Project One

Investment Portfolio

Joeny Vo

Portfolio Overview

- Initial Investment
  - $100,000
- Portfolio Duration
  - 1 year
  - July 1st, 2023 - June 30th , 2024
- Diversification
  - 3 Stocks
  - 1 Exchange-Traded Fund
  - 1 Forex

Investment Selection

- Stocks
  - Nvidia
  - Amazon
  - JPMorgan
- ETF
  - SPDR S&P 500 ETF Trust (SPY)
- Forex
  - USD/CAD

Investment Strategy

Initial Investment - $100,000

- 60% - Stocks
- 30% - ETF
- 10% - Forex

Additional investment of $4,000

- Every 4 months
- Based on analysis of time span

ETL Pipeline Overview

Extract

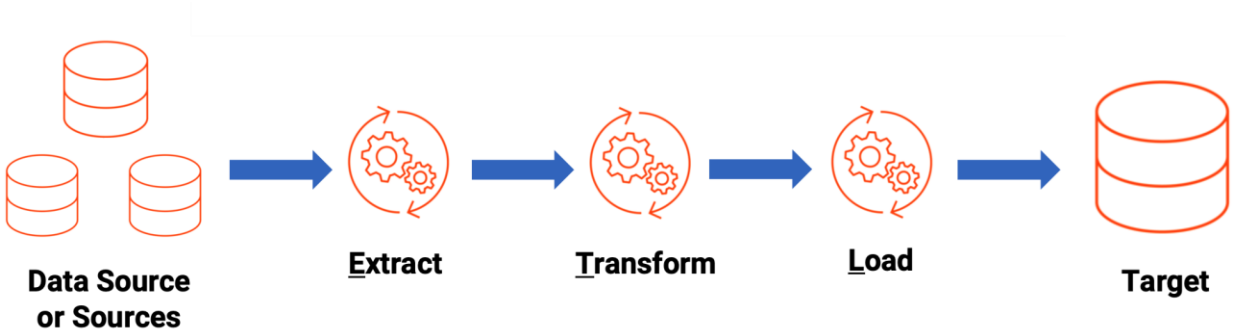
Data sources:
- Polygon.io

Transform

Data Cleaning
Adding Metrics
Star Schema

Load

Load Data
SQL Server



Extracting Data

- Polygon API
  - Requests
  - Save response to CSV
- Pandas
  - Load each response into separate data frame object

```
response = requests.get(
    f"{base_url}/v2/aggs/ticker/{investment["ticker"]}/range/1/day/2023-05-01/2024-07-01?adjusted=true&sort=asc",
    params,
)
```

```
    # Creating our Pandas dataframes from our saved csv file
    for investment in investments:
        investment["df"] = pd.read_csv(f"./Data/Fresh/{investment["name"]}.csv")
        print(f"name: {investment["name"]} length: {len(investment["df"])}")
```

✓ 0.0s

```
name: Nvidia length: 294
name: Amazon length: 294
name: JPMorgan length: 294
name: SPY length: 294
name: USDCAD length: 375
name: NDX length: 296
```

Transform – Cleaning

Cleaning up bad data

- Check for null values
- Remove duplicate rows

```
    # Check for null values
    for df in df_list:
        print(f"Number of missing values: {df.isnull().values.sum()}")
```

✓ 0.0s

```
Number of missing values: 0
Number of missing values: 0
Number of missing values: 0
Number of missing values: 0
Number of missing values: 0
Number of missing values: 0
```

```
# Remove duplicate rows if any
for df in df_list:
    df.drop_duplicates(inplace=True)
```

✓ 0.0s

Transform – Quality

Ensuring high quality data

- Renaming columns
- Drop unneeded columns
- Ensure correct data types
- Convert timestamp

