Gold and cryptocurrencies exchange rates correlation

Python Project

Polytechnique Institute of Paris
Applied Mathematics and Statistics - Master degree
Python for Datascience - Anas Hattay
Fall Semester 2025 - M1 APPMS

Zhousiying Wu, Nicolas Valety
18.12.2025 - v0.1.0 - draft

Contents

1 Introduction

1.1 Context

Gold has been used as currency since the 6^{th} century BC and has played an important role throughout world history. Especially after the establishment of the gold standard, gold becomes the basis for maintaining the world monetary system. Later, although the gold standard system was gradually abolished in the 20^{th} century due to the Great Depression, gold still occupies an important position in the modern economy. Conversely, cryptocurrencies are much more recent and have been launched in the 21^{th} century on the Blockchain. Even if the gold standard system has been abolished, does any correlation between gold and cryptocurrencies could be found? An analysis on several period of time in order to find out one (or more) year when gold stop (or begin) to act as a standard for cryptocurrencies. Our research will aim in a procedure to find it out by defining objectives, enumerating methods needed and the toolkit compulsory to compute data.

1.2 Aims and Objectives

1.2.1 Aims: We want to find out a correlation between gold and any cryptocurrency exchange rates.

> At this point, there may not be any correlation between both exchanges rates as one is very stable and the other is very volatile.

1. Objective 1: Compare the exchange rates, find out if there is a comparable period of time when the cryptocurrency and gold are correlated.
2. Objective 2: Identify some ranges which contain (or do not) periods of correlation between gold and cryptocurrencies.

1.3 Implementation

- Naive model: This is chosen as a baseline, the logic here is tomorrow's price = today's price. Because gold prices fluctuate very little in the short term,

I think this is a good benchmark.
- ARIMA: Because gold prices tend to revert to the mean in the long run, prices fluctuate over time, and past prices influence future prices. ARIMA can capture

the linear trend and autocorrelation of gold price changes.
- SARIMA: SARIMA and ARIMA operate on similar principles, but SARIMA incorporates seasonal patterns. Because gold prices are linked to seasonal demand

(e.g., as holiday gifts or wedding jewelry) and cultural patterns (e.g., the Indian wedding season and Chinese New Year), we believe economic cycles may exist. Additionally, the seasons in mining areas also affect production, thus influencing prices.
- Deep learning :

Evaluation: Naive model: ARIMA: SARIMA:

Then we add exgeous features to improve the predict, the exgeous feature we generated are

- day of week (0:monday - 6:sunday), because gold trading might have weekly pattern
- which week in the year(1-52), because gold jewelry trading might increase in certain months (e.g.chinese new year, indian wedding season)
- month of year(1-12), days, months, years
- month since start: As the data is started from 2022-05-16 to 2024-12-09, so this is how many months have passed since our data is started. Because it is obvious that gold price

have long-term upward trend. And models gradual inflation effects over time.

- if that day is a holiday / how many days to holiday: Since markets are closed for holidays and there is no trading data, we use the average price before and after the holidays to fill

the gap. Additionally, many investors adjust their positions before holidays, which also contributes to gold price fluctuations. The uncertainty during major holiday closures also affects gold prices.

2 Method

2.1 Libraries used

> NumPy, KaggleHub, YFinance and Pandas are directly handled by the Dataset class

- NumPy (>=1.19.5): array manipulation
- Pandas (>=1.4.4): dataset (dataframe) manipulation
- MatPlotLib (>=3.5.3): plotting graphs on the screen
- Seaborn (>=0.12.2): extends MatPlotLib in order to draw correlation matrices
- SciPy (>=1.9.1): base package for every Scikit libraries
- Scikit-Learn (1.7.2): toolkit used to scale our dataset and to use some metrics during tests
- StatsForecast (>=2.0.3): machine learning (used for the AutoARIMA model)
- PyTorch (>=2.9.1): machine learning (used for the LSTM model)
- UtilsForecast (>=0.2.14): evaluation of the models
- KaggleHub (>=0.3.8): data downloader from Kaggle without the need of an API key
- YFinance (>=0.2.18): data downloader from Yahoo! Finance

2.2 Sources

Dataset is composed of data scraped from two websites: Kaggle and Yahoo! Finance. Is has been processed and saved on the computer as a coma separated values (.csv) file.

