PREDICTION ANALYSIS USING WINTER’S METHOD

01

Introduction

Introduction

Prediction Analysis refers to the use of statistical techniques and algorithms to make predictions about future events or trends based on historical data. It helps in forecasting by identifying patterns and trends, allowing businesses or researchers to make informed decisions.

Winter's method is a time series forecasting technique that accounts for **trend** and **seasonality** in the data. It is particularly useful when the data exhibits regular patterns over time.

Introduction

There are two variations:

- **Additive**: Suitable when the seasonal variation is relatively constant.
- **Multiplicative**: Used when the seasonal variation increases over time.

Key Components:

1. **Level (L)** – Represents the baseline value in the time series.
2. **Trend (T)** – The direction and rate of change over time.
3. **Seasonality (S)** – Repeated patterns that occur at regular intervals (e.g., monthly or yearly).

02

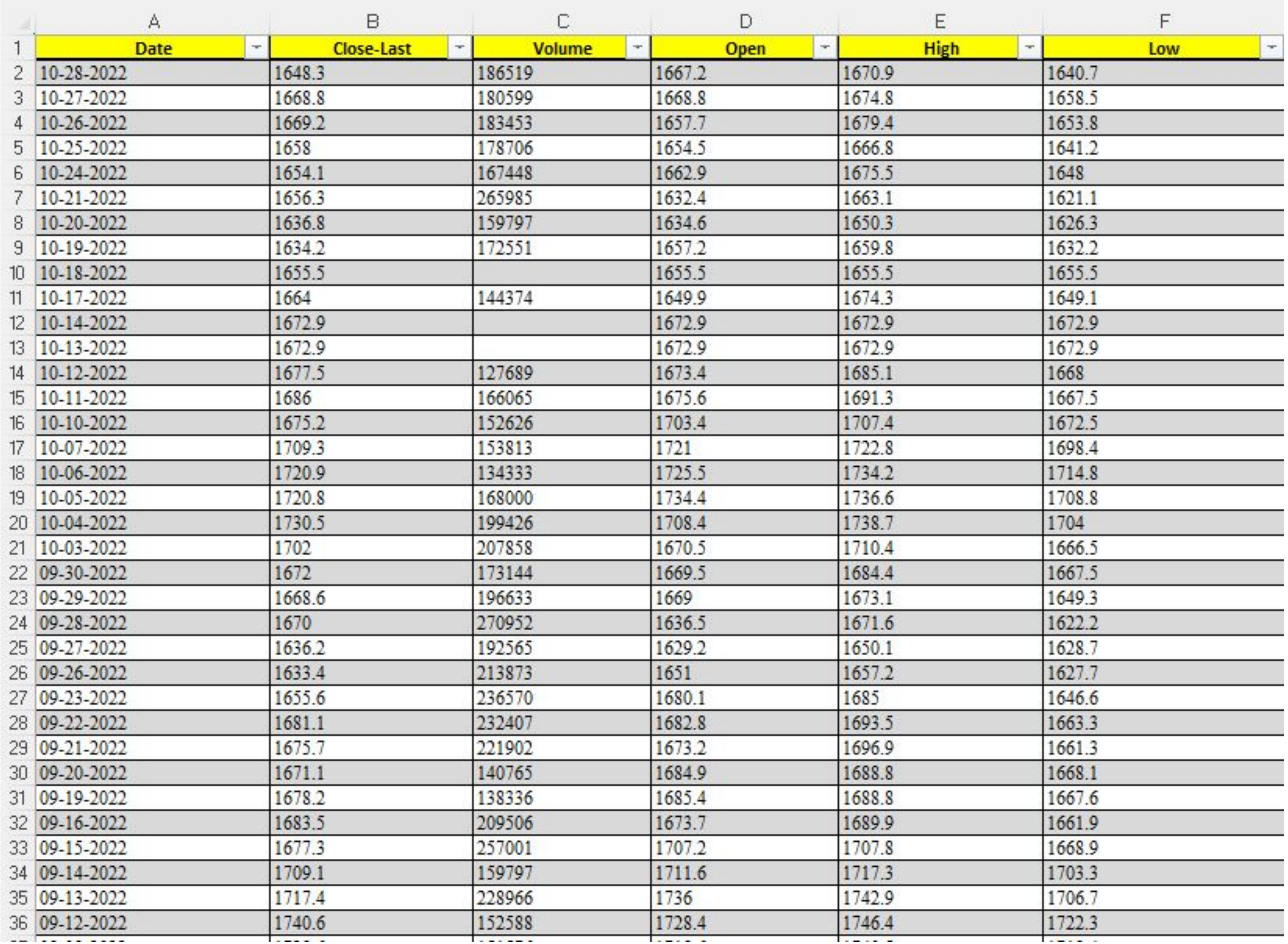
Problem Statement

Problem Statement

How accurately can the Winters' method forecast future gold prices by accounting for both trend and seasonal fluctuations in historical data?

02

Data Description



	A	B	C	D	E	F
1	Date	Close-Last	Volume	Open	High	Low
2	10-28-2022	1648.3	186519	1667.2	1670.9	1640.7
3	10-27-2022	1668.8	180599	1668.8	1674.8	1658.5
4	10-26-2022	1669.2	183453	1657.7	1679.4	1653.8
5	10-25-2022	1658	178706	1654.5	1666.8	1641.2
6	10-24-2022	1654.1	167448	1662.9	1675.5	1648
7	10-21-2022	1656.3	265985	1632.4	1663.1	1621.1
8	10-20-2022	1636.8	159797	1634.6	1650.3	1626.3
9	10-19-2022	1634.2	172551	1657.2	1659.8	1632.2
10	10-18-2022	1655.5		1655.5	1655.5	1655.5
11	10-17-2022	1664	144374	1649.9	1674.3	1649.1
12	10-14-2022	1672.9		1672.9	1672.9	1672.9
13	10-13-2022	1672.9		1672.9	1672.9	1672.9
14	10-12-2022	1677.5	127689	1673.4	1685.1	1668
15	10-11-2022	1686	166065	1675.6	1691.3	1667.5
16	10-10-2022	1675.2	152626	1703.4	1707.4	1672.5
17	10-07-2022	1709.3	153813	1721	1722.8	1698.4
18	10-06-2022	1720.9	134333	1725.5	1734.2	1714.8
19	10-05-2022	1720.8	168000	1734.4	1736.6	1708.8
20	10-04-2022	1730.5	199426	1708.4	1738.7	1704
21	10-03-2022	1702	207858	1670.5	1710.4	1666.5
22	09-30-2022	1672	173144	1669.5	1684.4	1667.5
23	09-29-2022	1668.6	196633	1669	1673.1	1649.3
24	09-28-2022	1670	270952	1636.5	1671.6	1622.2
25	09-27-2022	1636.2	192565	1629.2	1650.1	1628.7
26	09-26-2022	1633.4	213873	1651	1657.2	1627.7
27	09-23-2022	1655.6	236570	1680.1	1685	1646.6
28	09-22-2022	1681.1	232407	1682.8	1693.5	1663.3
29	09-21-2022	1675.7	221902	1673.2	1696.9	1661.3
30	09-20-2022	1671.1	140765	1684.9	1688.8	1668.1
31	09-19-2022	1678.2	138336	1685.4	1688.8	1667.6
32	09-16-2022	1683.5	209506	1673.7	1689.9	1661.9
33	09-15-2022	1677.3	257001	1707.2	1707.8	1668.9
34	09-14-2022	1709.1	159797	1711.6	1717.3	1703.3
35	09-13-2022	1717.4	228966	1736	1742.9	1706.7
36	09-12-2022	1740.6	152588	1728.4	1746.4	1722.3