#	Column	Non-Null Count	Dtype
0	v	294 non-null	float64
1	vw	294 non-null	float64
2	o	294 non-null	float64
3	c	294 non-null	float64
4	h	294 non-null	float64
5	l	294 non-null	float64
6	t	294 non-null	int64
7	n	294 non-null	int64

t
1682913600000
1683000000000
1683086400000
1683172800000
1683259200000

#	Column	Non-Null Count	Dtype
0	volume	294 non-null	int64
1	volume_weighted_average	294 non-null	float64
2	open_price	294 non-null	float64
3	close_price	294 non-null	float64
4	highest_price	294 non-null	float64
5	lowest_price	294 non-null	float64
6	date	294 non-null	datetime64[ns]

date
2023-05-01
2023-05-02
2023-05-03
2023-05-04
2023-05-05

Transform – Missing Data

Adding in missing data

- More days covered in some investments
- Inconsistent data issues
- Create dates range
  - Forward fill
- Dropped extra days

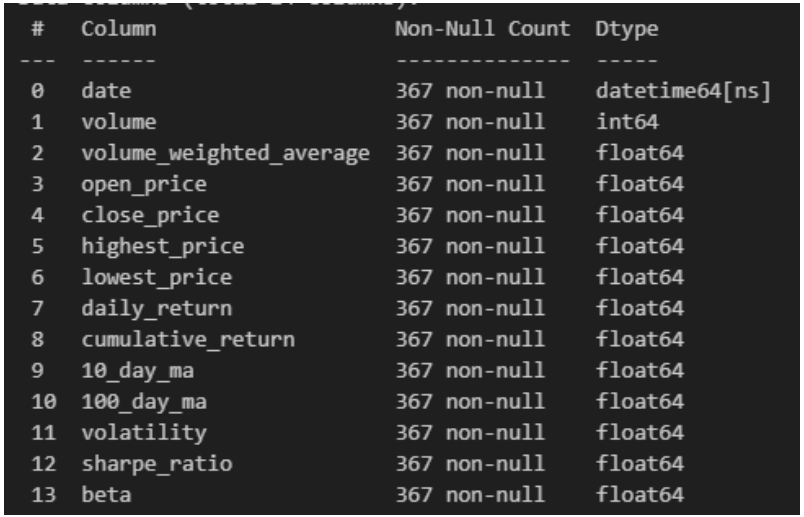
```
name: Nvidia length: 294
name: Amazon length: 294
name: JPMorgan length: 294
name: SPY length: 294
name: USDCAD length: 375
name: NDX length: 296
```

```
name: Nvidia length: 367
name: Amazon length: 367
name: JPMorgan length: 367
name: SPY length: 367
name: USDCAD length: 367
name: NDX length: 367
```

Transform – Aggregations

Adding in extra metrics

- Daily/Cumulative Returns
- 10/100 Day Moving Average
- Volatility
- Sharpe Ratio
- Beta



#	Column	Non-Null Count	Dtype
0	date	367 non-null	datetime64[ns]
1	volume	367 non-null	int64
2	volume_weighted_average	367 non-null	float64
3	open_price	367 non-null	float64
4	close_price	367 non-null	float64
5	highest_price	367 non-null	float64
6	lowest_price	367 non-null	float64
7	daily_return	367 non-null	float64
8	cumulative_return	367 non-null	float64
9	10_day_ma	367 non-null	float64
10	100_day_ma	367 non-null	float64
11	volatility	367 non-null	float64
12	sharpe_ratio	367 non-null	float64
13	beta	367 non-null	float64

Transform – Star Schema

Dimension Tables

- Investment Type
  - Stock, ETF, Forex, Index
- Name
- Sector
  - Tech, Finance, Broad Market
- Time

Fact Table

#	Column	Non-Null Count	Dtype
0	investment_id	2202 non-null	int64
1	investment_name_id	2202 non-null	int64
2	investment_type_id	2202 non-null	int64
3	investment_sector_id	2202 non-null	int64
4	time_id	2202 non-null	int64
5	close_price	2202 non-null	float64
6	daily_return	2202 non-null	float64
7	cumulative_return	2202 non-null	float64
8	10_day_ma	2202 non-null	float64
9	100_day_ma	2202 non-null	float64
10	volatility	2202 non-null	float64
11	sharpe_ratio	2202 non-null	float64
12	beta	2202 non-null	float64

