notebook

January 5, 2022

1 Should your fund invest in Bitcoin?

1.1 Background

You work as an analyst at an investment fund in New York. Your CFO wants to explore if it is a good idea to invest some of the fund's assets in Bitcoin. You have to prepare a report on this asset and how it compares to the stock market in general.

2 Should your fund invest in Bitcoin?

2.1 Background

You work as an analyst at an investment fund in New York. Your CFO wants to explore if it is a good idea to invest some of the fund's assets in Bitcoin. You have to prepare a report on this asset and how it compares to the stock market in general.

2.2 The data

You have access to three files:

Bitcoin daily data in US dollars

- "date" - date from September 17, 2014 to November 17, 2021
- "open" - the price at the beginning of the trading day
- "high" - the highest price reached that day
- "low" - the lowest price reached that day
- "close" - the price at the closing of the trading day
- "volume" - how many Bitcoin were traded that day

S&P 500 daily data

- "date" - date from September 17, 2014 to November 17, 2021
- "open" - the index level at the beginning of the trading day
- "high" - the highest level reached that day
- "low" - the lowest level reached that day
- "close" - the level at the closing of the trading day
- "volume" - how many shares in the companies that make up the index were traded that day

inflation and gold as monthly data

- "date" - date from September, 2014 to November, 2021
- "gold_usd" - price in usd of gold for that month
- "cpi_us" - the inflation index for the US for that month (cpi = consumer price index)

CPI data from the U.S. Bureau of Labor Statistics. Publicly available information.

Exploratory analysis

[3]:
```python
import pandas as pd

monthly_data = pd.read_csv('./data/monthly_data.csv', parse_dates=['date'])


monthly_data.head()
```

[3]:
```
        date  gold_usd   cpi_us
0 2014-09-01  1241.330  237.852
1 2014-10-01  1223.565  238.031
2 2014-11-01  1176.413  237.433
3 2014-12-01  1200.440  236.151
4 2015-01-01  1249.333  234.812
```

[ ]:
```python
#Assume we bought bitcoin on a particular date and then we compute performance
sp500 = pd.read_csv('./data/sp500.csv', parse_dates=['date'])
bitcoin = pd.read_csv('./data/bitcoin-usd.csv', parse_dates=['date'])
bitcoin.head()
```

[ ]:
```
        date        open        high         low       close      volume
0 2014-09-17  465.864014  468.174011  452.421997  457.334015  21056800.0
1 2014-09-18  456.859985  456.859985  413.104004  424.440002  34483200.0
2 2014-09-19  424.102997  427.834991  384.532013  394.795990  37919700.0
3 2014-09-20  394.673004  423.295990  389.882996  408.903992  36863600.0
4 2014-09-21  408.084991  412.425995  393.181000  398.821014  26580100.0
```

[ ]:
```python
print(sp500.date.min())
print(bitcoin.date.min())
print(monthly_data.date.min())
print(bitcoin.shape)
print(sp500.shape)
print(monthly_data.shape)
print(bitcoin.describe())
print(sp500.describe())
print(monthly_data.describe())
```

```
2014-09-17 00:00:00
2014-09-17 00:00:00
2014-09-01 00:00:00
```

```
(2619, 6)
(1805, 6)
(87, 3)
```

	open	high	low	close	volume
count	2615.000000	2615.000000	2615.000000	2615.000000	2.615000e+03
mean	10051.643066	10334.482966	9750.736512	10073.814423	1.400155e+10
std	14892.430109	15326.320248	14422.269302	14923.069664	1.993158e+10
min	176.897003	211.731003	171.509995	178.102997	5.914570e+06
25%	582.071015	588.960998	575.311981	582.555999	7.489110e+07
50%	5745.599121	5865.881836	5544.089844	5750.799805	4.679500e+09
75%	9866.986328	10136.996094	9642.615235	9870.199219	2.287606e+10
max	67549.734375	68789.625000	66382.062500	67566.828125	3.509679e+11

	open	high	low	close	volume
count	1805.000000	1805.000000	1805.000000	1805.000000	1.805000e+03
mean	2755.938758	2769.524277	2741.245103	2756.455533	3.844502e+09
std	698.212835	701.268104	695.674679	698.850564	9.781460e+08
min	1833.400024	1847.000000	1810.099976	1829.079956	1.296540e+09
25%	2123.159912	2129.870117	2114.719971	2124.290039	3.254950e+09
50%	2664.439941	2682.860107	2648.870117	2663.989990	3.623320e+09
75%	3045.750000	3068.669922	3012.590088	3039.419922	4.154240e+09
max	4707.250000	4718.500000	4694.390137	4701.700195	9.878040e+09