Data Description

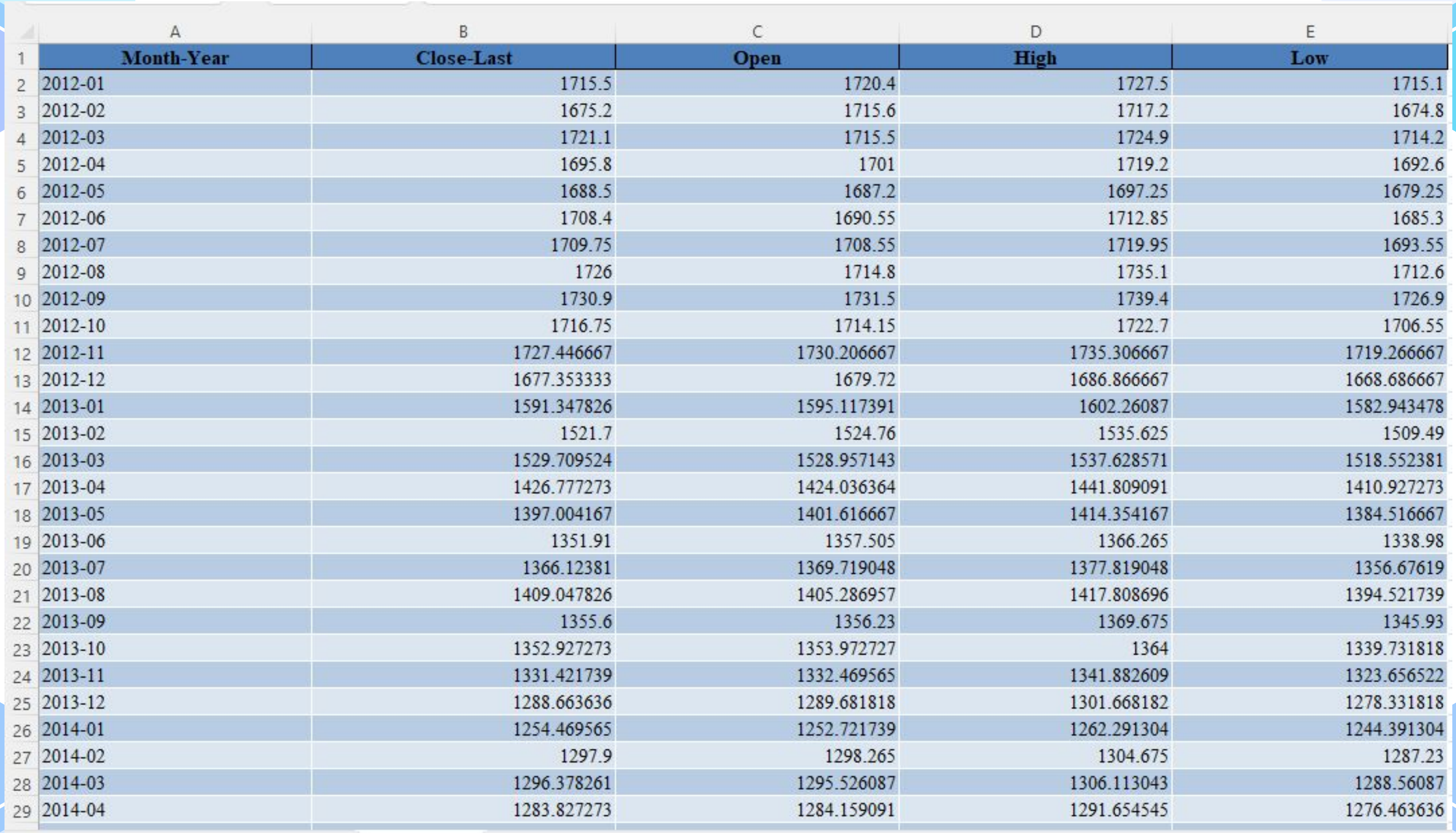
The original dataset was taken from Kaggle.com. The dataset contained 2548 rows of data, and 6 columns.

The ‘Date’ column showed the dates of the records (particular dates from 2012 - 2022); ‘Close - Last’ column shows the closing price of gold on that date; ‘Volume’ showed the trading volume; ‘Open’ shows the opening price for gold on that date; ‘High’ shows the highest price recorded that day; ‘Low’ shows the lowest price recorded that day.

We cleaned the dataset to our desired needs.

03

Data Pre - Processing

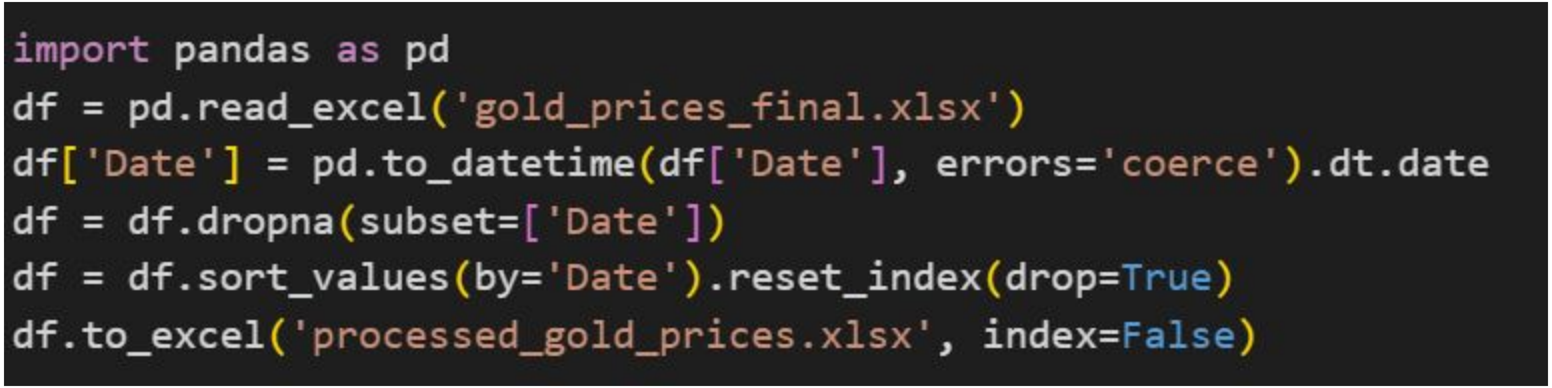


	A	B	C	D	E
1	Month-Year	Close-Last	Open	High	Low
2	2012-01	1715.5	1720.4	1727.5	1715.1
3	2012-02	1675.2	1715.6	1717.2	1674.8
4	2012-03	1721.1	1715.5	1724.9	1714.2
5	2012-04	1695.8	1701	1719.2	1692.6
6	2012-05	1688.5	1687.2	1697.25	1679.25
7	2012-06	1708.4	1690.55	1712.85	1685.3
8	2012-07	1709.75	1708.55	1719.95	1693.55
9	2012-08	1726	1714.8	1735.1	1712.6
10	2012-09	1730.9	1731.5	1739.4	1726.9
11	2012-10	1716.75	1714.15	1722.7	1706.55
12	2012-11	1727.446667	1730.206667	1735.306667	1719.266667
13	2012-12	1677.353333	1679.72	1686.866667	1668.686667
14	2013-01	1591.347826	1595.117391	1602.26087	1582.943478
15	2013-02	1521.7	1524.76	1535.625	1509.49
16	2013-03	1529.709524	1528.957143	1537.628571	1518.552381
17	2013-04	1426.777273	1424.036364	1441.809091	1410.927273
18	2013-05	1397.004167	1401.616667	1414.354167	1384.516667
19	2013-06	1351.91	1357.505	1366.265	1338.98
20	2013-07	1366.12381	1369.719048	1377.819048	1356.67619
21	2013-08	1409.047826	1405.286957	1417.808696	1394.521739
22	2013-09	1355.6	1356.23	1369.675	1345.93
23	2013-10	1352.927273	1353.972727	1364	1339.731818
24	2013-11	1331.421739	1332.469565	1341.882609	1323.656522
25	2013-12	1288.663636	1289.681818	1301.668182	1278.331818
26	2014-01	1254.469565	1252.721739	1262.291304	1244.391304
27	2014-02	1297.9	1298.265	1304.675	1287.23
28	2014-03	1296.378261	1295.526087	1306.113043	1288.56087
29	2014-04	1283.827273	1284.159091	1291.654545	1276.463636

Data Pre - Processing

The data was modified to simplify calculations.
The 'Date' column was converted to 'datetime' format using Python.
The data was brought down to 145 rows, while selecting a date's data for each month of each year. The data was then sorted using Excel.

Data Pre - Processing



```python
import pandas as pd
df = pd.read_excel('gold_prices_final.xlsx')
df['Date'] = pd.to_datetime(df['Date'], errors='coerce').dt.date
df = df.dropna(subset=['Date'])
df = df.sort_values(by='Date').reset_index(drop=True)
df.to_excel('processed_gold_prices.xlsx', index=False)
```

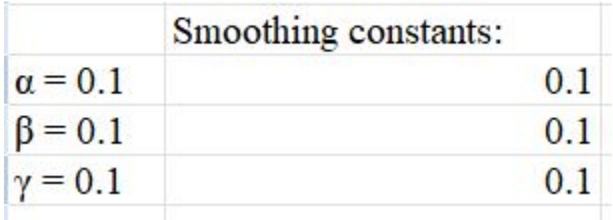
The Python code used to transform the data into the ‘datetime’ format.