Load – SQL Server

1. Saving fact and dimension table as CSV
2. Create database and tables
3. Bulk insert csv file into tables
4. Ready to query and analyze

	investment_name_id	investment_name
1	1	Nvidia
2	2	Amazon
3	3	JPMorgan
4	4	SPY
5	5	USDCAD
6	6	NDX

	investment_type_id	investment_type
1	1	Stock
2	2	ETF
3	3	Forex
4	4	Market

	investment_sector_id	investment_sector
1	1	Tech
2	2	Consumer
3	3	Finance
4	4	Broad Market
5	5	Currency

	time_id	investment_date	investment_month	investment_quarter	investment_year
1	1	2023-07-01 00:00:00.000	7	3	2023
2	2	2023-07-02 00:00:00.000	7	3	2023
3	3	2023-07-03 00:00:00.000	7	3	2023
4	4	2023-07-04 00:00:00.000	7	3	2023
5	5	2023-07-05 00:00:00.000	7	3	2023
6	6	2023-07-06 00:00:00.000	7	3	2023
7	7	2023-07-07 00:00:00.000	7	3	2023
8	8	2023-07-08 00:00:00.000	7	3	2023
9	9	2023-07-09 00:00:00.000	7	3	2023
10	10	2023-07-10 00:00:00.000	7	3	2023

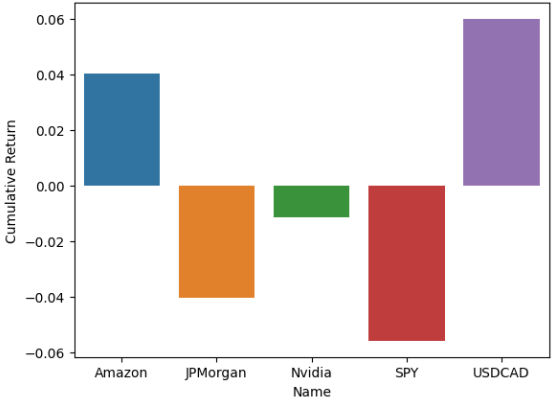
Portfolio Tracking Overview

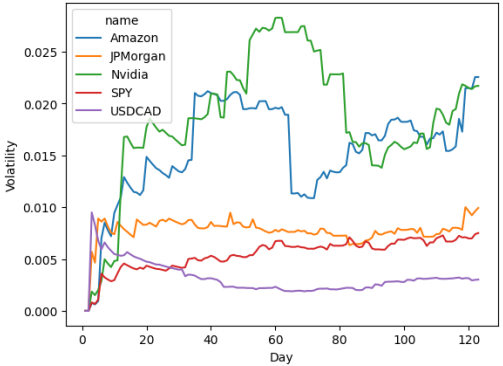
1. Use Queries from SQL to select a time frame for analysis
2. Track portfolio performance
3. Visualize individual investments with Seaborn
4. Invest additional money based on analysis

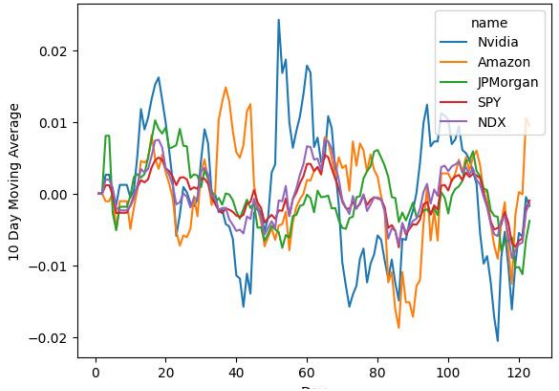
Portfolio Performance
July 2023 – October 2023