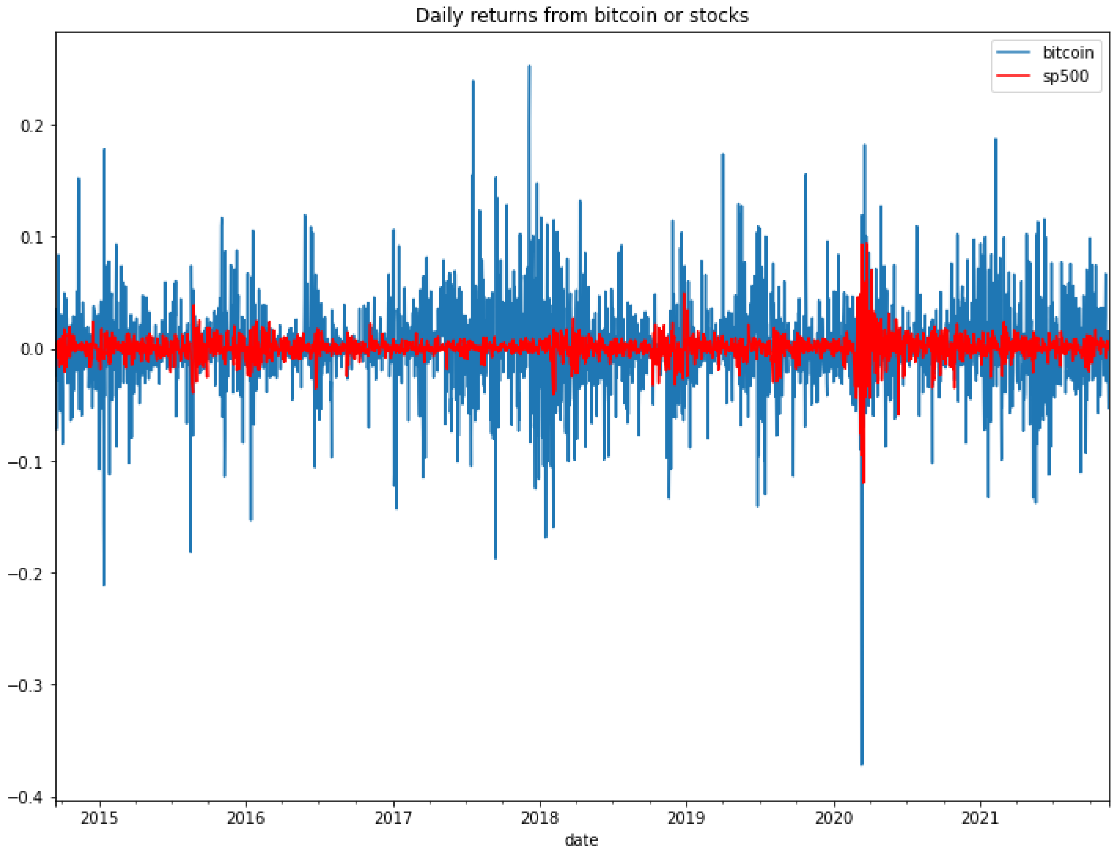
	gold_usd	cpi_us
count	87.000000	87.000000
mean	1403.186678	249.790759
std	257.985374	10.733951
min	1068.317000	233.707000
25%	1231.081500	240.428500
50%	1283.189000	249.554000
75%	1577.216000	257.091000
max	2041.700000	276.589000

Bitcoin versus stock volatility

If we analyse daily returns of bitcoin versus stock market we find bitcoin is very volatile

[ ]:
```python
bitcoin['returns']=bitcoin.close.pct_change(1)
bitcoin.set_index('date',inplace=True)
ax1=bitcoin.returns.plot(figsize=(12,9),title='Daily returns from bitcoin or␣
 ↪stocks')
sp500['returns_sp']=sp500.close.pct_change(1)
sp500.set_index('date',inplace=True)
sp500.returns_sp.plot(figsize=(12,9),color='red')
ax1.legend(['bitcoin','sp500'])
```

