

Gold Price ETL Pipeline & Forecast



ECUTBILDNING

Hani Abraksiea

EC Utbildning

Fördjupning i Pythonprogrammering

2025–09

1 Abstract

This project demonstrates a complete data workflow for gold price analysis. The pipeline automatically fetches gold prices from an API, stores the data in a SQLite database, and logs all executions. Historical prices are analyzed and visualized, while a simple machine learning model is applied to forecast the next day's gold price. The project shows how modern data engineering and data science methods can be combined into an end-to-end solution, from raw data extraction to prediction.

Innehållsförteckning

1	Abstract	2
2	Introduktion.....	1
3	Projektmål	1
4	Metod / Implementation.....	1
4.1	ETL (Extract – Transform – Load)	1
4.2	Logging	1
4.3	Analys.....	1
4.4	Prognos (Forecast)	2
4.5	Testning.....	3
5	Resultat.....	3
6	Diskussion	4
7	Slutsats	4
8	Bilagor och Källor.....	4

2 Introduktion

Syftet med detta projekt är att demonstrera en fullständig data-pipeline för att hämta, lagra, analysera och förutsäga guldpiser. Projektet omfattar alla steg i en modern datahanteringsprocess: datainsamling via API, lagring i en databas, analys och visualisering av historiska priser samt en enkel maskininlärningsmodell som förutsäger nästa dagspris. Arbetet är relevant eftersom det visar hur man kan bygga en automatiserad pipeline från rådata till insikt och prognos – ett tillvägagångssätt som är centralt inom Data Science och Business Intelligence.

3 Projektmål

Projektet hade följande mål:

- Hämta aktuellt guldpriß i USD från ett öppet API (GoldAPI).
- Lagra insamlade data i en SQLite-databas.
- Logga alla körningar och eventuella fel i en loggfil.
- Analysera historiska priser och visualisera trender.
- Bygga en enkel modell för att förutsäga nästa dags guldpriß.
- Testa pipelinekomponenterna för att säkerställa att de fungerar som avsett.

4 Metod / Implementation

4.1 ETL (Extract – Transform – Load)

Pipeline-scriptet (pipeline.py) hämtar guldpriß från GoldAPI med requests. Daten omvandlas till ett pandas-DataFrame, kompletteras med en tidsstämpel (loaded_at), och sparas i en SQLite-databas (gold_prices.db).

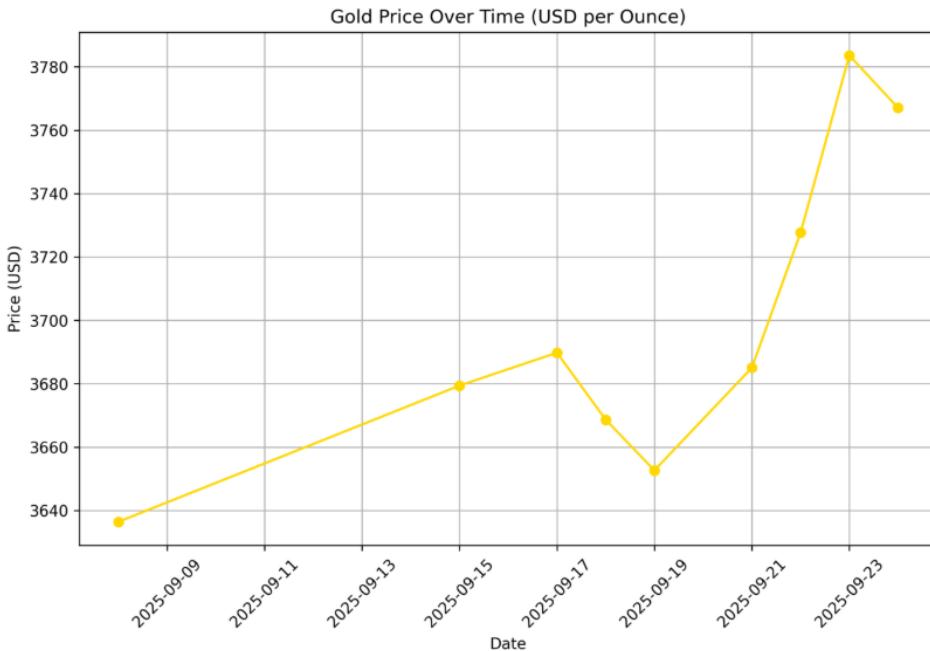
4.2 Logging

Med Pythons logging-modul sparas information om varje körning samt eventuella fel i pipeline.log. Detta gör det enkelt att spåra problem eller följa körhistorik.