04

Implementation & Analysis

Implementation

- The **'Close-Last'** column was used mainly in the forecasting.
- Once the dataset was pre-processed, we used the **Multiplicative method of Winter's Forecasting Method**.
- The **Smoothing Constants** were set to the following: **α = 0.1, β = 0.1, γ = 0.1**



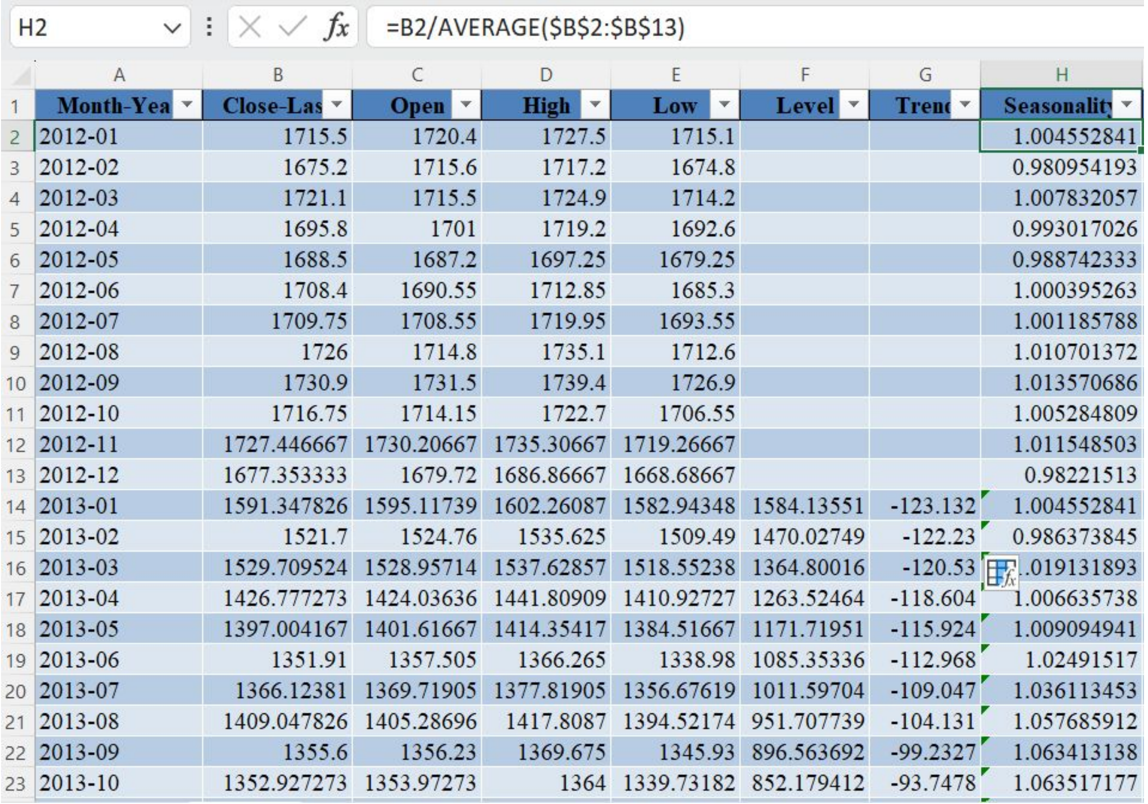
	Smoothing constants:
α = 0.1	0.1
β = 0.1	0.1
γ = 0.1	0.1

Implementation

- The **Base Seasonality** was taken as:

= **Close-Last / AVERAGE(Close-Last for first 12 rows)**

Implementation



H2

=B2/AVERAGE($B$2:$B$13)

	A	B	C	D	E	F	G	H
1	Month-Yea	Close-Las	Open	High	Low	Level	Trenc	Seasonality
2	2012-01	1715.5	1720.4	1727.5	1715.1			1.004552841
3	2012-02	1675.2	1715.6	1717.2	1674.8			0.980954193
4	2012-03	1721.1	1715.5	1724.9	1714.2			1.007832057
5	2012-04	1695.8	1701	1719.2	1692.6			0.993017026
6	2012-05	1688.5	1687.2	1697.25	1679.25			0.988742333
7	2012-06	1708.4	1690.55	1712.85	1685.3			1.000395263
8	2012-07	1709.75	1708.55	1719.95	1693.55			1.001185788
9	2012-08	1726	1714.8	1735.1	1712.6			1.010701372
10	2012-09	1730.9	1731.5	1739.4	1726.9			1.013570686
11	2012-10	1716.75	1714.15	1722.7	1706.55			1.005284809
12	2012-11	1727.446667	1730.20667	1735.30667	1719.26667			1.011548503
13	2012-12	1677.353333	1679.72	1686.86667	1668.68667			0.98221513
14	2013-01	1591.347826	1595.11739	1602.26087	1582.94348	1584.13551	-123.132	1.004552841
15	2013-02	1521.7	1524.76	1535.625	1509.49	1470.02749	-122.23	0.986373845
16	2013-03	1529.709524	1528.95714	1537.62857	1518.55238	1364.80016	-120.53	.019131893
17	2013-04	1426.777273	1424.03636	1441.80909	1410.92727	1263.52464	-118.604	1.006635738
18	2013-05	1397.004167	1401.61667	1414.35417	1384.51667	1171.71951	-115.924	1.009094941
19	2013-06	1351.91	1357.505	1366.265	1338.98	1085.35336	-112.968	1.02491517
20	2013-07	1366.12381	1369.71905	1377.81905	1356.67619	1011.59704	-109.047	1.036113453
21	2013-08	1409.047826	1405.28696	1417.8087	1394.52174	951.707739	-104.131	1.057685912
22	2013-09	1355.6	1356.23	1369.675	1345.93	896.563692	-99.2327	1.063413138
23	2013-10	1352.927273	1353.97273	1364	1339.73182	852.179412	-93.7478	1.063517177

Implementation

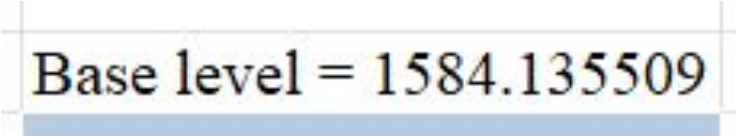
- The first 12 months were left as a base.

- The **Base Level** for 01/2013 was taken as

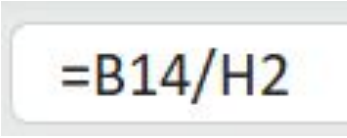
= **Gold Price for 01/2013 / Base Seasonality**

- The next **Levels** were taken as

= **alpha * (Close-Last / Seasonality of 12 months before) + (1 - alpha) * (Level_Previous + Trend_Previous)**

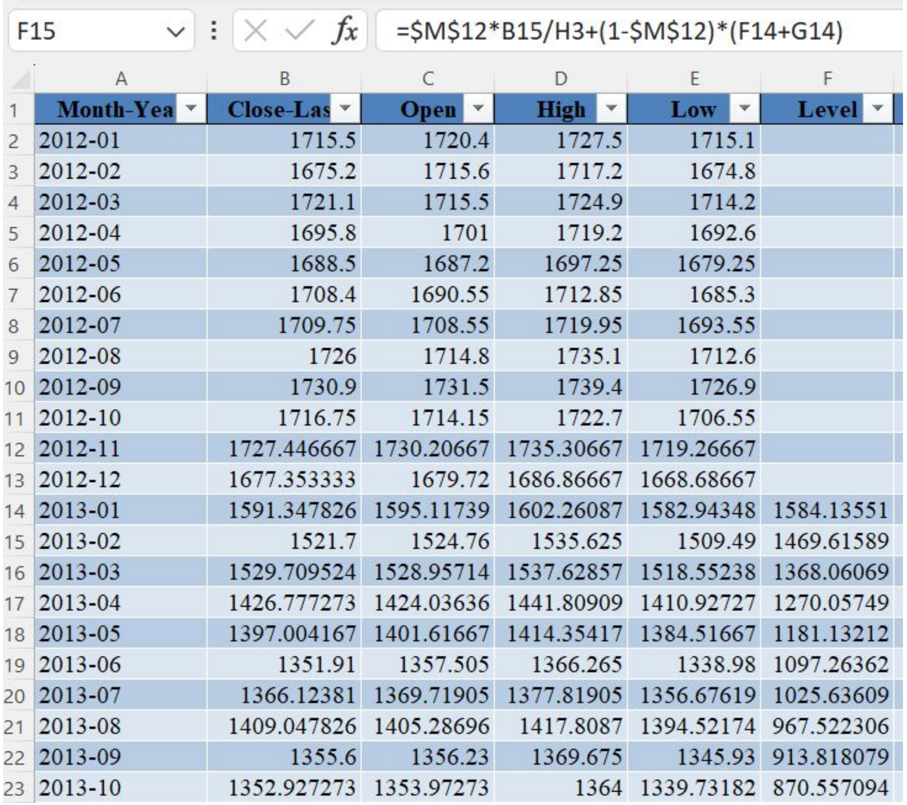


Base level = 1584.135509



=B14/H2

Implementation



F15

=$M$12*B15/H3+(1-$M$12)*(F14+G14)