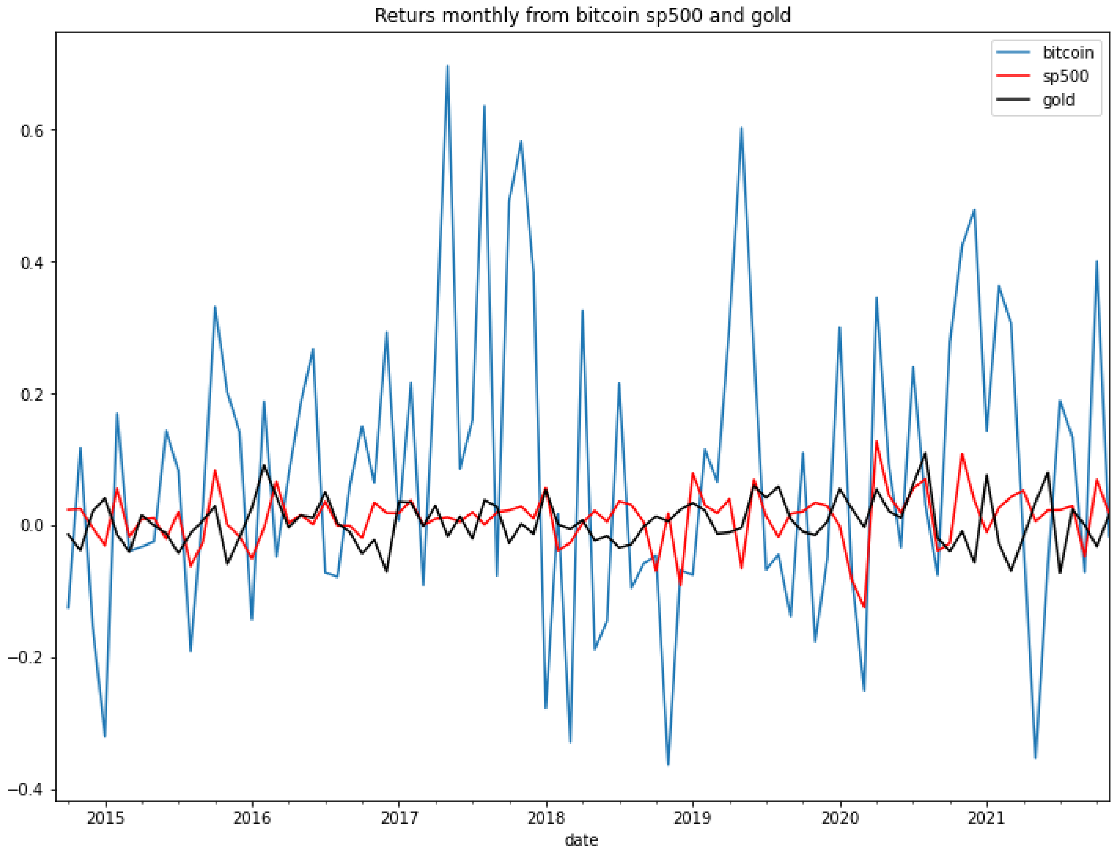
[ ]:
```
<matplotlib.legend.Legend at 0x7f63a3a9c430>
```



Understanding volatilty monthly for gold bitcoin and sp500

[ ]:
```
import matplotlib.pyplot as plt

monthly_bitcoin_returns=bitcoin['close'].resample('M').ffill().pct_change()
ax1=monthly_bitcoin_returns.plot(figsize=(12,9))

monthly_sp500_returns=sp500.close.resample('M').ffill().pct_change()
monthly_sp500_returns.plot(figsize=(12,9),color='red')
monthly_data['returns_gold']=monthly_data.gold_usd.pct_change(1)

monthly_data.set_index(monthly_data.date,inplace=True)
monthly_data['returns_gold'].plot(figsize=(12,9),color='black')
ax1.legend(['bitcoin','sp500','gold'])
ax1.set_title('Returs monthly from bitcoin sp500 and gold')
```

