Data Analytics

The United States Oil Fund (USO): Performance Analysis and Predictive Modeling

Katerina Kovaleva

March 2023

Table of content

1. Introduction: Business Use Case

The United States Oil Fund (USO) is an exchange-traded fund (ETF) that invests in futures contracts for crude oil, heating oil, gasoline, and other petroleum-based fuels. The USO aims to track the daily price movements of West Texas Intermediate (WTI) light sweet crude oil. It is designed to provide investors with exposure to crude oil without requiring them to purchase and store barrels of oil physically.

The USO has become a popular investment vehicle for those who want to gain from oil price fluctuations, hedge against inflation, or diversify their portfolios. However, it is important to note that the USO is not designed for long-term investments and should not be considered as a substitute for owning physical oil assets or a balanced investment portfolio. Additionally, investing in the USO involves a significant degree of risk, and investors should carefully consider their investment goals and risk tolerance before investing.

Which factors can impact the performance of USO ETF? That's the question many investors ask.

There are several factors and one of them is the performance of certain stocks:

1. Energy Stocks: Energy stocks are the most obvious stocks that can impact oil prices, as they are directly related to the production and distribution of oil. When the stock prices of energy companies rise, it can indicate an increase in demand for oil, which can lead to higher oil prices.
2. Transportation Stocks: Transportation stocks, particularly those in the airline industry, can impact oil prices. When the stock prices of airline companies rise, it can indicate an increase in demand for air travel, which in turn can lead to increased demand for jet fuel and therefore, higher oil prices.
3. Technology Stocks: Technology stocks can impact oil prices indirectly. Advances in technology, such as the development of electric cars or renewable energy sources, can decrease the demand for oil, which can lead to lower oil prices.
4. Consumer Goods Stocks: Consumer goods stocks, particularly those in the retail and automotive industries, can also impact oil prices. When the stock prices of these companies rise, it can indicate an increase in consumer spending, which can lead to increased demand for gasoline and therefore, higher oil prices.
5. Financial Stocks: Financial stocks, particularly those in the banking and investment sectors, can also impact oil prices. When these stocks rise, it can indicate increased investor confidence in the economy, which can lead to increased demand for oil and higher oil prices.
   6. Automakers: Automakers are sensitive to changes in oil prices since they rely on petroleum products to power their vehicles. If the price of oil rises, the cost of producing and operating cars increases, which can lead to a decline in automaker stock prices.

Crude oil prices can also significantly impact the performance of the United States Oil

Fund stock. Any change in the price of crude oil can impact the value of the USO's portfolio of futures contracts, and subsequently, the USO's stock price. For instance, if the price of crude oil increases, the value of the USO's portfolio of futures contracts also increases, and this can lead to a rise in the USO's stock price. Conversely, if the price of crude oil decreases, the value of the USO's futures contracts decreases, leading to a decline in the USO's stock price.

The performance of the United States Oil Fund can be impacted by several other factors, including:

1. Global oil supply and demand: The USO's price is highly dependent on the supply and demand dynamics of the global oil market. Any disruption in oil supply or increase in demand can cause oil prices to rise, which can result in the USO's price increasing.
2. Geopolitical tensions: Political and economic tensions in oil-producing countries can affect the supply and demand of oil, leading to fluctuations in the USO's price.
3. OPEC decisions: The Organization of the Petroleum Exporting Countries (OPEC) plays a crucial role in global oil supply and can influence the USO's performance by adjusting their production levels.
4. Economic indicators: Economic indicators such as GDP growth, inflation, and interest rates can affect the demand for oil and, in turn, the USO's performance.
5. Currency fluctuations: Since the USO is traded in U.S. dollars, any fluctuations in the value of the U.S. dollar can impact the USO's performance.
6. Speculation: As an ETF, the USO is subject to speculation and trading activity, which can cause its price to deviate from the underlying value of the oil futures contracts it holds.

Investors in the USO should consider these factors and other risks associated with this ETF before making investment decisions.

I decided to analyse more deeply the impact of the following factors on the the USO ETF Performance:

- Major Energy Stock Prices. I selected 5 major oil and gas exploration and production companies;
- WTI Crude Oil Prices;
- Dow Jones Industrial Average and S&P 500 Indexes Prices;
- US Dollar Index fluctuations and USD to EUR ratio fluctuations.

To achieve the goal of the project, I am going to go through the different steps such as finding the proper data sources, data collection, data cleaning, building ERD, exploratory data analysis, having some insights.

2. Data and data sources

The data is taken from Yahoo! Finance https://finance.yahoo.com/ and Nasdaq https://www.nasdaq.com.

Yahoo! Finance is an excellent, free website that provides investment and financial information, including news, stock quotes, press releases, reports and more. It is a the go-to site for all news and data on markets, investments and stocks.

NASDAQ.com is a website that provides financial news, analysis, and market data for investors interested in stocks and other securities listed on the NASDAQ stock exchange. The NASDAQ stock exchange is one of the largest stock exchanges in the world, and is home to many technology companies, including Apple, Amazon, Google, and Facebook. On NASDAQ.com, investors can track the performance of individual stocks and indexes, read news articles and analysis about the financial markets, and access a variety of financial tools and resources, including stock screeners, market calendars, and educational content. The website also offers real-time quotes and charts, allowing investors to monitor the performance of their investments in real-time.

3. Data collection

Data for this study is collected from March 11th 2013 to September 20th 2022. The final dataset contains historical prices of Crude Oil, Standard and Poor's (S&P) 500 Index, Dow Jones Index, US Dollar Index and 5 Major Energy Stocks (Petroleo Brasileiro Petrobras SA, Exxon Mobil Corp, Total Energies SE, Chevron Corp, Shell PLC).