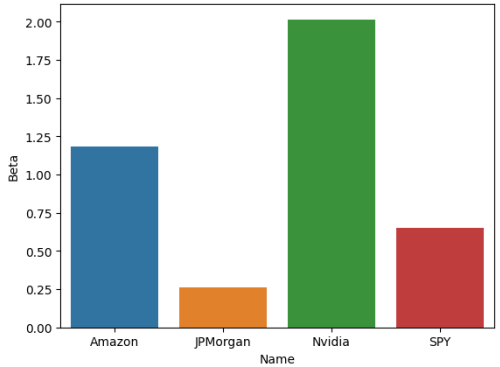
	date	total_value	total_return
0	2023-07-01	100000.000000	0.000000
1	2023-07-02	100132.448305	132.448305
2	2023-07-03	100357.020320	357.020320
3	2023-07-04	100338.361321	338.361321
4	2023-07-05	100049.134483	49.134483
...	...	...	...
118	2023-10-27	95812.480335	-4187.519665
119	2023-10-28	95812.480335	-4187.519665
120	2023-10-29	95800.550811	-4199.449189
121	2023-10-30	97420.222409	-2579.777591
122	2023-10-31	97737.332216	-2262.667784

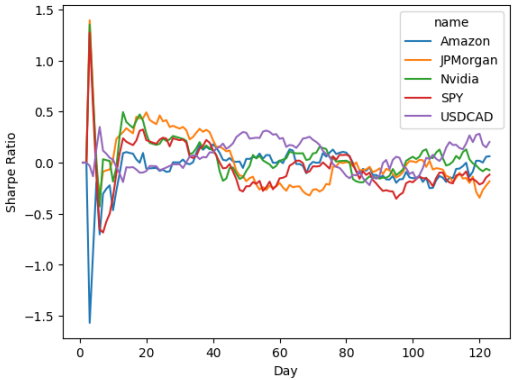
Investments Dashboard
July 2023 – October 2023