[ ]:
```
Text(0.5, 1.0, 'Returs monthly from bitcoin sp500 and gold')
```

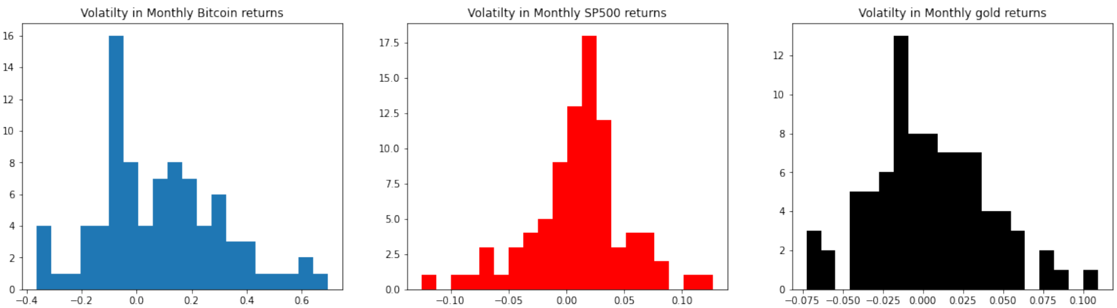


From the graph we can see that bitcoin is the most volatile with returns on a monthly basis changing close to 5-6%

The below histogram helps to quantify the spread of the returns

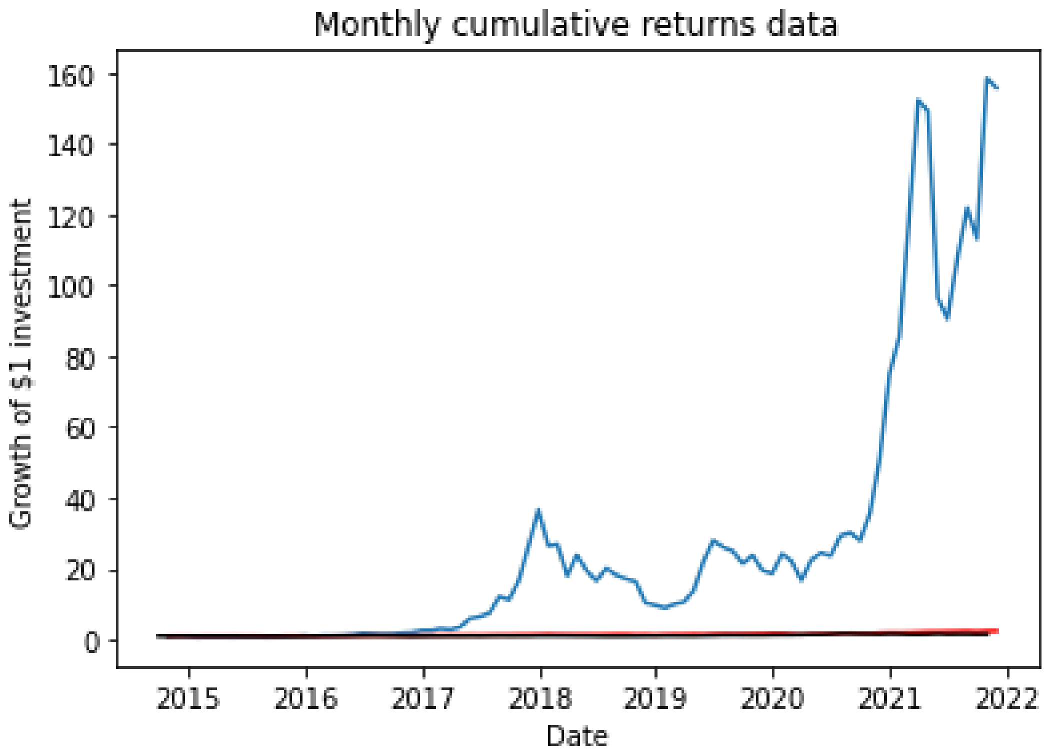
[ ]:
```
fig, axs = plt.subplots(nrows=1,ncols=3,figsize=(20,5))

axs[0].hist(monthly_bitcoin_returns,bins=20)
axs[0].set_title('Volatilty in Monthly Bitcoin returns')
axs[1].hist(monthly_sp500_returns,color='red',bins=20)
axs[1].set_title('Volatilty in Monthly SP500 returns')
axs[2].hist(monthly_data.returns_gold,color='black',bins=20)
axs[2].set_title('Volatilty in Monthly gold returns')
```