	A	B	C	D	E	F
1	Month-Yea	Close-Las	Open	High	Low	Level
2	2012-01	1715.5	1720.4	1727.5	1715.1	
3	2012-02	1675.2	1715.6	1717.2	1674.8	
4	2012-03	1721.1	1715.5	1724.9	1714.2	
5	2012-04	1695.8	1701	1719.2	1692.6	
6	2012-05	1688.5	1687.2	1697.25	1679.25	
7	2012-06	1708.4	1690.55	1712.85	1685.3	
8	2012-07	1709.75	1708.55	1719.95	1693.55	
9	2012-08	1726	1714.8	1735.1	1712.6	
10	2012-09	1730.9	1731.5	1739.4	1726.9	
11	2012-10	1716.75	1714.15	1722.7	1706.55	
12	2012-11	1727.446667	1730.20667	1735.30667	1719.26667	
13	2012-12	1677.353333	1679.72	1686.86667	1668.68667	
14	2013-01	1591.347826	1595.11739	1602.26087	1582.94348	1584.13551
15	2013-02	1521.7	1524.76	1535.625	1509.49	1469.61589
16	2013-03	1529.709524	1528.95714	1537.62857	1518.55238	1368.06069
17	2013-04	1426.777273	1424.03636	1441.80909	1410.92727	1270.05749
18	2013-05	1397.004167	1401.61667	1414.35417	1384.51667	1181.13212
19	2013-06	1351.91	1357.505	1366.265	1338.98	1097.26362
20	2013-07	1366.12381	1369.71905	1377.81905	1356.67619	1025.63609
21	2013-08	1409.047826	1405.28696	1417.8087	1394.52174	967.522306
22	2013-09	1355.6	1356.23	1369.675	1345.93	913.818079
23	2013-10	1352.927273	1353.97273	1364	1339.73182	870.557094

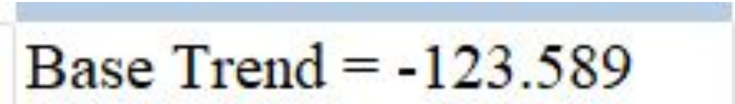
Implementation

- The **Base Trend** was taken as the formula:

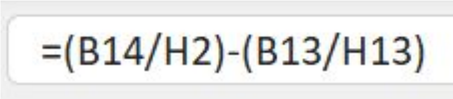
= **(Current Data / Base Seasonality) – (Previous Data / Seasonality for 12/2012)**

- Rest of the Trends were taken as

= beta * (Level_Today – Level_Previous) + (1 – beta) * Trend_Previous

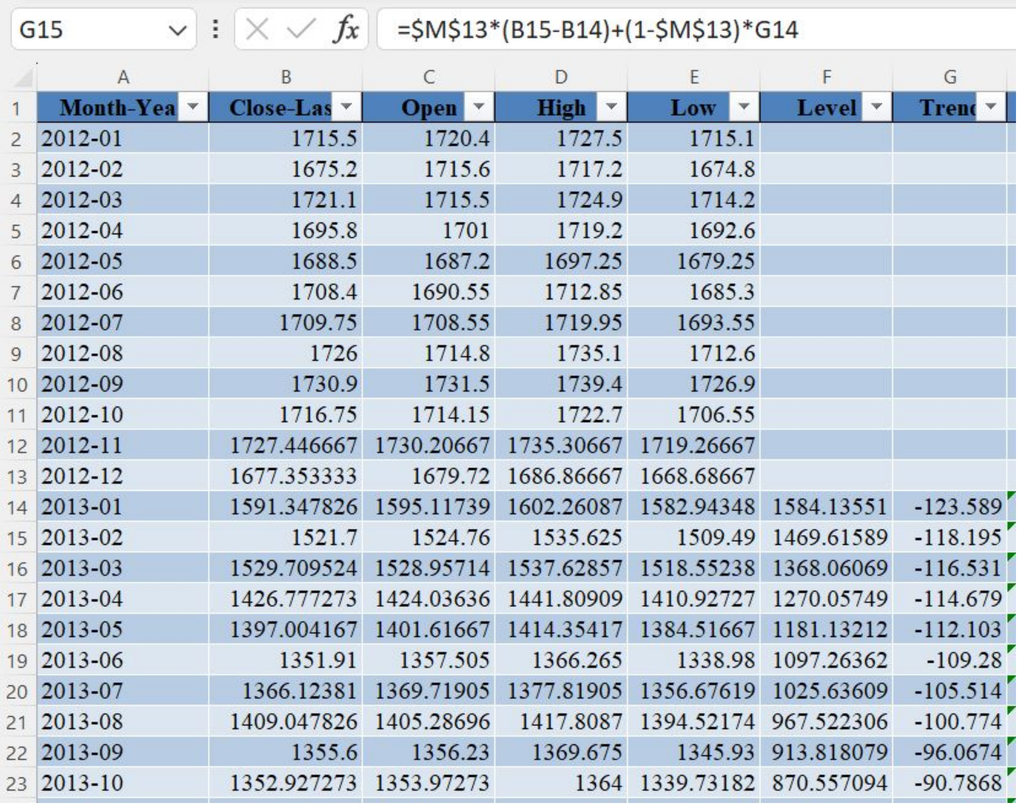


Base Trend = -123.589



=(B14/H2)-(B13/H13)

Implementation



G15 =$M$13*(B15-B14)+(1-$M$13)*G14

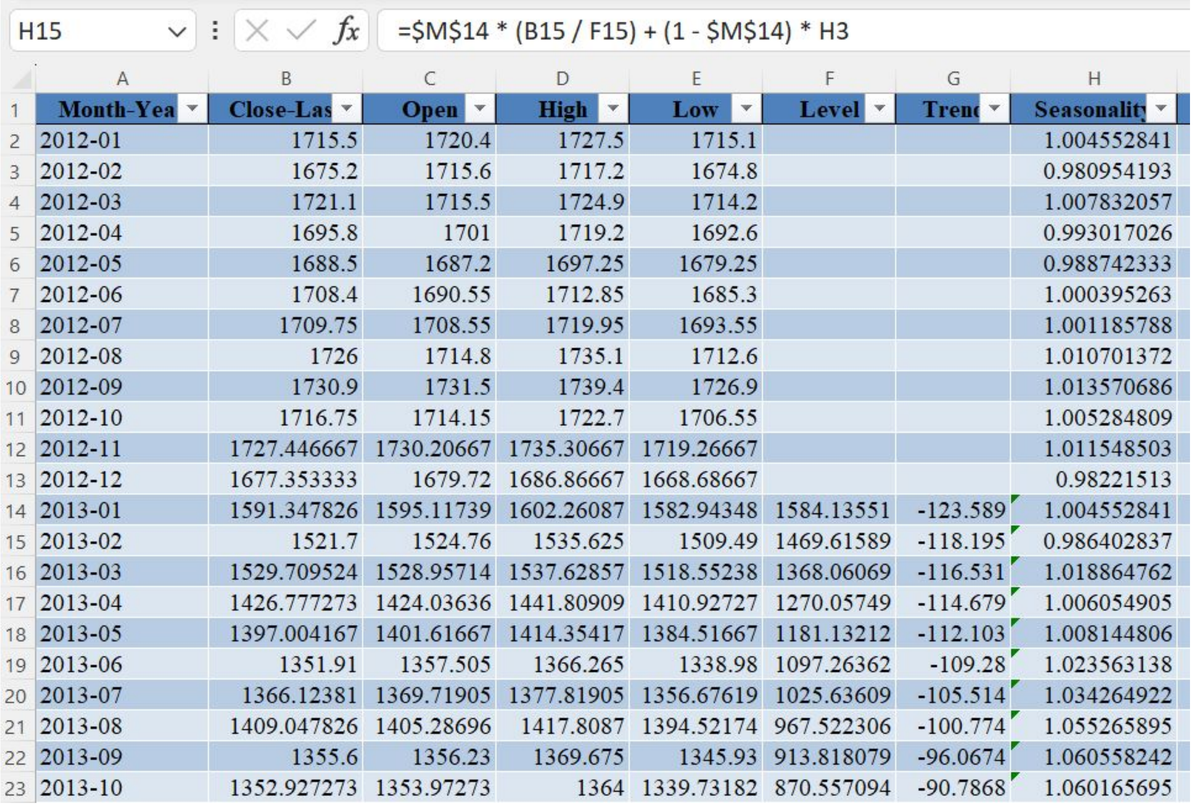
	A	B	C	D	E	F	G
1	Month-Yea	Close-Las	Open	High	Low	Level	Trenc
2	2012-01	1715.5	1720.4	1727.5	1715.1		
3	2012-02	1675.2	1715.6	1717.2	1674.8		
4	2012-03	1721.1	1715.5	1724.9	1714.2		
5	2012-04	1695.8	1701	1719.2	1692.6		
6	2012-05	1688.5	1687.2	1697.25	1679.25		
7	2012-06	1708.4	1690.55	1712.85	1685.3		
8	2012-07	1709.75	1708.55	1719.95	1693.55		
9	2012-08	1726	1714.8	1735.1	1712.6		
10	2012-09	1730.9	1731.5	1739.4	1726.9		
11	2012-10	1716.75	1714.15	1722.7	1706.55		
12	2012-11	1727.446667	1730.20667	1735.30667	1719.26667		
13	2012-12	1677.353333	1679.72	1686.86667	1668.68667		
14	2013-01	1591.347826	1595.11739	1602.26087	1582.94348	1584.13551	-123.589
15	2013-02	1521.7	1524.76	1535.625	1509.49	1469.61589	-118.195
16	2013-03	1529.709524	1528.95714	1537.62857	1518.55238	1368.06069	-116.531
17	2013-04	1426.777273	1424.03636	1441.80909	1410.92727	1270.05749	-114.679
18	2013-05	1397.004167	1401.61667	1414.35417	1384.51667	1181.13212	-112.103
19	2013-06	1351.91	1357.505	1366.265	1338.98	1097.26362	-109.28
20	2013-07	1366.12381	1369.71905	1377.81905	1356.67619	1025.63609	-105.514
21	2013-08	1409.047826	1405.28696	1417.8087	1394.52174	967.522306	-100.774
22	2013-09	1355.6	1356.23	1369.675	1345.93	913.818079	-96.0674
23	2013-10	1352.927273	1353.97273	1364	1339.73182	870.557094	-90.7868