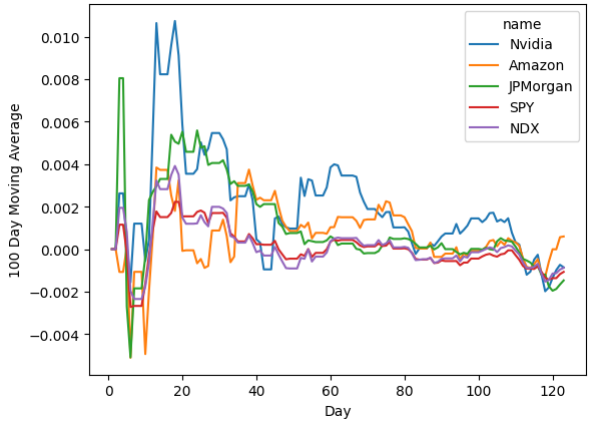




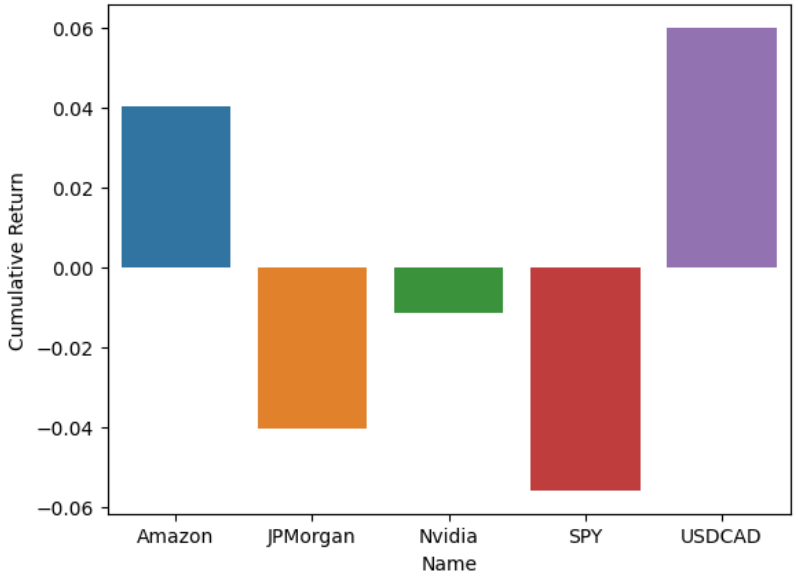


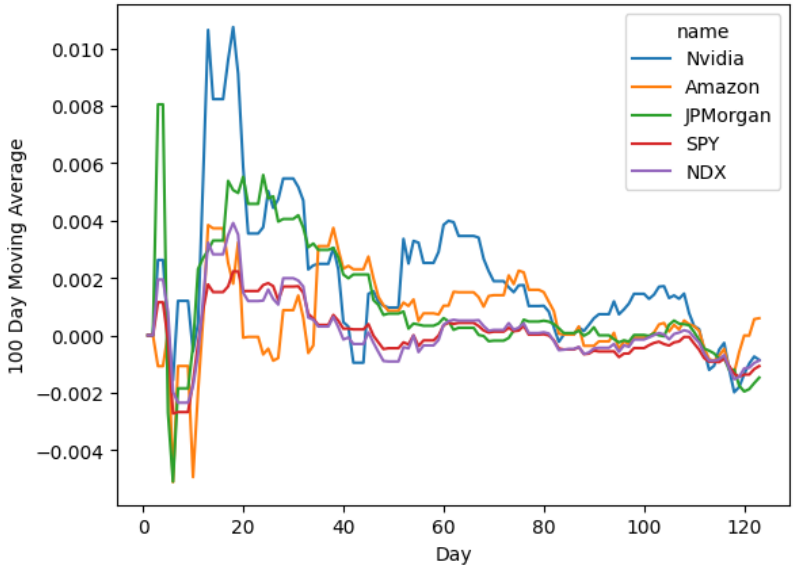






Analyze Investments
July 2023 – October 2023

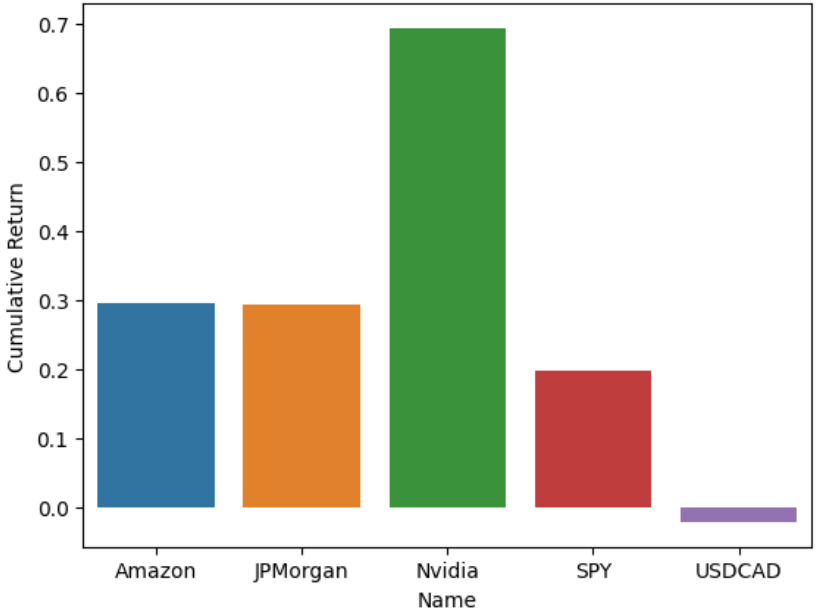


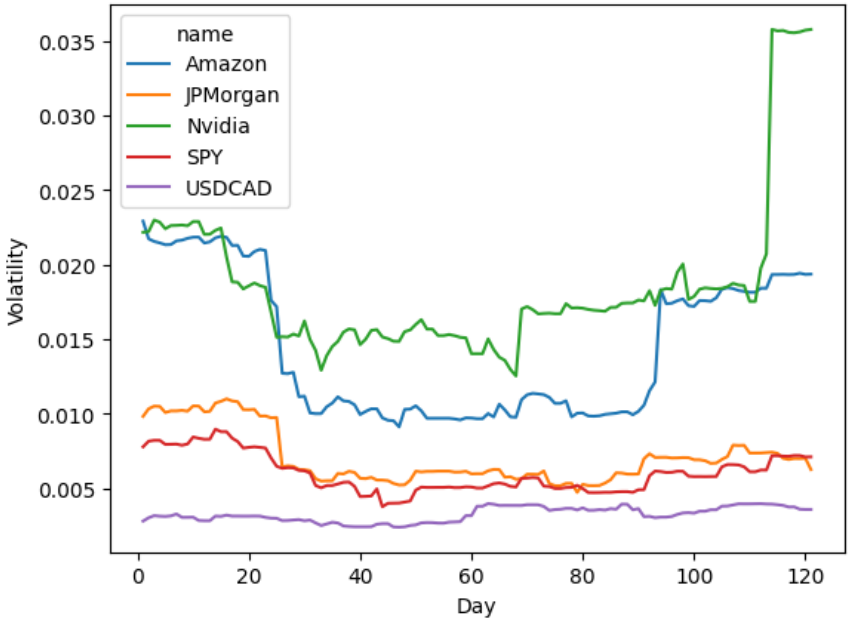


Portfolio Performance Nov 2023 – Feb 2024

	date	total_value	total_return
0	2023-11-01	103445.614674	1708.282458
1	2023-11-02	105043.681685	3306.349469