[ ]:
```
Text(0.5, 1.0, 'Volatilty in Monthly gold returns')
```



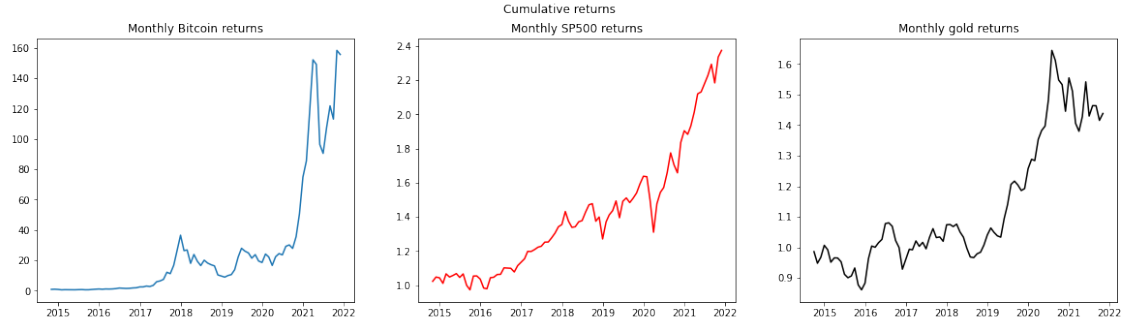
PERFORMANCE OF BITCOIN VERSUS STOCK VERSUS GOLD

[ ]:
```
fig = plt.figure()
ax1 = fig.add_axes([0.1,0.1,0.8,0.8])
#bitcoin_cum_returns = (bitcoin.returns+ 1).cumprod()

monthly_cum_returns = (monthly_bitcoin_returns + 1).cumprod()
monthly_sp500_cum_returns = (monthly_sp500_returns+ 1).cumprod()
monthly_data_cum_returns = (monthly_data.returns_gold+ 1).cumprod()


ax1.plot(monthly_cum_returns)
ax1.plot(monthly_sp500_cum_returns,color='red')
ax1.plot(monthly_data_cum_returns,color='black')
ax1.set_xlabel("Date")
ax1.set_ylabel("Growth of $1 investment")
ax1.set_title("Monthly cumulative returns data")
plt.show()
```



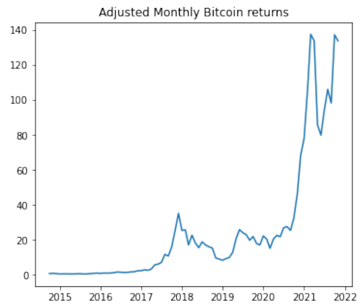
[ ]:
```
fig, axs = plt.subplots(nrows=1,ncols=3,figsize=(20,5))

fig.suptitle('Cumulative returns')
axs[0].plot(monthly_cum_returns)
axs[0].set_title('Monthly Bitcoin returns')
axs[1].plot(monthly_sp500_cum_returns,color='red')
axs[1].set_title('Monthly SP500 returns')
axs[2].plot(monthly_data_cum_returns,color='black')
axs[2].set_title('Monthly gold returns')
```

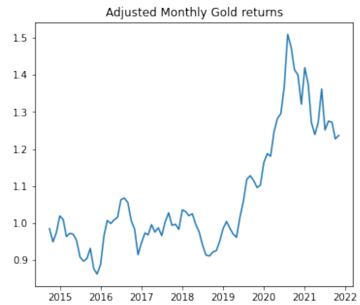
[ ]:
```
Text(0.5, 1.0, 'Monthly gold returns')
```

