date', 'name'])['market_billion'].mean().unstack().plot();
ax.set_ylabel("Market Cap (in billion USD)");
plt.title("Market cap per Currency (in 2018)");
```

```
ax = data_top_5_currencies[data_top_5_currencies.date.dt.year >= 2018].groupby(['date', 'name'])['volume_million'].mean().unstack().plot();
ax.set_ylabel("Transaction Volume (in million)");
plt.title("Transaction Volume per Currency (in 2018)");
```

```
ax = data_top_5_currencies[data_top_5_currencies.date.dt.year >= 2018].groupby(['date', 'name'])['close'].mean().unstack().plot();
ax.set_ylabel("Closing Price");
plt.title("Closing Price per Currency (in 2018)");
```

```
yearly_closing_prices_bitcoin_and_ethereum = closing_prices_bitcoin_and_ethereum.groupby(closing_prices_bitcoin_and_ethereum.date.dt.year).last()
yearly_closing_prices_bitcoin_and_ethereum.drop(columns='date', inplace=True)
plot_with_textvalue(yearly_closing_prices_bitcoin_and_ethereum)
plt.title("Yearly closing price of Top 2 cryptocurrencies (ranked by market value)")
```

```
years = mdates.YearLocator()   # every year
months = mdates.MonthLocator()  # every month
years_fmt = mdates.DateFormatter('%Y')

fig, ((ax1, ax2), (ax3, ax4)) = plt.subplots(2,2)
ax1.plot('date', 'open', data=data_bitcoin)
ax2.plot('date', 'high', data=data_bitcoin)
ax3.plot('date', 'low', data=data_bitcoin)
ax4.plot('date', 'close', data=data_bitcoin)

fig, ((ax1, ax2), (ax3, ax4)) = plt.subplots(2,2)
ax1.plot('date', 'open_log', data=data_bitcoin)
ax2.plot('date', 'high_log', data=data_bitcoin)
ax3.plot('date', 'low_log', data=data_bitcoin)
ax4.plot('date', 'close_log', data=data_bitcoin)
```

```
data_bitcoin['open_log'] = np.log1p(data_bitcoin['open'])
data_bitcoin['high_log'] = np.log1p(data_bitcoin['high'])
data_bitcoin['low_log'] = np.log1p(data_bitcoin['low'])
data_bitcoin['close_log'] = np.log1p(data_bitcoin['close'])
```

```
fig, axes = plt.subplots(3,2, figsize=(12,12))

fig.tight_layout(pad=4)
fig.suptitle("Correlation plot for Bitcoin (by year)")

sb.heatmap(data_bitcoin_2013.corr(), cmap="YlGnBu", annot=True, ax=axes[0,0]).set(title='2013')
sb.heatmap(data_bitcoin_2014.corr(), cmap="YlGnBu", annot=True, ax=axes[0,1]).set(title='2014')
sb.heatmap(data_bitcoin_2015.corr(), cmap="YlGnBu", annot=True, ax=axes[1,0]).set(title='2015')
sb.heatmap(data_bitcoin_2016.corr(), cmap="YlGnBu", annot=True, ax=axes[1,1]).set(title='2016')
sb.heatmap(data_bitcoin_2017.corr(), cmap="YlGnBu", annot=True, ax=axes[2,0]).set(title='2017')
sb.heatmap(data_bitcoin_2018.corr(), cmap="YlGnBu", annot=True, ax=axes[2,1]).set(title='2018')
```

```
import plotly.graph_objects as go

fig, ((ax1, ax2), (ax3, ax4), (ax5, ax6)) = plt.subplots(3,2, figsize = (8,8))

fig.tight_layout(pad=4)
fig.suptitle("High vs Low prices for Bitcoin (by year)")

y_2013 = df_BTC.loc[(df_BTC['date'] >= '28-04-2013')
                    & (df_BTC['date'] < '01-01-2015')]
df_BTC['date'] = pd.to_datetime(df_BTC.date)
monthvise_2013= y_2013.groupby(y_2013['date'].dt.strftime('%B'))[['open','close']].mean()
new_order = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August',
             'September', 'October', 'November', 'December']
monthvise_2013 = monthvise_2013.reindex(new_order, axis=0)
monthvise_low_2013 = y_2013.groupby(y_2013['date'].dt.strftime('%B'))['low'].min()
monthvise_low_2013 = monthvise_low_2013.reindex(new_order, axis=0)

monthvise_high_2013 = y_2013.groupby(df['date'].dt.strftime('%B'))['high'].max()
monthvise_high_2013 = monthvise_high_2013.reindex(new_order, axis=0)

ax1.plot(monthvise_low_2013.index,monthvise_low_2013, label="low")
ax1.plot(monthvise_high_2013.index,monthvise_high_2013, label="high")
ax1.set_title('2013')
ax1.legend(loc='upper left')
ax1.tick_params(axis="x", direction="in", labelrotation = 90)


y_2014 = df_BTC.loc[(df_BTC['date'] >= '01-01-2014')
                    & (df_BTC['date'] < '01-01-2015')]
df_BTC['date'] = pd.to_datetime(df_BTC.date)
monthvise_2014= y_2014.groupby(y_2014['date'].dt.strftime('%B'))[['open','close']].mean()
monthvise_2014 = monthvise_2014.reindex(new_order, axis=0)
monthvise_low_2014 = y_2014.groupby(y_2014['date'].dt.strftime('%B'))['low'].min()
monthvise_low_2014 = monthvise_low_2014.reindex(new_order, axis=0)

monthvise_high_2014 = y_2014.groupby(df['date'].dt.strftime('%B'))['high'].max()
monthvise_high_2014 = monthvise_high_2014.reindex(new_order, axis=0)

ax2.plot(monthvise_low_2014.index,monthvise_low_2014, label="low")
ax2.plot(monthvise_high_2014.index,monthvise_high_2014, label="high")
ax2.set_title('2014')
ax2.legend(loc='upper left')
ax2.tick_params(axis="x", direction="in", labelrotation = 90)
```