2	2023-11-03	106305.933007	4568.600791
3	2023-11-04	106305.933007	4568.600791
4	2023-11-05	106299.509418	4562.177201
...	...	...	...
116	2024-02-25	139990.944278	38253.612062
117	2024-02-26	139862.559155	38125.226938
118	2024-02-27	139557.572425	37820.240209
119	2024-02-28	139092.446312	37355.114096
120	2024-02-29	140794.212938	39056.880722

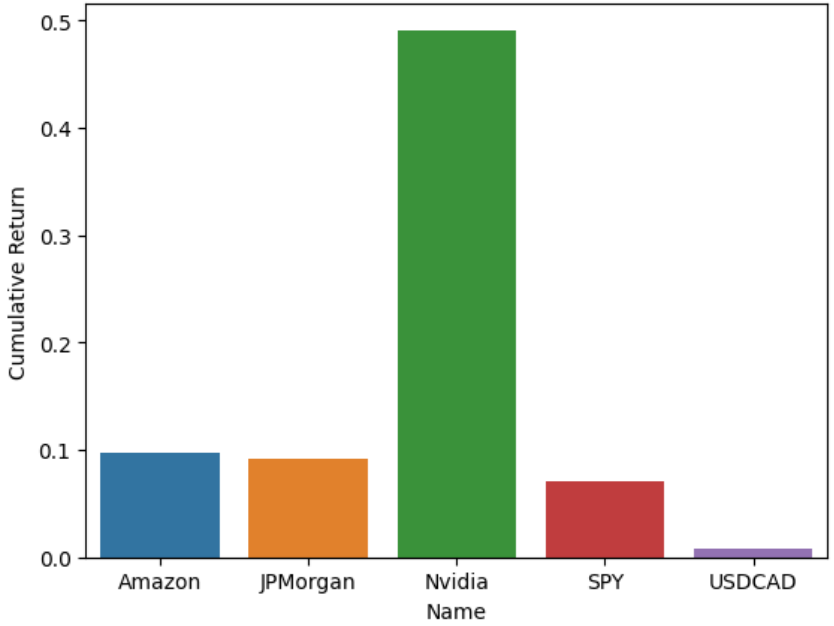
Analyze Investments
July 2023 – October 2023