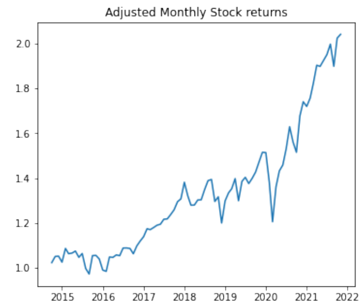


[ ]:
```
idx=pd.DatetimeIndex(monthly_cum_returns.index.year.
 ↪astype(str)+'-'+monthly_cum_returns.index.month.astype('str'))
bitcoin_df=pd.DataFrame(monthly_cum_returns).set_index(idx)

idx1=pd.DatetimeIndex(monthly_sp500_cum_returns.index.year.
 ↪astype(str)+'-'+monthly_sp500_cum_returns.index.month.astype(str))
sp500_df=pd.DataFrame(monthly_sp500_cum_returns).set_index(idx1)
idx2=pd.DatetimeIndex(monthly_data_cum_returns.index.year.
 ↪astype(str)+'-'+monthly_data_cum_returns.index.month.astype(str))

gold_df=pd.DataFrame(monthly_data_cum_returns).set_index(idx2)


df1=gold_df.merge(bitcoin_df,left_index=True,right_index=True)
df1.columns=['gold_returns','bitcoin_returns']
df2=df1.merge(monthly_data,left_index=True,right_index=True)
reqd_df=df2[['gold_returns','bitcoin_returns','cpi_us']]
reqd_df=reqd_df.merge(sp500_df,left_index=True,right_index=True)
reqd_df.columns=['gold_returns','bitcoin_returns','cpi_us','sp500_returns']
reqd_df['base_cpi']=reqd_df.cpi_us[0]
reqd_df['adj_gold']=reqd_df.gold_returns*reqd_df.base_cpi/reqd_df.cpi_us
reqd_df['adj_bitcoin']=reqd_df.bitcoin_returns*reqd_df.base_cpi/reqd_df.cpi_us
reqd_df['adj_stock']=reqd_df.sp500_returns*reqd_df.base_cpi/reqd_df.cpi_us


adj_bitcoin_df=reqd_df['adj_bitcoin']
adj_gold_df=reqd_df['adj_gold']
adj_stock_df=reqd_df['adj_stock']




fig, axs = plt.subplots(nrows=1,ncols=3,figsize=(20,5))

axs[0].plot(adj_bitcoin_df)
axs[0].set_title('Adjusted Monthly Bitcoin returns')
axs[1].plot(adj_gold_df)
axs[1].set_title('Adjusted Monthly Gold returns')
axs[2].plot(adj_stock_df)
axs[2].set_title('Adjusted Monthly Stock returns')
```

[ ]:
```
Text(0.5, 1.0, 'Adjusted Monthly Stock returns')
```







Bitcoin has a good growth but is much more volatile than both stocks and gold. After inflation adjusting we observe the returns are still good for bitcoin.For every 1 $ invested we get 140 $in 2022 .Depending on the expected return and risk profile investing in bitcoin may be to a good hedge against inflation

Portfolio expectaions

Since we are looking to lower our volatility we want to limit exposure to bitcoin. Depending on our risk appetite a maximum of 10-15% of asset allocation would be ideal.Stocks have higher returns than gold but they are also more volative .So I would suggest 70-75% exposure to stocks and the rest I would invest in gold 15-20%.

2.3 Competition challenge

Create a report that covers the following:

1. How does the performance of Bitcoin compare to the S&P 500 and the price of gold?
2. Analyze Bitcoin's returns and volatility profile. Do you believe it could help improve the performance of a portfolio? Do you believe Bitcoin could be used as a hedge versus inflation?
3. The CFO is looking to lower volatility in the fund. Explore building a portfolio using some or all of these assets. Make a recommendation that minimizes overall risk.

```
[ ]:
```

2.4 Judging criteria

CATEGORY	WEIGHTING	DETAILS

| **Recommendations** | 35% |

Clarity of recommendations - how clear and well presented the recommendation is.

Quality of recommendations - are appropriate analytical techniques used & are the conclusions valid?

Number of relevant insights found for the target audience.

|

| **Storytelling** | 30% |

How well the data and insights are connected to the recommendation.

How the narrative and whole report connects together.

Balancing making the report in depth enough but also concise.

| | **Visualizations** | 25% |

Appropriateness of visualization used.

Clarity of insight from visualization.

| | **Votes** | 10% |

Up voting - most upvoted entries get the most points.

|

2.5 Checklist before publishing into the competition

- Rename your workspace to make it descriptive of your work. N.B. you should leave the notebook name as notebook.ipynb.
- Remove redundant cells like the judging criteria so the workbook is focused on your story.
- Make sure the workbook reads well and explains how you found your insights.
- Check that all the cells run without error.

2.6 Time is ticking. Good luck!

[ ]:

```
```