In total I downloaded 12 csv files with historical prices. These files have similar structure. The datasets contain following columns:

USO ETF : USO_Open, USO_High, USO_Low, USO_Close, USO_ Adj Close, USO_Volume;

S&P 500 Index : 'SPX_open', 'SPX_high', 'SPX_low', 'SPX_close';

Dow Jones Index : ‘'DJIA_open','DJIA_high', 'DJIA_low', 'DJIA_close';

EURO - USD Exchange Rate : 'USDEUR_Open', 'USDEUR_High', 'USDEUR_Low', 'USDEUR_Close', 'USDEUR_Adj_Close’';

Brent Crude Oil : 'BZ_Open', 'BZ_High', 'BZ_Low', 'BZ_Close', 'BZ_Volume';

WTI Crude Oil : 'CL_Open', 'CL_High', 'CL_Low', 'CL_Close', 'CL_Volume';

US Dollar Index : 'USDI_Price', 'USDI_Open', 'USDI_High','USDI_Low';

Petroleo Brasileiro Petrobras SA, : PBR_Open, PBR_High, PBR_Low, PBR_Close, PBR_ Adj Close, PBR_Volume;

Exxon Mobil Corp : XOM_Open, XOM_High, XOM_Low, XOM_Close, XOM_ Adj Close, XOM_Volume;

Total Energies SE : TTE_Open, TTE_High, TTE_Low, TTE_Close, TTE_ Adj Close, TTE_Volume;

Chevron Corp : CVX_Open, CVX_High, CVX_Low, CVX_Close, CVX_ Adj Close, CVX_Volume;

Shell PLC : SHEL_Open, SHEL_High, SHEL_Low, SHEL_Close, SHEL_ Adj Close, SHEL_Volume;

4. Data cleaning and Exploratory data analysis (EDA)

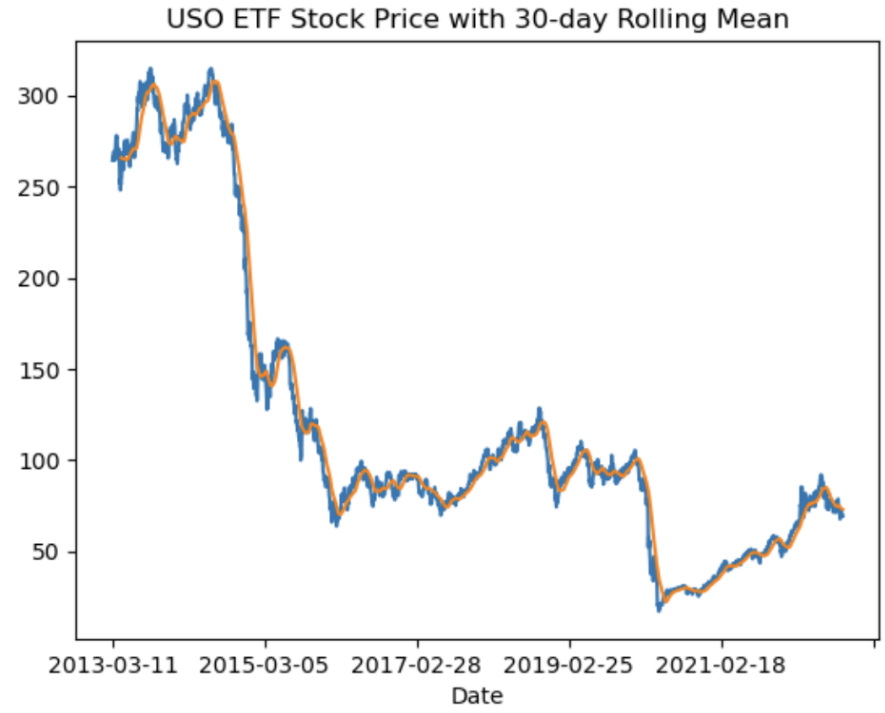
The data was cleaned using pandas library in Python. I had 29 missing values in the column 'Volume' in files 'BZ_Crude_Oil' and 'CL_Crude_Oil' so I replaced it by the average of the column. All the values in the column 'Volume' in files 'SPX' and 'DJIA were missing so I dropped the entire column for these files. Then I added the additional column 'Name' in each dataset.

In order to develop this project, it was necessary to create a database; the explanation of this process as well as the reasons for choosing a SQL database will be the subject of the seventh part of this document. We will then display the entity-relationship diagram of the database, as well as the procedures for its creation.

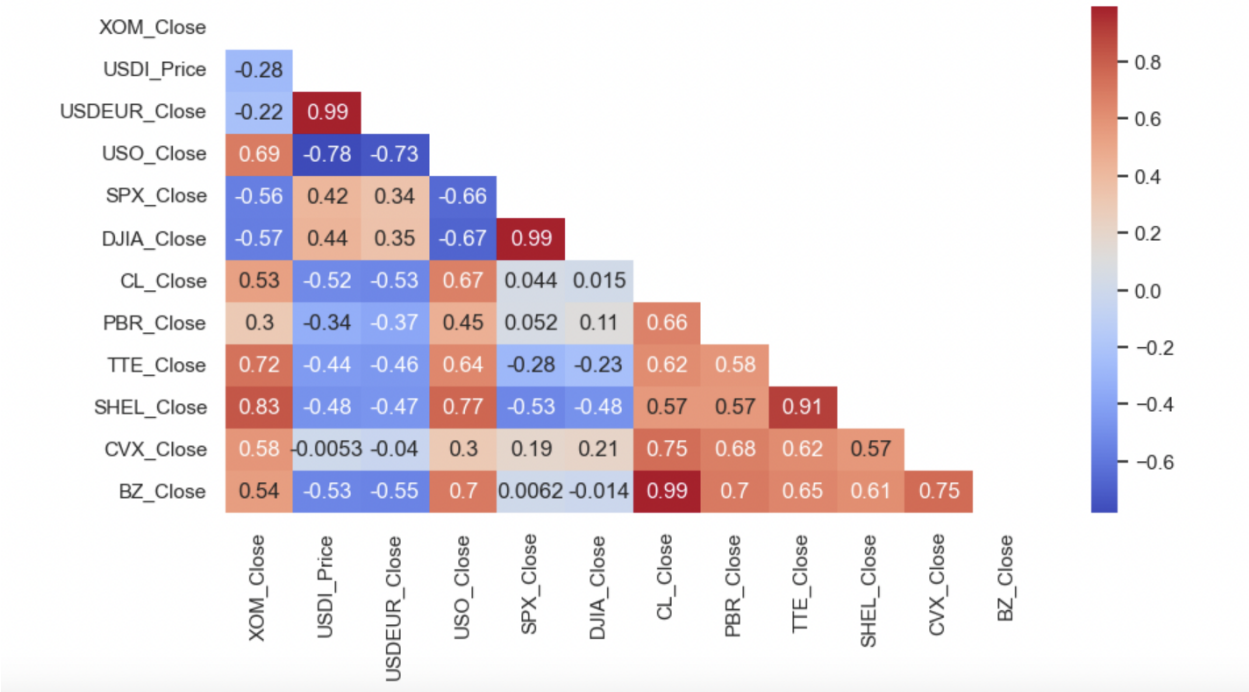
I concatenated the datasets into a single dataframe 'USO_ETF' using MySQL. The final dataset has 2401 rows in total and 63 columns in total.