2.2.1 Kaggle

- Crypto cryptocurrencies daily prices
- Gold historical data daily updated
- There are only an hundred of tickers on Kaggle and the files sometimes disappear from the remote dataset, leading us to ajust the source code because Kaggle does not release a public API usable without API key.
- There is a strange hole in the cryptocurrencies dataset from December 10th 2024 to January 26th 2025.
- In the other hand, the dataset for gold value is really complete and has a longer date range than the one from yfinance.

2.2.2 Yahoo! Finance

- There are much more tickers available on yfinance than on Kaggle.
- Date range is limited.

2.2.3 Local .csv file

- Really faster loading from a local file than downloading from Internet.
- Prevent changes of results or loss of ticker once the date range has been set up.

2.3 Requirements

2.3.1 Dataset restrictions

- Due to the high number of tickers, we have only used the crytos with at least 65% of correlation with gold (whatever the method used).
- There is a hole in the dataset downloaded from Kaggle: no data are available **between December 10th 2024 and January 26th 2025**. So the data analysis will occur from the first common date with all the interesting tickers and December 9th 2024.
- No cryptocurrency were available before July 17th 2010 so all dates (from gold dataset) before that moment have been droped

2.3.2 Dataset modifications

- In order to use a time series model, the dataset must have its columns renamed:
  1. All the dates are in a **ds** column (on a daily basis)
  2. Gold values are in a **y** column
  3. All exogenous data (cryptocurrency tickers) are normalized
  4. Two normalization methods have been used:
     - MinMax scaler from Scikit Learn
     - Mathematical normalization with the following formula: $\frac{X-\overline{X}}{\sigma_X}$

2.4 Dataset

2.4.1 Dataset structure

	date	ticker	close	closeNormalized
index	date	currency short name	price at closing time in \$	normalized (by ticker) price

Table 1 - Dataset structure as a Pandas Dataframe

2.4.2 Timeseries structure

ds	exogenous tickers	y	unique_id
date	normalized price at closing time (one column for each ticker)	gold price at closing time (in \$)	XAU

Table 2 - Timeseries structure (1/3)

day	week	month
day of the month	week of the year	month of the year

Table 3 - Timeseries structure (2/3)

months_since_start	is_holiday	days_to_holiday
number of month since the first date	holiday with respect of US federal calendar	amount of days remaining before next holidays

Table 4 - Timeseries structure (3/3)

2.4.3 Tickers

Three different correlation matrices have been computed:

1. **Pearson's correlation matrix** $= \mathbb{E}\left[\frac{(X-\overline{X})\cdot(Y-\overline{Y})}{\sigma_X \cdot \sigma_Y}\right] = \frac{\mathrm{Cov}(X,Y)}{\sigma_X \cdot \sigma_Y}$

2. **Kendall's correlation matrix** $= \frac{(\#\text{ concordant pairs})-(\#\text{ discordent pairs})}{\#\text{ pairs}}$
3. **Spearman's correlation matrix** $= \mathrm{Pearson}(\mathrm{rank}(X), \mathrm{rank}(Y))$

In order to select the most relevant tickers, the 3 correlation matrices have been computed once with the whole dataset. Then, only the 65% most correlated tickers were kept and only the common date range between all tickers were kept. Then the computation of the 3 correlation matrices were started again to have more accurate values.

The following tickers have more than 65% of correlation (with respect to Pierson's, Kendall's and Spearman's methods) with gold over the whole period and have been selected to be part of the time series:

ticker	Pierson	Kendall	Spearman
BTC	0.9107031967844699	0.730110653102119	0.9175005834238465
GT	0.872068366942732		0.7556076533399873
SOL	0.872068366942732		0.7116827915226329
BNB	0.870625369815244		0.6897266806989514
TRX	0.865148992461277	0.7023226930482401	0.8838661408310834
SUN	0.831424628414087		0.7411085422153962
ETH	0.7737201985896954	0.6539055159233185	0.8732200136775444
RAY	0.7691850017394904		
FET	0.7599349834769431	0.7124420078786323	0.9065053525645362
APE	−0.6945792605468512		−0.7850699740155144
LEO	0.6721238538115092		
DOGE			0.7364563538003932
XRP			0.6927795287025804
LINK			0.6619425674230894