4.3 Analys

I analyze.py analyseras historiska guldpiser med hjälp av pandas och matplotlib. Grafer genereras som visar prisutveckling över tid samt enklare statistiska mått som medelvärde och variation.

För att analysera utvecklingen av guldpriset över tid genererades en graf baserad på de historiska priserna i databasen. Grafen visar hur priset har varierat dag för dag under insamlingsperioden.



Figur 1. Utveckling av guldpriset över tid (USD per ounce).

4.4 Prognos (Forecast)

I forecast.py implementeras en linjär regressionsmodell med scikit-learn. Datan delas i tränings- och testmängd. Modellen används för att förutsäga nästa dagspris. Noggrannheten utvärderas med Mean Absolute Error (MAE).

Tabellen nedan visar en jämförelse mellan de faktiska guldpriserna och de priser som predicerats av vår enkla maskininlärningsmodell (Linear Regression). Modellen har tränats på historiska priser och utvärderats med hjälp av Mean Absolute Error (MAE), vilket visar den genomsnittliga avvikelsen mellan prediktionerna och de faktiska priserna.

Mean Absolute Error (MAE): 26.23 USD		
	Actual	Predicted
0	3727.310	3683.710786
1	3728.005	3717.090245
2	3783.605	3717.639132
3	3763.805	3761.550120
4	3765.685	3745.912754

Predicted next day price: 3750.37 USD		
---------------------------------------	--	--

Figur 2. Faktiska vs. predicerade guldpriser (USD per ounce) för testdata. Nästa dags predicerade pris är 3750,37 USD.

4.5 Testning

För att säkerställa robusthet skapades testfilen `test_pipeline.py` med pytest. Testerna verifierar att:

- API-svaret innehåller prisfältet (price).
- Data kan sparas i SQLite-databasen.
- Loggfilen (pipeline.log) skapas korrekt.

Tester körs med:

```
pytest test_pipeline.py -v
```

Exempel på resultat:

```
Ran 3 tests in 0.007s
```

```
OK
```

Detta bekräftar att pipeline och dess huvudkomponenter fungerar som avsett.

5 Resultat

Projektet resulterade i en fungerande ETL-pipeline som kan köras manuellt eller schemaläggas med Windows Task Scheduler. Följande resultat observerades:

- **Databas:** Alla körningar sparas i `gold_prices.db`. Varje rad innehåller basvaluta (USD), mål (XAU), pris samt tidsstämpel.
- **Loggning:** `pipeline.log` innehåller tidsstämplade rader som visar lyckade körningar samt eventuella fel (t.ex. nätverksproblem).
- **Analys:** Grafer visar historiska prisvariationer. Exempel: trendkurva, glidande medelvärde.
- **Forecast:** Den enkla linjära modellen lyckades förutsäga prisnivåer nära de faktiska med ett rimligt MAE, men precisionen är begränsad av liten mängd historiska data.
- **Tester:** Alla tester passerade, vilket stärker att implementationen är stabil.

6 Diskussion

Arbetet fungerade bra överlag. Huvudkomponenterna – API-anrop, datalagring, loggning och analys – integrerades utan större problem. Några utmaningar var:

- API-nyckeln behövde hanteras säkert (lösning: .env-fil och .gitignore).
- Datamängden är liten, vilket begränsar maskininlärningsmodellens noggrannhet.
- Prognoserna kan förbättras med mer data eller mer avancerade modeller (t.ex. ARIMA, Prophet eller LSTM).

Förslag på förbättringar:

- Schemalägg pipelinen så att den körs dagligen automatiskt.
- Utöka datainsamlingen till flera metaller eller valutor.
- Bygga ett interaktivt dashboard för analys.

7 Slutsats

Projektet visar en fullständig pipeline från datainsamling till analys och prognos. Resultatet är en fungerande ETL-lösning som automatiskt hämtar guldpriiser, sparar dem i en databas, loggar körningar, visualiseringar historiska data och gör enkla prediktioner. Trots enkla metoder demonstrerar projektet en systematisk arbetsprocess och tydlig förståelse för datadrivna arbetsflöden.

8 Bilagor och Källor

- Källkoden för projektet finns tillgänglig på GitHub:
 GitHub-repository
https://github.com/HaniAbraksiea/ds24_pyadv/tree/main/gold_price_project
- API-dokumentation: <https://www.goldapi.io/>
- Bilder: grafer från `analyze.py` och `forecast.py`.