I used matplotlib in pandas to visualize the historical prices of USO ETF. All prices are in USD.

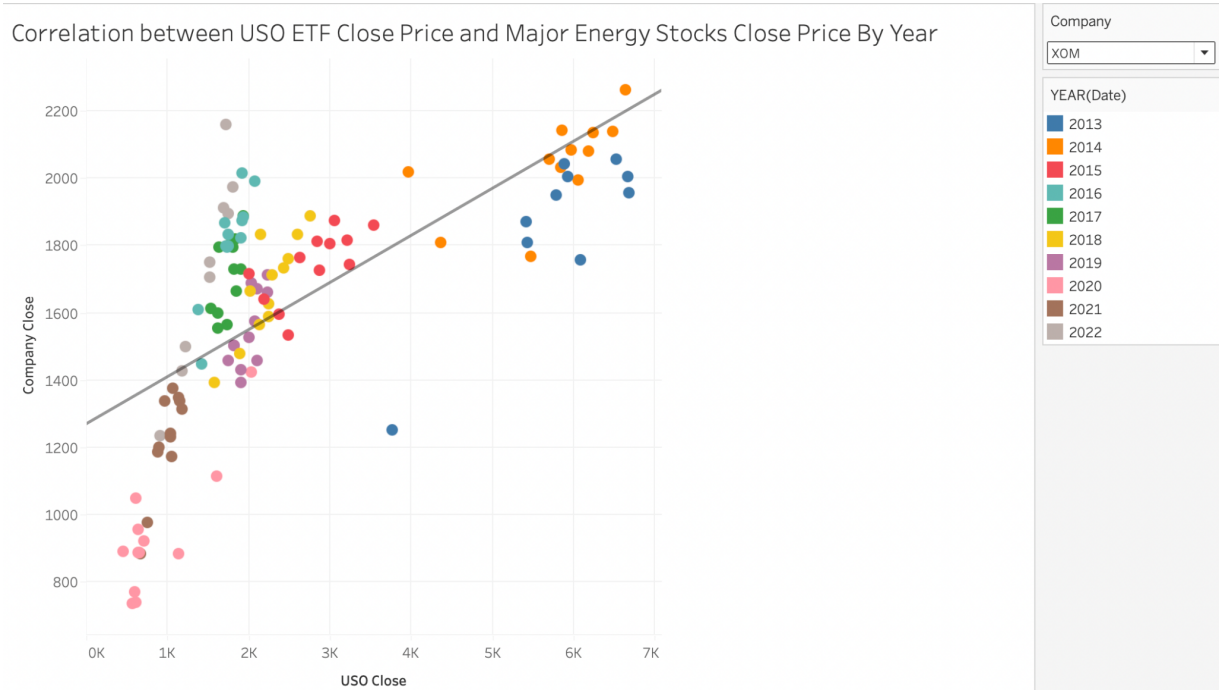
```
Autocorrelation of USO ETF Stock Price: 0.9995775214250296
```

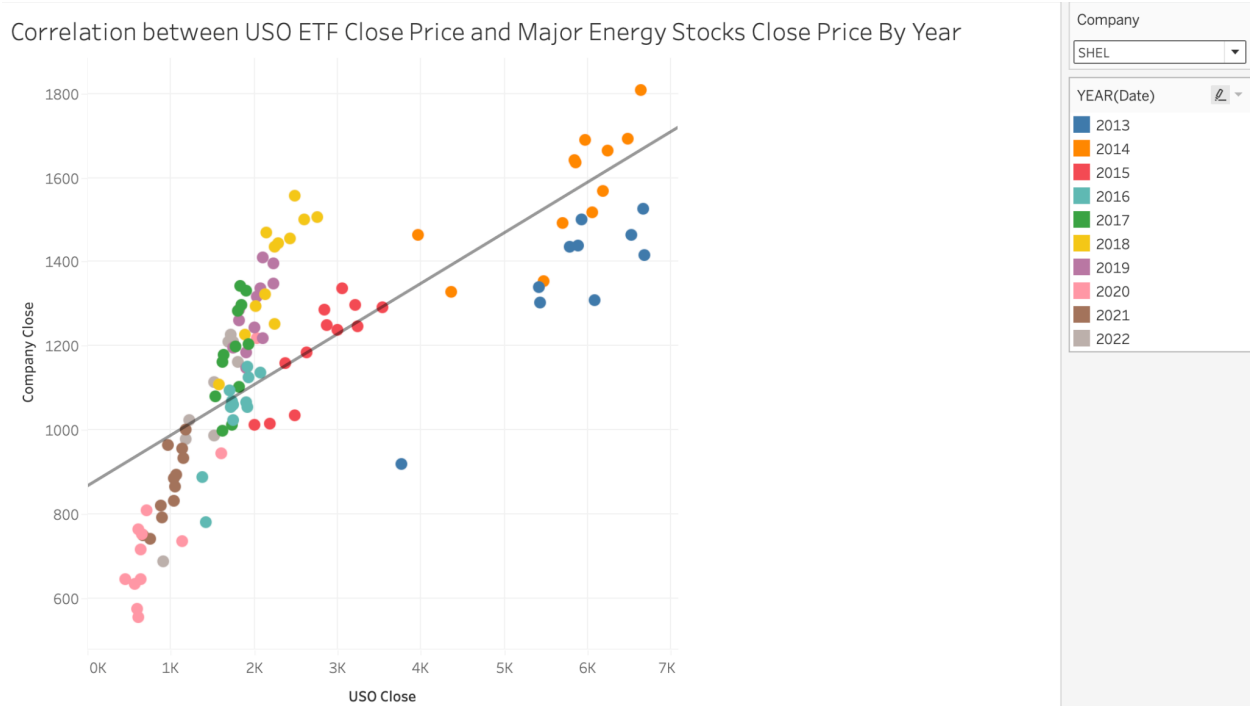


Correlation Matrix in pandas:

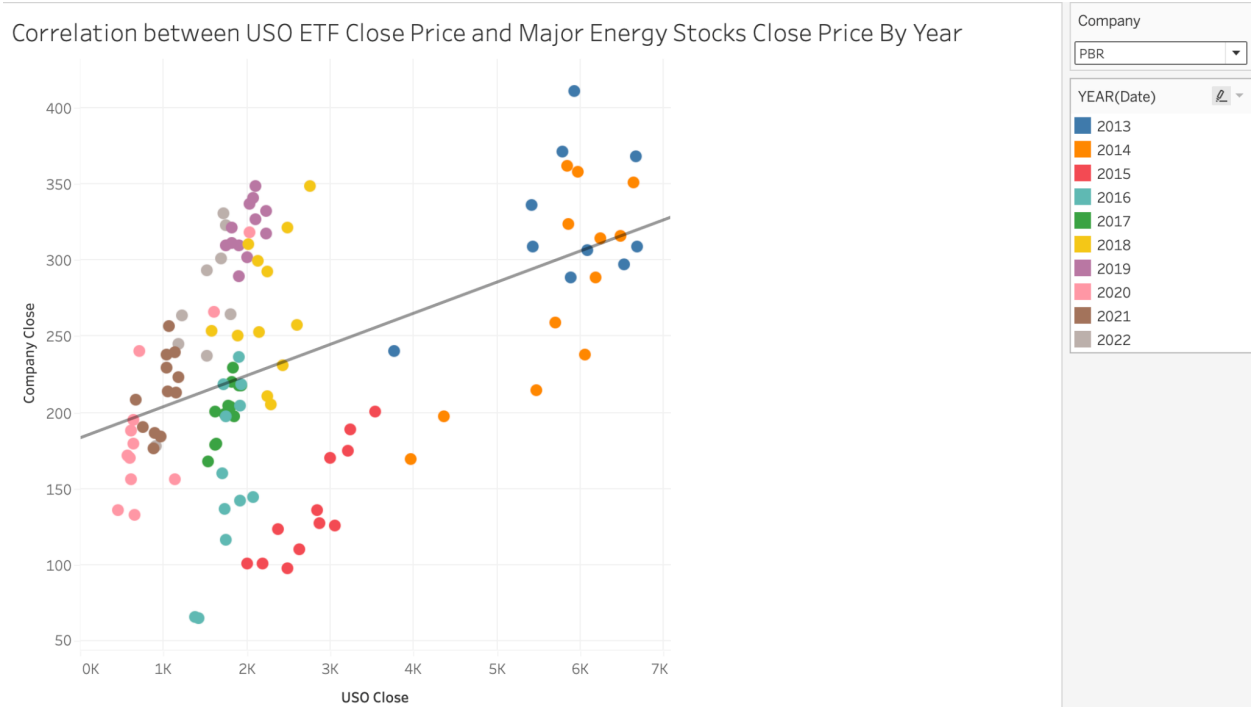


The exploration of the data showed that there is a positive correlation between the close prices of USO ETF, XOM (Exxon Mobil Corp) and SHEL (SHELL PLC) which makes sense because these companies operate in the same industry. This means that the two stocks tend to move in the same direction, either up or down, in response to market conditions or other factors that affect the industry.