Table 5 - Tickers correlation with gold

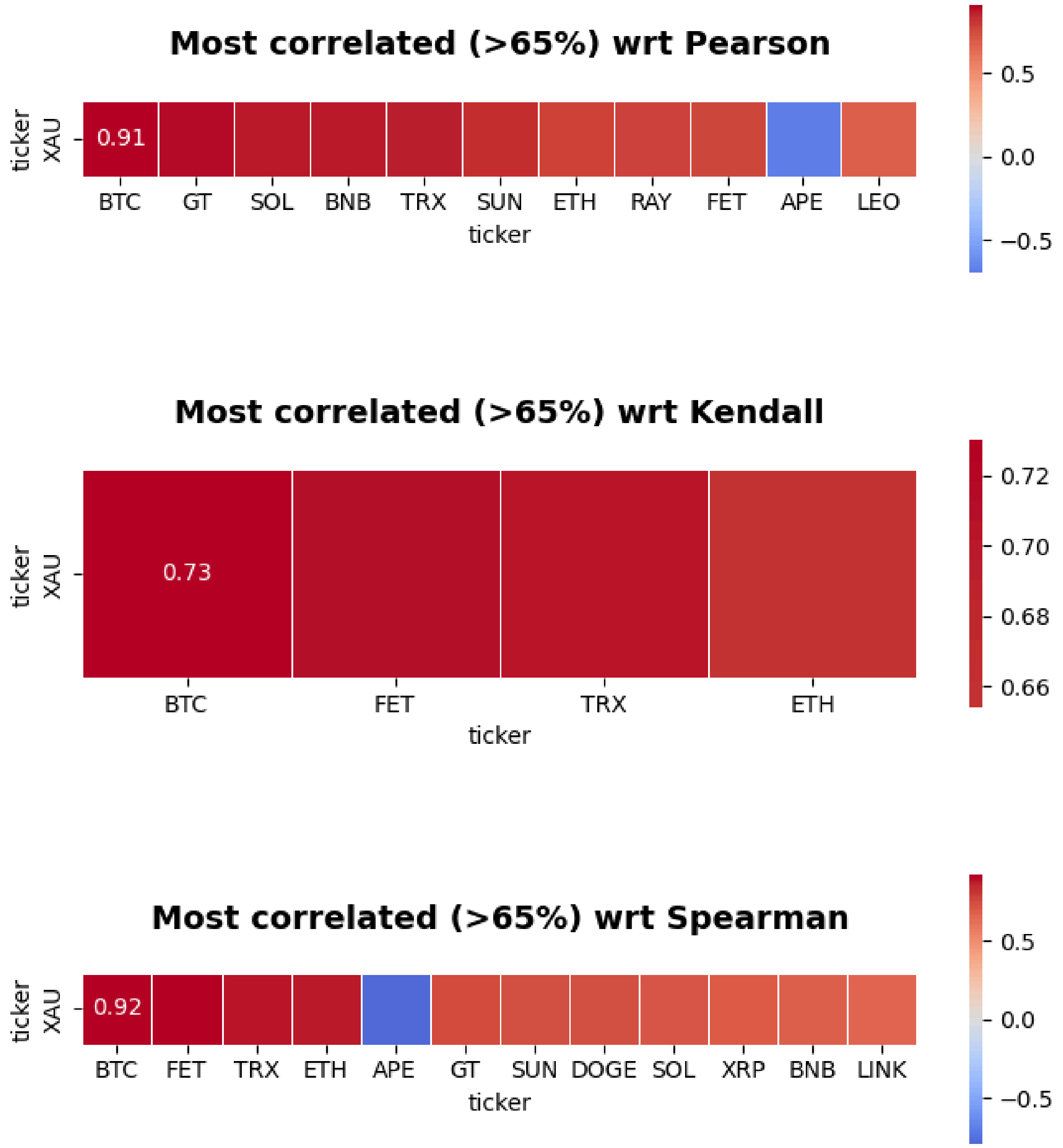

Figure 1 - Correlation matrices

2.5 Time series

2.5.1 Algorithms

With the most correlated cryptocurrencies writen as exogenous datas in the time series of gold price, we tried to predict gold price using two algorithms:

1. ARIMA
2. LSTM
3. a combination of both

3 Result

metric	Naive	HistoricAverage	WindowAverage	SeasonalNaive	ARIMA	SARIMA
MAE	57.471655	598.324263	83.421234	86.408374	71.508333	71.508333
MSE	4161.423763	359019.774221	7986.952859	9092.011023	5916.408276	5916.408276

Table 6 - Baseline Model comparison

From the baseline models, we found that ARIMA and SARIMA have the best performance although it is worse than Naive model. That's may because Naive model use yesterday's price as today's price and the flactuation for gold price between a short periode will only have small change. So we use ARIMA and SARIMA to do further analysis.

Model	MAE	RMSE
ARIMA_TimeOnly	74.022	79.428
SARIMA_TimeOnly	74.022	79.428
ARIMA_Crypto+Time	92.358	101.845
SARIMA_Crypto+Time	117.592	132.474

Table 7 - ARIMA/SARIMA comparison between different features

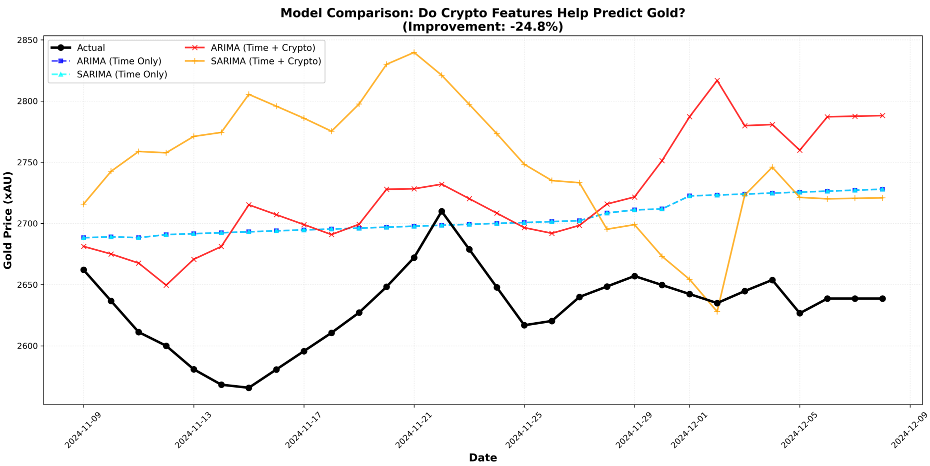

Figure 2 - Visuliazition ARIMA/SARIMA

Although from the table above, it seems crypto features do not improve the prediction, the error for models with time features is smaller than those with time features and crypto features, that because time-only model predicts near average. But when we look at the visuliazition for these models, we can see that time-model is basically a flat line, just like naive model. So it is not useful for trading prediction. And time+crypto model captures actual price movements, it has flactuations like real market price on the graph, so we think time+crypto model si more valuable even if point estimate are slightly off and have bigger error.

And in time+crypto models, ARIMA is better than SARIMA. So we decided to use ARIMA and another complex model(deep learning) to do further analysis.

Model	MAE	RMSE	R^2
ARIMA_TimeOnly	45.66	61.00	−0.358
LSTM_TimeOnly	540.974	543.786	−106.889
Ensemble_Avg_TimeOnly	278.607	284.345	−28.499
Ensemble_Weighted_TimeOnly	60.585	81.713	−1.436

Table 8 - Time-Only Dataset Result

Model	MAE	RMSE	R^2
ARIMA_Time+Crypto	47.485	58.477	−0.248
LSTM_Time+Crypto	59.376	70.860	0.0239
Ensemble_Avg_Time+Crypto	43.320	57.908	−0.223
Ensemble_Weighted_Time+Crypto	42.270	57.227	−0.195