Implementation

- Rest of the **Seasonalities** were taken as

= **gamma * (Close-Last / Level_Today) + (1 - gamma) * Seasonality 12 months before**

Implementation



```
=$M$14 * (B15 / F15) + (1 - $M$14) * H3
```

	A	B	C	D	E	F	G	H
1	Month-Yea	Close-Las	Open	High	Low	Level	Trenc	Seasonality
2	2012-01	1715.5	1720.4	1727.5	1715.1			1.004552841
3	2012-02	1675.2	1715.6	1717.2	1674.8			0.980954193
4	2012-03	1721.1	1715.5	1724.9	1714.2			1.007832057
5	2012-04	1695.8	1701	1719.2	1692.6			0.993017026
6	2012-05	1688.5	1687.2	1697.25	1679.25			0.988742333
7	2012-06	1708.4	1690.55	1712.85	1685.3			1.000395263
8	2012-07	1709.75	1708.55	1719.95	1693.55			1.001185788
9	2012-08	1726	1714.8	1735.1	1712.6			1.010701372
10	2012-09	1730.9	1731.5	1739.4	1726.9			1.013570686
11	2012-10	1716.75	1714.15	1722.7	1706.55			1.005284809
12	2012-11	1727.446667	1730.20667	1735.30667	1719.26667			1.011548503
13	2012-12	1677.353333	1679.72	1686.86667	1668.68667			0.98221513
14	2013-01	1591.347826	1595.11739	1602.26087	1582.94348	1584.13551	-123.589	1.004552841
15	2013-02	1521.7	1524.76	1535.625	1509.49	1469.61589	-118.195	0.986402837
16	2013-03	1529.709524	1528.95714	1537.62857	1518.55238	1368.06069	-116.531	1.018864762
17	2013-04	1426.777273	1424.03636	1441.80909	1410.92727	1270.05749	-114.679	1.006054905
18	2013-05	1397.004167	1401.61667	1414.35417	1384.51667	1181.13212	-112.103	1.008144806
19	2013-06	1351.91	1357.505	1366.265	1338.98	1097.26362	-109.28	1.023563138
20	2013-07	1366.12381	1369.71905	1377.81905	1356.67619	1025.63609	-105.514	1.034264922
21	2013-08	1409.047826	1405.28696	1417.8087	1394.52174	967.522306	-100.774	1.055265895
22	2013-09	1355.6	1356.23	1369.675	1345.93	913.818079	-96.0674	1.060558242
23	2013-10	1352.927273	1353.97273	1364	1339.73182	870.557094	-90.7868	1.060165695

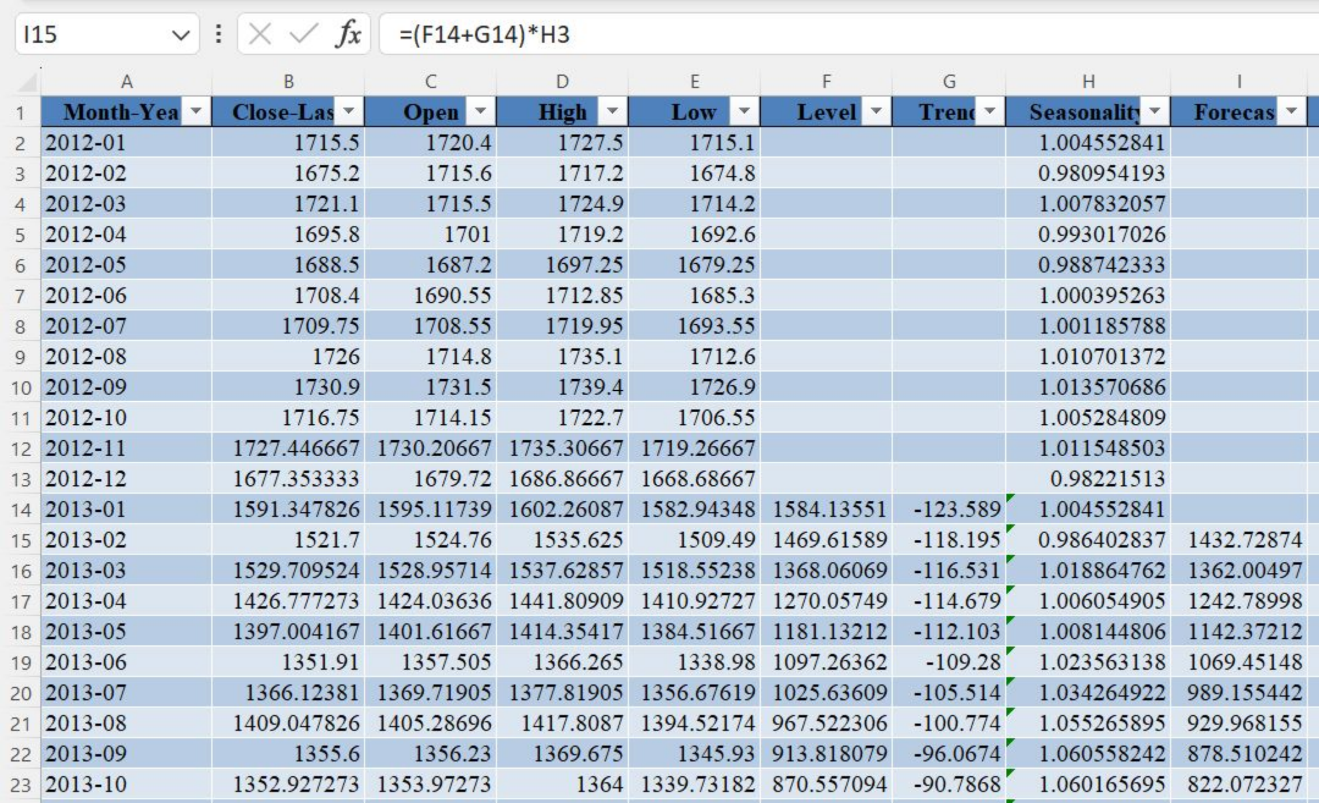
Analysis

- We start forecasting the gold prices from 02/2013.
- The formula used is:

= (Level + Trend) x Seasonality of 12 months before

- This formula is then applied in all rows.

Analysis



I15 =(F14+G14)*H3

	A	B	C	D	E	F	G	H	I
1	Month-Yea	Close-Las	Open	High	Low	Level	Trenc	Seasonality	Forecas
2	2012-01	1715.5	1720.4	1727.5	1715.1			1.004552841	
3	2012-02	1675.2	1715.6	1717.2	1674.8			0.980954193	
4	2012-03	1721.1	1715.5	1724.9	1714.2			1.007832057	
5	2012-04	1695.8	1701	1719.2	1692.6			0.993017026	
6	2012-05	1688.5	1687.2	1697.25	1679.25			0.988742333	
7	2012-06	1708.4	1690.55	1712.85	1685.3			1.000395263	
8	2012-07	1709.75	1708.55	1719.95	1693.55			1.001185788	
9	2012-08	1726	1714.8	1735.1	1712.6			1.010701372	
10	2012-09	1730.9	1731.5	1739.4	1726.9			1.013570686	
11	2012-10	1716.75	1714.15	1722.7	1706.55			1.005284809	
12	2012-11	1727.446667	1730.20667	1735.30667	1719.26667			1.011548503	
13	2012-12	1677.353333	1679.72	1686.86667	1668.68667			0.98221513	
14	2013-01	1591.347826	1595.11739	1602.26087	1582.94348	1584.13551	-123.589	1.004552841	
15	2013-02	1521.7	1524.76	1535.625	1509.49	1469.61589	-118.195	0.986402837	1432.72874
16	2013-03	1529.709524	1528.95714	1537.62857	1518.55238	1368.06069	-116.531	1.018864762	1362.00497
17	2013-04	1426.777273	1424.03636	1441.80909	1410.92727	1270.05749	-114.679	1.006054905	1242.78998
18	2013-05	1397.004167	1401.61667	1414.35417	1384.51667	1181.13212	-112.103	1.008144806	1142.37212
19	2013-06	1351.91	1357.505	1366.265	1338.98	1097.26362	-109.28	1.023563138	1069.45148
20	2013-07	1366.12381	1369.71905	1377.81905	1356.67619	1025.63609	-105.514	1.034264922	989.155442
21	2013-08	1409.047826	1405.28696	1417.8087	1394.52174	967.522306	-100.774	1.055265895	929.968155
22	2013-09	1355.6	1356.23	1369.675	1345.93	913.818079	-96.0674	1.060558242	878.510242
23	2013-10	1352.927273	1353.97273	1364	1339.73182	870.557094	-90.7868	1.060165695	822.072327