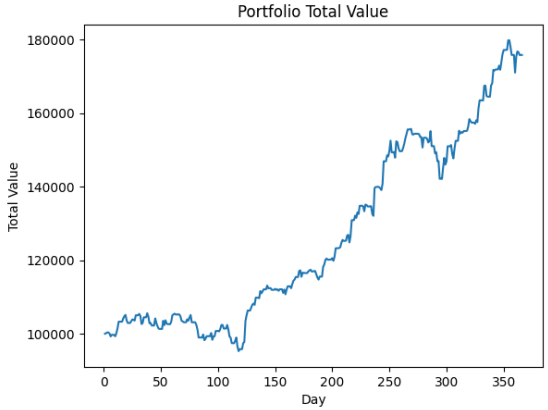


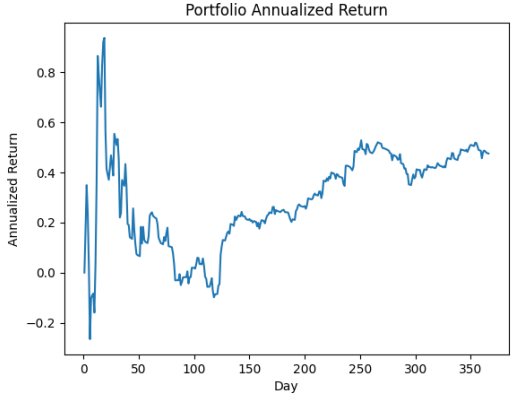
Analyze Performance March 2024 – June 2024

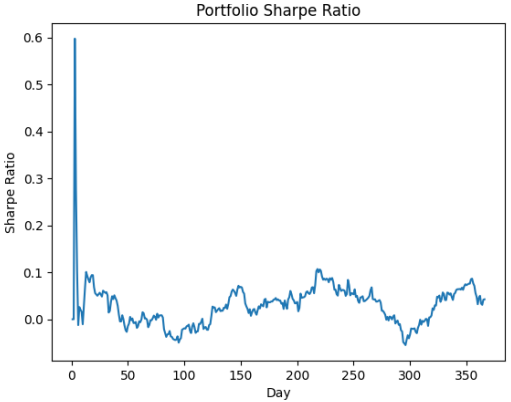
	date	total_value	total_return
0	2024-03-01	146909.476457	2115.263518
1	2024-03-02	146909.476457	2115.263518
2	2024-03-03	146901.294027	2107.081088
3	2024-03-04	148520.731117	3726.518179
4	2024-03-05	148234.805763	3440.592825
...	...	...	...
117	2024-06-26	176794.256464	32000.043526
118	2024-06-27	176591.943752	31797.730813
119	2024-06-28	175823.804995	31029.592057
120	2024-06-29	175823.804995	31029.592057
121	2024-06-30	175840.858097	31046.645159

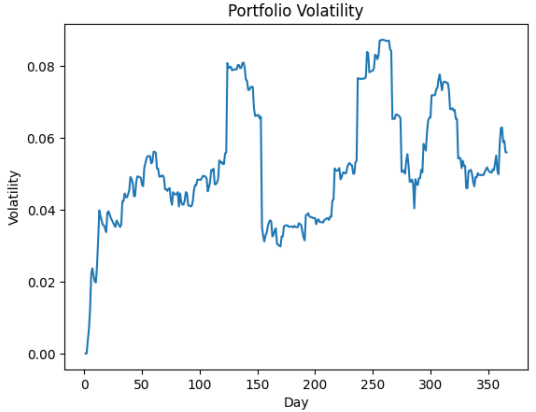


Overall Portfolio Performance Dashboard









Thank you!

Questions?