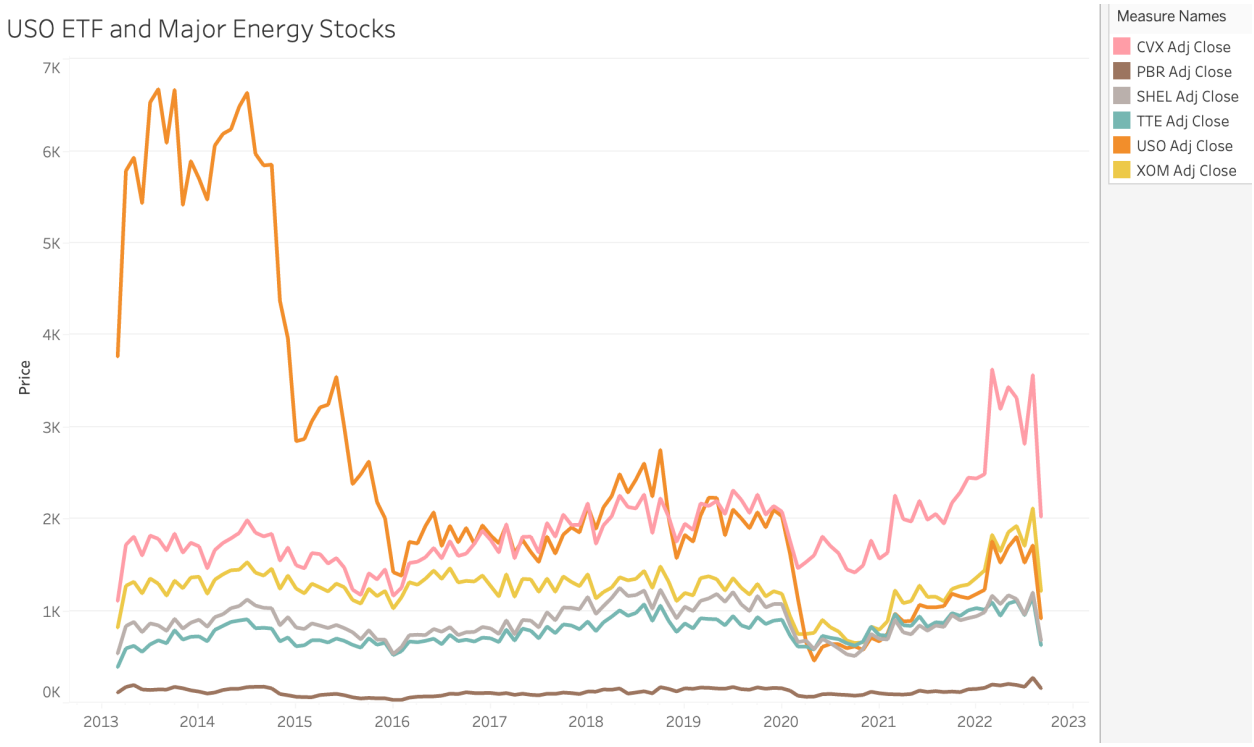




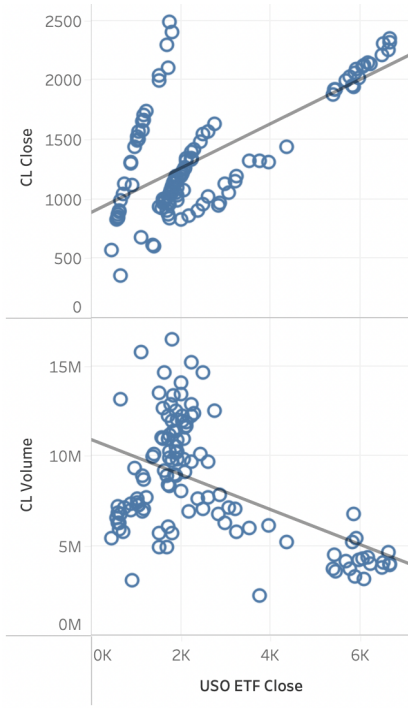
However the close prices of USO ETF, PBR (Petrobras SA), TTE (Total Energies SE) and CVX (Chevron Corp) are much less correlated:



It is interesting to see that stock prices of USO ETF, Chevron Corp, Shell PLC and Total Energies SE have similar trend during the period February 2016 - March 2020 (graph above).



We can see from the chart below that there is a positive correlation between the close prices of USO ETF and the close prices of WTI Crude Oil (CL) which is obvious because USO ETF is designed to track the performance of the spot price of West Texas Intermediate (WTI) crude oil.



The negative correlation between the USO ETF and WTI Crude Oil trading volume may be due to the relationship between trading volume and market volatility. Typically, higher trading volume is associated with higher market volatility, which means that prices may fluctuate more rapidly in response to changes in supply and demand or other market events. Conversely, lower trading volume is associated with lower market volatility, which means that prices may be more stable.

When trading volume in crude oil futures contracts increases, it can lead to greater market volatility and price fluctuations. This can have a negative impact on the performance of the USO ETF, as sudden price movements may lead to losses for investors.

On the other hand, when trading volume in crude oil futures contracts decreases, it can lead to a more stable market environment, which may be beneficial for the performance of the USO ETF.

There is also a negative correlation between the close prices of USO ETF and market indexes (S&P 500 and Dow Jones Industrial Average) performance:

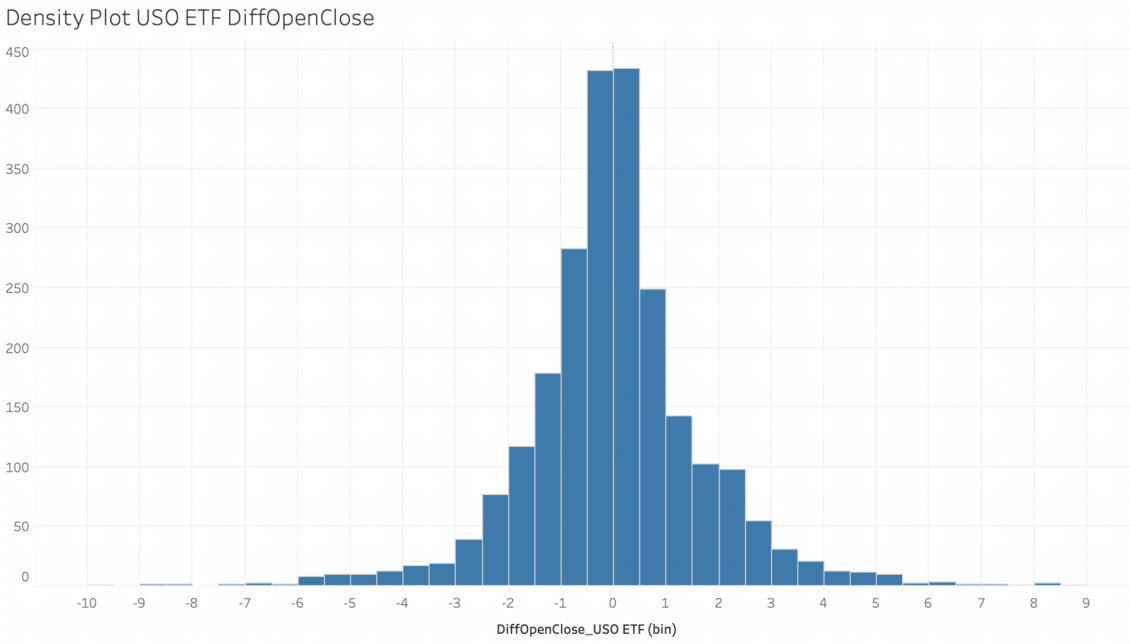
It can be explained by the nature of the oil industry and its relationship with the overall economy.

When the economy is performing well, demand for oil increases, and this leads to higher oil prices. However, higher oil prices can have a negative impact on other sectors of the economy, such as manufacturing and transportation, which can cause a decline in stock prices.