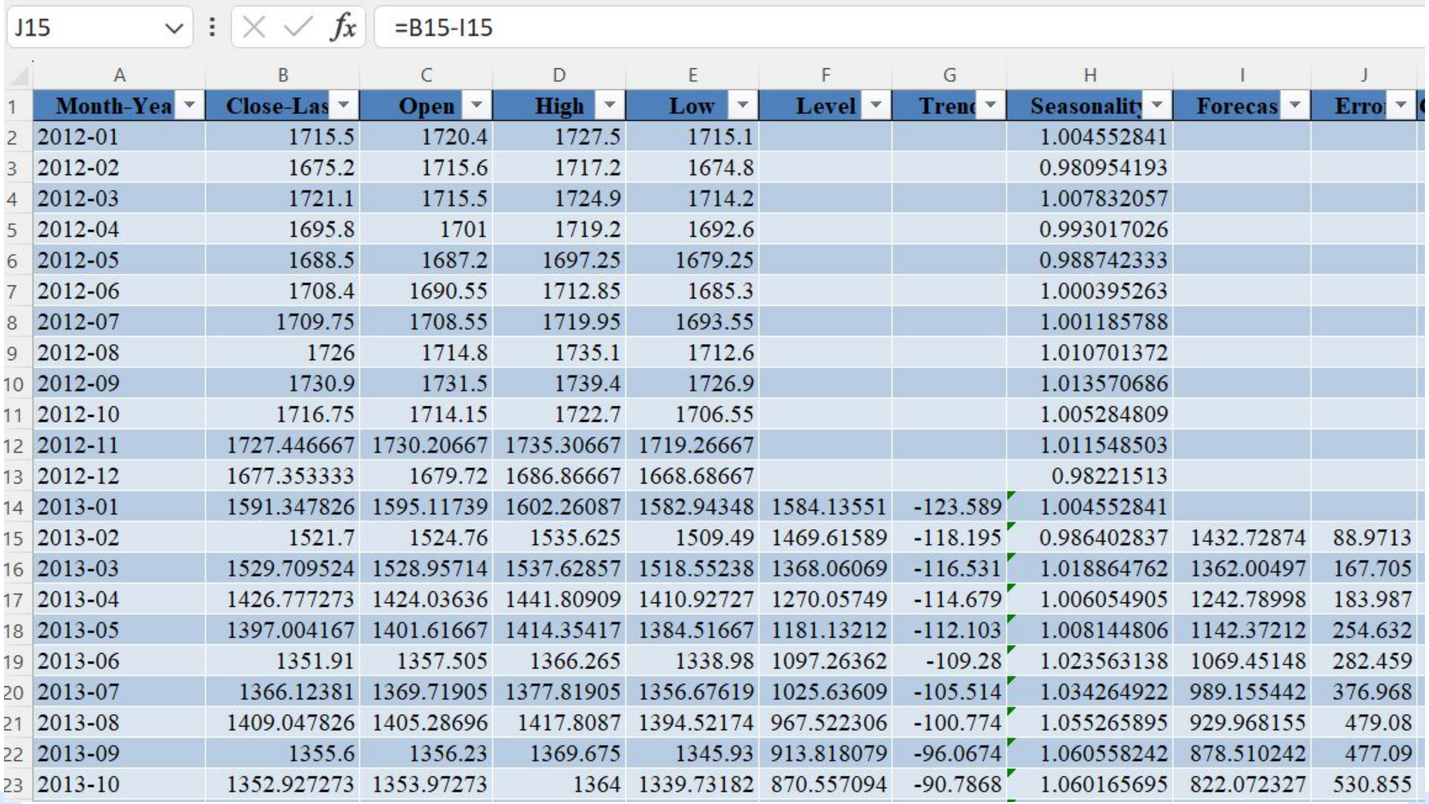
Analysis

- We then found the errors.
- Errors are calculated by:

= **Actual Price - Forecasted Price**

- Initially, the number seems high. It then decreases as we go on.

Analysis



J15

=B15-I15

	A	B	C	D	E	F	G	H	I	J
1	Month-Yea	Close-Las	Open	High	Low	Level	Trend	Seasonality	Forecas	Erro
2	2012-01	1715.5	1720.4	1727.5	1715.1			1.004552841		
3	2012-02	1675.2	1715.6	1717.2	1674.8			0.980954193		
4	2012-03	1721.1	1715.5	1724.9	1714.2			1.007832057		
5	2012-04	1695.8	1701	1719.2	1692.6			0.993017026		
6	2012-05	1688.5	1687.2	1697.25	1679.25			0.988742333		
7	2012-06	1708.4	1690.55	1712.85	1685.3			1.000395263		
8	2012-07	1709.75	1708.55	1719.95	1693.55			1.001185788		
9	2012-08	1726	1714.8	1735.1	1712.6			1.010701372		
10	2012-09	1730.9	1731.5	1739.4	1726.9			1.013570686		
11	2012-10	1716.75	1714.15	1722.7	1706.55			1.005284809		
12	2012-11	1727.446667	1730.20667	1735.30667	1719.26667			1.011548503		
13	2012-12	1677.353333	1679.72	1686.86667	1668.68667			0.98221513		
14	2013-01	1591.347826	1595.11739	1602.26087	1582.94348	1584.13551	-123.589	1.004552841		
15	2013-02	1521.7	1524.76	1535.625	1509.49	1469.61589	-118.195	0.986402837	1432.72874	88.9713
16	2013-03	1529.709524	1528.95714	1537.62857	1518.55238	1368.06069	-116.531	1.018864762	1362.00497	167.705
17	2013-04	1426.777273	1424.03636	1441.80909	1410.92727	1270.05749	-114.679	1.006054905	1242.78998	183.987
18	2013-05	1397.004167	1401.61667	1414.35417	1384.51667	1181.13212	-112.103	1.008144806	1142.37212	254.632
19	2013-06	1351.91	1357.505	1366.265	1338.98	1097.26362	-109.28	1.023563138	1069.45148	282.459
20	2013-07	1366.12381	1369.71905	1377.81905	1356.67619	1025.63609	-105.514	1.034264922	989.155442	376.968
21	2013-08	1409.047826	1405.28696	1417.8087	1394.52174	967.522306	-100.774	1.055265895	929.968155	479.08
22	2013-09	1355.6	1356.23	1369.675	1345.93	913.818079	-96.0674	1.060558242	878.510242	477.09
23	2013-10	1352.927273	1353.97273	1364	1339.73182	870.557094	-90.7868	1.060165695	822.072327	530.855

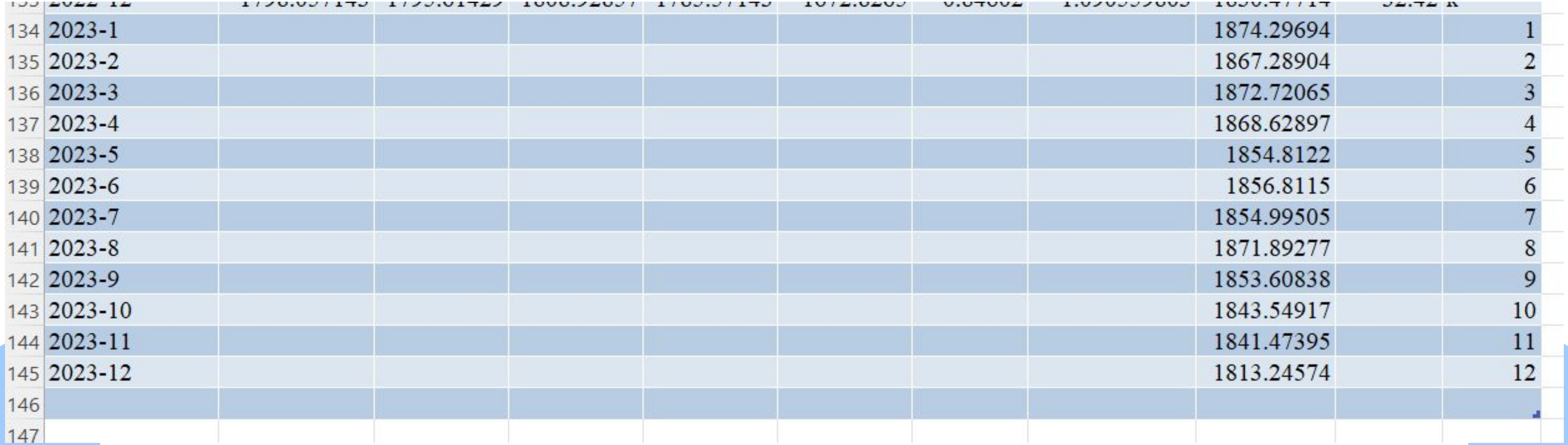
Analysis

2023 Predictions:

- Taken values of ‘K’
- Formula:

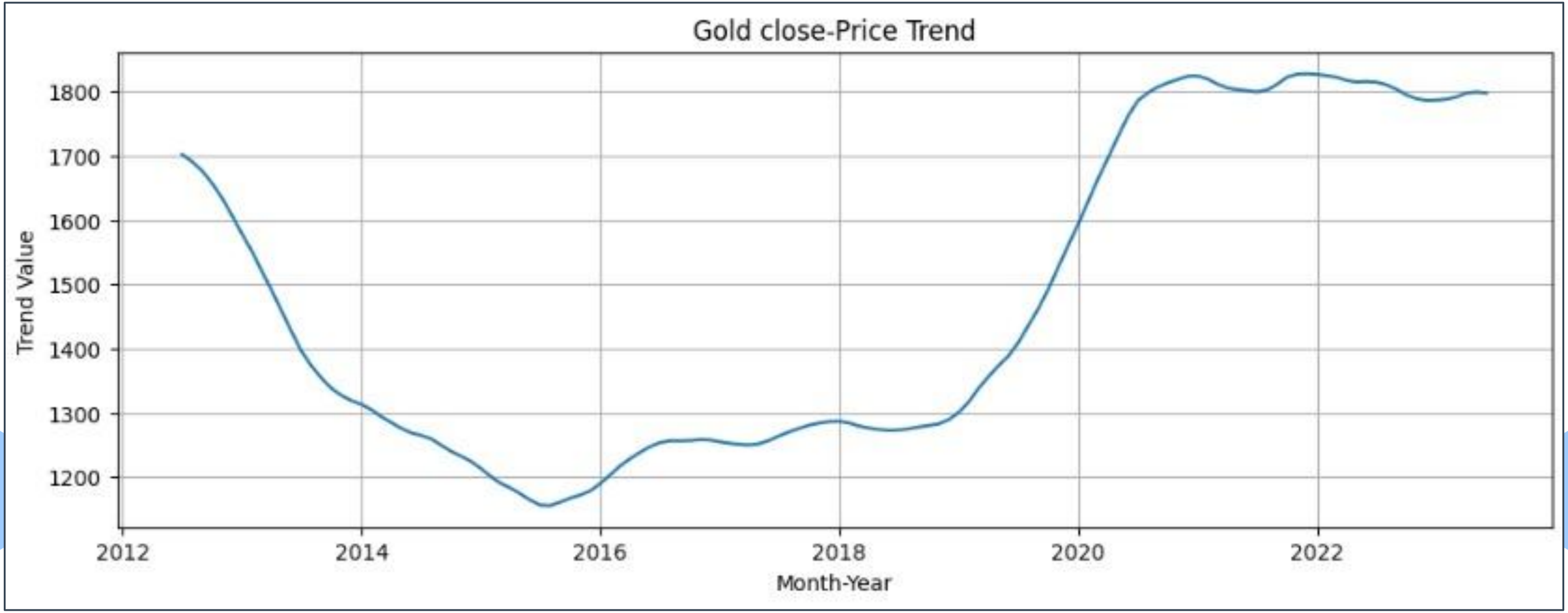
= (Level of Previous Year + k x Trend of Previous Year) x Seasonality of 12 months before

Analysis

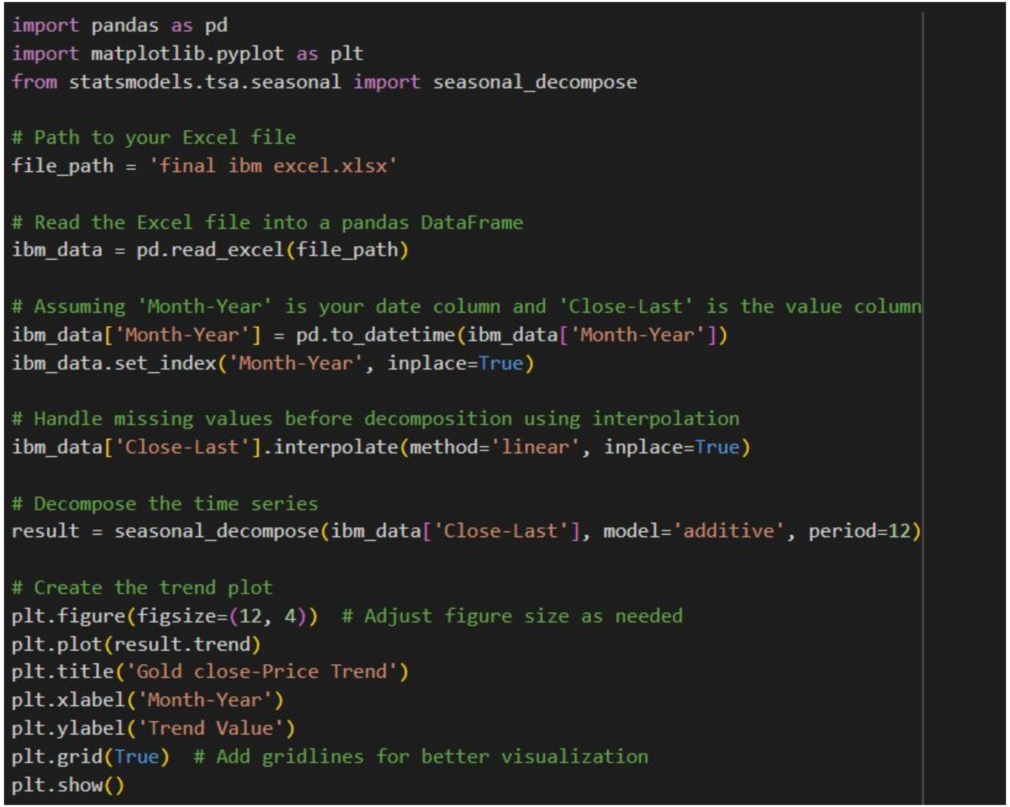


134	2023-1								1874.29694		1
135	2023-2								1867.28904		2
136	2023-3								1872.72065		3
137	2023-4								1868.62897		4
138	2023-5								1854.8122		5
139	2023-6								1856.8115		6
140	2023-7								1854.99505		7
141	2023-8								1871.89277		8
142	2023-9								1853.60838		9
143	2023-10								1843.54917		10
144	2023-11								1841.47395		11
145	2023-12								1813.24574		12
146											
147											