Table 9 - Time + Crypto Dataset Result

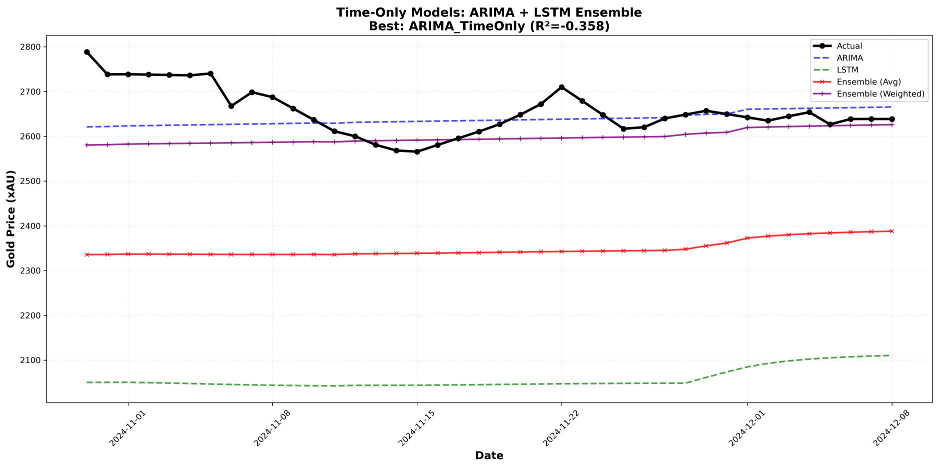

Figure 3 - Visuliazition time-only models

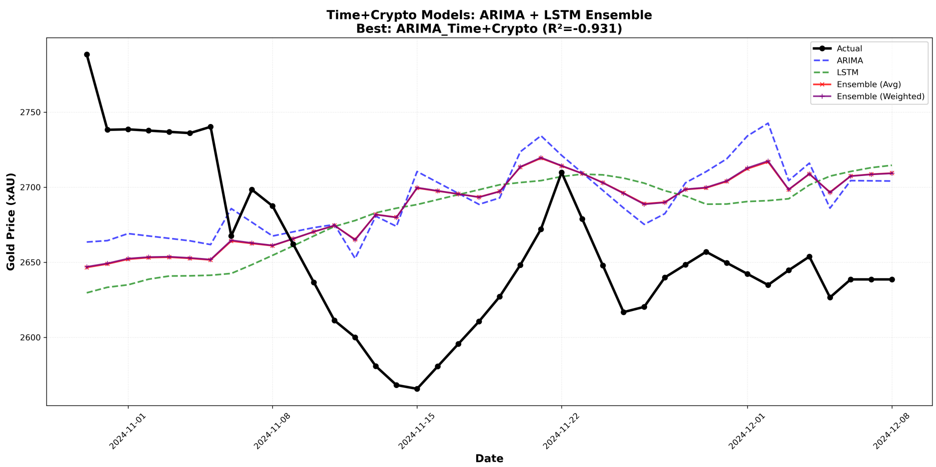

Figure 4 - Visuliazition time+crypto models

Overall, all of our models have negative R^2, which means our model are all failing. But we believe this is normal, because market trading prices themselves cannot be predicted by our simple models. However, our goal is to find out if crypto features can help with prediction. The models here are

ARIMA, deep laearning and the model combined by them. Same as before the visuliazition for time-only models are all like flat line, but the models with time and crypto prices exhibit fluctuations similar to those in the real market. So our conclusion is crpto will improve the model.

We adjust test period for ARIMA in combined model comparision(from 30 days to 60 days), and the performance improved significantly. And from the table, we found that weighted emsemble model is best, this aligns with our predictions, and using more complex models can improve our forecasts.

4 Conclusion

This final chapter summarizes the project outcomes, comparing them with the initial objectives. It also reflects on encountered difficulties and discusses possible improvements or future developments.

Key elements:
- ***Summarize the results*** *– What are the key takeaways from the project?*
- ***Compare with objectives*** *– Did the project meet its original goals? Why or why not?*
- ***Reflect on challenges*** *– What were the biggest difficulties, and what was learned?*
- ***Discuss future work*** *– What are possible improvements or next steps?*

This project leads us to realise that there are a high correlation between a non-negligeable amount of cryptocurrencies and gold value. Therefore, predicting future gold price regarding cryptocurrencies is not that simple and even a mix of powerfull machine learning algorithm (ARIMA + LSTM) cannot give useful insight of any change in gold exchange rate.

Glossary

Bibliography