Trend Graph

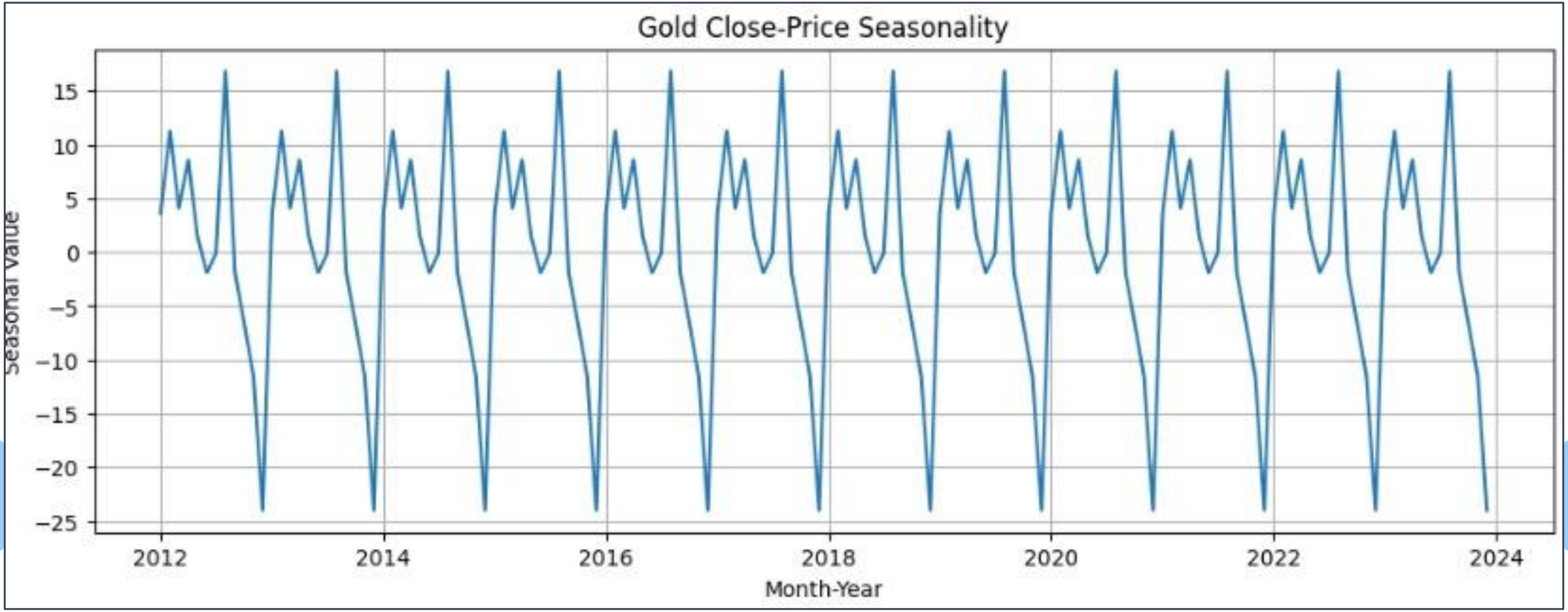

Gold close-Price Trend
Trend Value
1800
1700
1600
1500
1400
1300
1200
2012
2014
2016
2018
2020
2022
Month-Year

Trend Graph Code

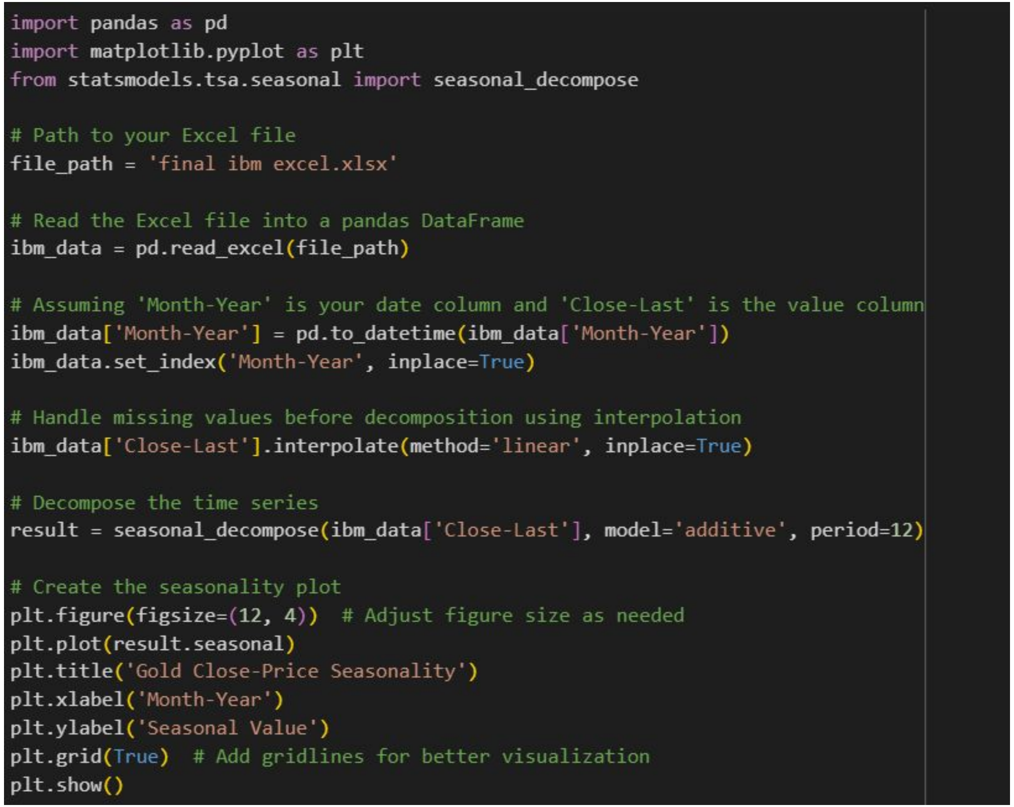


```
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.tsa.seasonal import seasonal_decompose

# Path to your Excel file
file_path = 'final ibm excel.xlsx'

# Read the Excel file into a pandas DataFrame
ibm_data = pd.read_excel(file_path)

# Assuming 'Month-Year' is your date column and 'Close-Last' is the value column
ibm_data['Month-Year'] = pd.to_datetime(ibm_data['Month-Year'])
ibm_data.set_index('Month-Year', inplace=True)

# Handle missing values before decomposition using interpolation
ibm_data['Close-Last'].interpolate(method='linear', inplace=True)

# Decompose the time series
result = seasonal_decompose(ibm_data['Close-Last'], model='additive', period=12)

# Create the trend plot
plt.figure(figsize=(12, 4))  # Adjust figure size as needed
plt.plot(result.trend)
plt.title('Gold close-Price Trend')
plt.xlabel('Month-Year')
plt.ylabel('Trend Value')
plt.grid(True)  # Add gridlines for better visualization
plt.show()
```

Seasonality Graph



Seasonality Graph Code



```
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.tsa.seasonal import seasonal_decompose

# Path to your Excel file
file_path = 'final ibm excel.xlsx'

# Read the Excel file into a pandas DataFrame
ibm_data = pd.read_excel(file_path)

# Assuming 'Month-Year' is your date column and 'Close-Last' is the value column
ibm_data['Month-Year'] = pd.to_datetime(ibm_data['Month-Year'])
ibm_data.set_index('Month-Year', inplace=True)

# Handle missing values before decomposition using interpolation
ibm_data['Close-Last'].interpolate(method='linear', inplace=True)

# Decompose the time series
result = seasonal_decompose(ibm_data['Close-Last'], model='additive', period=12)

# Create the seasonality plot
plt.figure(figsize=(12, 4))  # Adjust figure size as needed
plt.plot(result.seasonal)
plt.title('Gold Close-Price Seasonality')
plt.xlabel('Month-Year')
plt.ylabel('Seasonal Value')
plt.grid(True)  # Add gridlines for better visualization
plt.show()
```

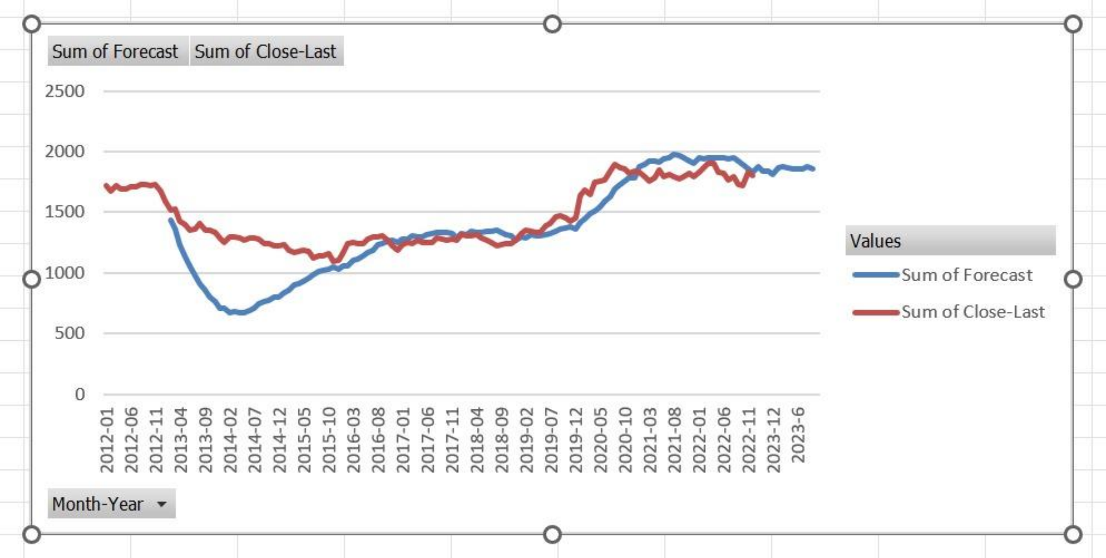
05 Results and Conclusion

Results

Thus, we found the forecasting prices for each month from 02/2013 to 12/2023.

The errors were high at first, then gradually decreased.

Results



This graph shows the difference between our forecasted prices and the actual values

Conclusion

Winter's method of Forecasting was used to predict gold prices.

05

References

References

- https://youtu.be/4_ciGzvrQl8?si=mVC2RozCjsfPRT1q

THANK YOU!