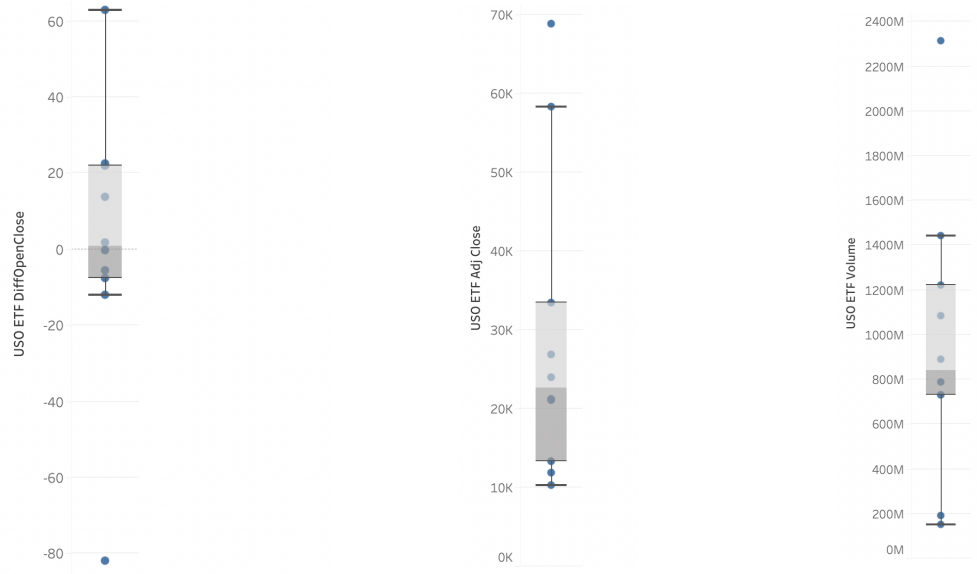
Conversely, when the economy is not performing well, demand for oil decreases, which can lead to lower oil prices. This can have a positive impact on other sectors of the economy, such as manufacturing and transportation, which can cause an increase in stock prices.

As a result, when the market indexes S&P 500 and Dow Jones are performing well, the USO ETF may be underperforming due to the negative impact of higher oil prices on other sectors of the economy. Conversely, when S&P 500 and Dow Jones are not performing well, the USO ETF may be outperforming due to the positive impact of lower oil prices on other sectors of the economy.

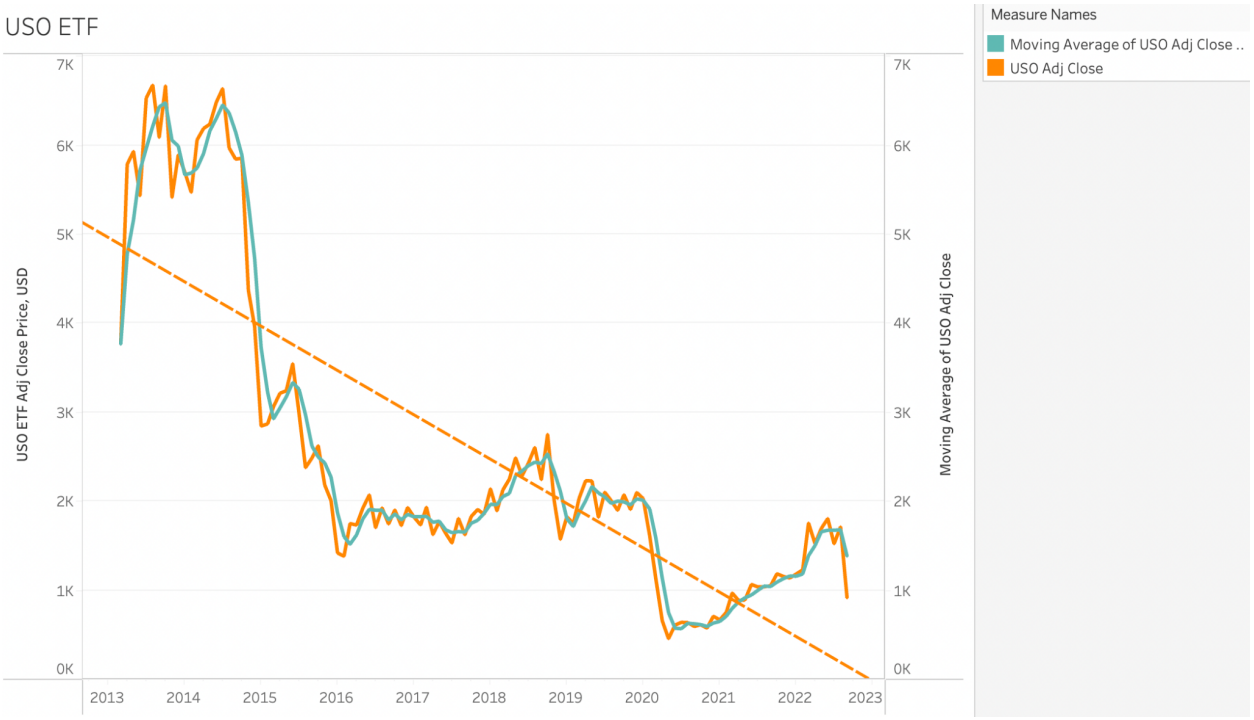
We can see from the Density Plot below that daily returns (difference between open and close price) of USO ETF are normally distributed:

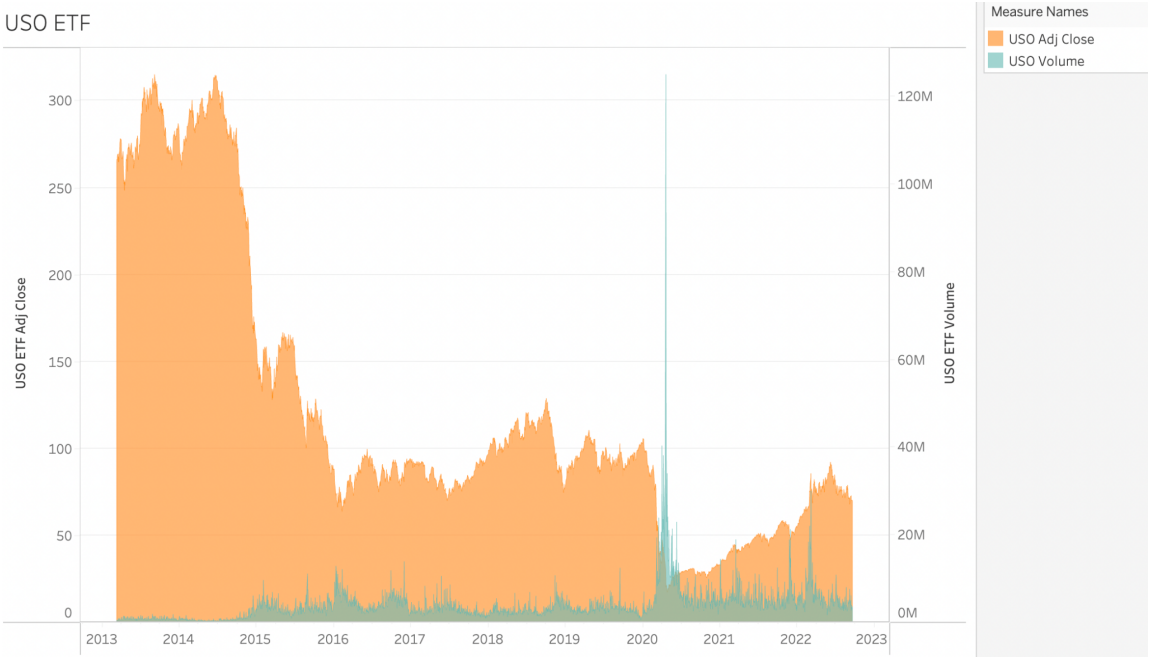


I created the Box Plots to visualize the daily returns for USO ETF, its adjusted close price distribution and trading volume distribution:



The chart below shows the historical prices and trading volume of USO ETF from 2013 to 2022:





We can see the downtrend of USO ETF prices since mid-2014.
We can also note that in April 2020 USO ETF trading volume increased sharply. If a stock with a high trading volume is rising, it usually means there is a strong buying pressure as investors demand pushes the stock to higher and higher prices. However in the case of USO ETF the stock with a high trading volume is falling. It suggests that there is a lot of selling pressure which could be the effect of Covid-19.

The graphs above were built in Tableau.

5. Databases type comparison

Choice of database type. Choosing the database type I took into account the fact that the data was tabular. Following the definition of SQL database, it is a "collection of highly structured tables, wherein each row reflects a data entity, and every column defines a specific information field. Relational databases are built using the structured query language (SQL) to create, store, update, and retrieve data". So I decided to use MySQL database because it was the most convenient for tabular data. It also allows you to make queries quickly and to join different information from various tables.

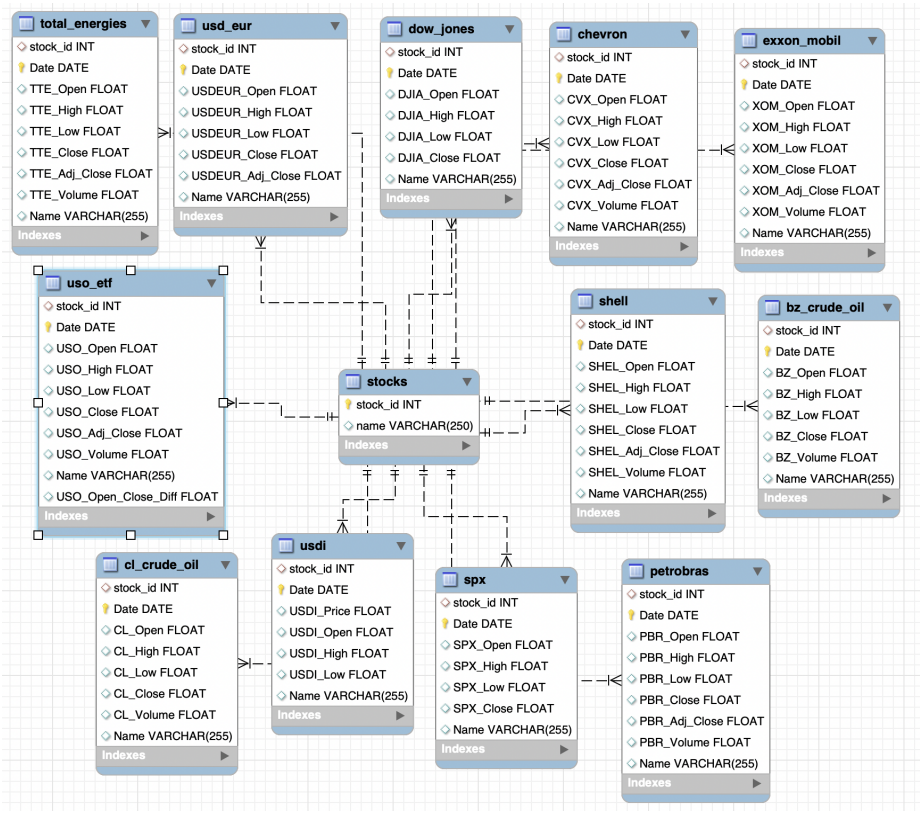
MySQL was used as the database management system for its construction and implementation.

Main reasons why I chose SQL and no NoSQL:

- SQL databases are relational, NoSQL databases are non-relational;
- SQL databases use structured query language and have a predefined schema. NoSQL databases have dynamic schemas for unstructured data;
- SQL databases are vertically scalable, while NoSQL databases are horizontally scalable;
- SQL databases are table-based, while NoSQL databases are documents, key-value, graph, or wide-column stores;
- SQL databases are better for multi-row transactions, while NoSQL is better for unstructured data (ex. documents, JSON).

6. Entities. ERD

While structuring the database I took into account that there was one type of relationships: "one to one" (Every company has only one stock_id; in table “stocks” all the stock_id are unique). This relationship led to the following database structure:



7. Creation of the database and data importation

Database creation. The creation of the database was done using MySQL script. My database is called “USO_ETF”. The first step was to create tables, then I filled them using 'Table Data Import Wizard' with data resulting from the data cleaning. For the creation of the database the primary keys as well as the foreign keys were established.

```
CREATE DATABASE USO_ETF;

USE USO_ETF;

CREATE TABLE

   stocks(

      stock_id INT AUTO_INCREMENT,

      name VARCHAR(250) UNIQUE,

      PRIMARY KEY (stock_id)

   );
```

```
INSERT INTO stocks (name)

VALUES ('United States Oil ETF'), ('Brent Crude Oil'), ('WTI Crude Oil'), ('Chevron Corp'), ('Dow Jones Industrial Average'), ('Petroleo Brasileiro SA Petrobras'), ('Shell PLC'),('SPX'),('TotalEnergies SE'),('USD to EUR exchange rate'),('US dollar index'), ('Exxon Mobil Corp');

##1

CREATE TABLE

    IF NOT EXISTS exxon_mobil(

        stock_id INT,

        Date DATE NOT NULL,

        XOM_Open FLOAT,

        XOM_High FLOAT,

        XOM_Low FLOAT,

        XOM_Close FLOAT,

        XOM_Adj_Close FLOAT,

        XOM_Volume FLOAT,

        Name VARCHAR(255),

        PRIMARY KEY (Date),

        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)

    );


UPDATE exxon_mobil e, stocks s

SET e.stock_id = s.stock_id

WHERE e.Name = s.name;


##2

CREATE TABLE

    IF NOT EXISTS bz_crude_oil(

        stock_id INT,

        Date DATE NOT NULL,

        BZ_Open FLOAT,

        BZ_High FLOAT,
```

```
        BZ_Low FLOAT,
        BZ_Close FLOAT,
        BZ_Volume FLOAT,
        Name VARCHAR(255),
        PRIMARY KEY (Date),
        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)
    );

UPDATE bz_crude_oil bz, stocks s
SET bz.stock_id = s.stock_id
WHERE bz.Name = s.name;

##3
CREATE TABLE
    IF NOT EXISTS cl_crude_oil(
        stock_id INT,
        Date DATE NOT NULL,
        CL_Open FLOAT,
        CL_High FLOAT,
        CL_Low FLOAT,
        CL_Close FLOAT,
        CL_Volume FLOAT,
        Name VARCHAR(255),
        PRIMARY KEY (Date),
        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)
    );

UPDATE cl_crude_oil cl, stocks s
SET cl.stock_id = s.stock_id
WHERE cl.Name = s.name;
```

```
select * from cl_crude_oil;


##4
CREATE TABLE
    IF NOT EXISTS chevron(
        stock_id INT,
        Date DATE NOT NULL,
        CVX_Open FLOAT,
        CVX_High FLOAT,
        CVX_Low FLOAT,
        CVX_Close FLOAT,
        CVX_Adj_Close FLOAT,
        CVX_Volume FLOAT,
        Name VARCHAR(255),
        PRIMARY KEY (Date),
        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)
    );

UPDATE chevron c, stocks s
SET c.stock_id = s.stock_id
WHERE c.Name = s.name;


##5
CREATE TABLE
    IF NOT EXISTS dow_jones(
        stock_id INT,
        Date DATE NOT NULL,
        DJIA_Open FLOAT,
        DJIA_High FLOAT,
```

```
        DJIA_Low FLOAT,
        DJIA_Close FLOAT,
        Name VARCHAR(255),
        PRIMARY KEY (Date),
        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)
    );

UPDATE dow_jones dj, stocks s
SET dj.stock_id = s.stock_id
WHERE dj.Name = s.name;

select * from dow_jones;

##6
CREATE TABLE
    IF NOT EXISTS petrobras(
        stock_id INT,
        Date DATE NOT NULL,
        PBR_Open FLOAT,
        PBR_High FLOAT,
        PBR_Low FLOAT,
        PBR_Close FLOAT,
        PBR_Adj_Close FLOAT,
        PBR_Volume FLOAT,
        Name VARCHAR(255),
        PRIMARY KEY (Date),
        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)
    );

UPDATE petrobras p, stocks s
```

```
SET p.stock_id = s.stock_id

WHERE p.Name = s.name;


##7

CREATE TABLE

    IF NOT EXISTS shell(

        stock_id INT,

        Date DATE NOT NULL,

        SHEL_Open FLOAT,

        SHEL_High FLOAT,

        SHEL_Low FLOAT,

        SHEL_Close FLOAT,

        SHEL_Adj_Close FLOAT,

        SHEL_Volume FLOAT,

        Name VARCHAR(255),

        PRIMARY KEY (Date),

        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)

    );


UPDATE shell sh, stocks s

SET sh.stock_id = s.stock_id

WHERE sh.Name = s.name;


##8

CREATE TABLE

    IF NOT EXISTS spx(

        stock_id INT,

        Date DATE NOT NULL,

        SPX_Open FLOAT,

        SPX_High FLOAT,
```

```
        SPX_Low FLOAT,
        SPX_Close FLOAT,
        Name VARCHAR(255),
        PRIMARY KEY (Date),
        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)
    );

UPDATE spx sp, stocks s
SET sp.stock_id = s.stock_id
WHERE sp.Name = s.name;

##9
CREATE TABLE
    IF NOT EXISTS total_energies(
        stock_id INT,
        Date DATE NOT NULL,
        TTE_Open FLOAT,
        TTE_High FLOAT,
        TTE_Low FLOAT,
        TTE_Close FLOAT,
        TTE_Adj_Close FLOAT,
        TTE_Volume FLOAT,
        Name VARCHAR(255),
        PRIMARY KEY (Date),
        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)
    );

UPDATE total_energies t, stocks s
SET t.stock_id = s.stock_id
WHERE t.Name = s.name;
```

```
##10
CREATE TABLE
    IF NOT EXISTS usd_eur(
        stock_id INT,
        Date DATE NOT NULL,
        USDEUR_Open FLOAT,
        USDEUR_High FLOAT,
        USDEUR_Low FLOAT,
        USDEUR_Close FLOAT,
        USDEUR_Adj_Close FLOAT,
        Name VARCHAR(255),
        PRIMARY KEY (Date),
        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)
    );

UPDATE usd_eur, stocks s
SET usd_eur.stock_id = s.stock_id
WHERE usd_eur.Name = s.name;

##11
CREATE TABLE
    IF NOT EXISTS usdi(
        stock_id INT,
        Date DATE NOT NULL,
        USDI_Price FLOAT,
        USDI_Open FLOAT,
        USDI_High FLOAT,
        USDI_Low FLOAT,
        Name VARCHAR(255),
```

```
        PRIMARY KEY (Date),
        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)
    );
UPDATE usdi, stocks s
SET usdi.stock_id = s.stock_id
WHERE usdi.Name = s.name;

##12
CREATE TABLE
    IF NOT EXISTS uso_etf(
        stock_id INT,
        Date DATE NOT NULL,
        USO_Open FLOAT,
        USO_High FLOAT,
        USO_Low FLOAT,
        USO_Close FLOAT,
        USO_Adj_Close FLOAT,
        USO_Volume FLOAT,
        Name VARCHAR(255),
        PRIMARY KEY (Date),
        FOREIGN KEY(stock_id) REFERENCES stocks(stock_id)
    );

UPDATE uso_etf u, stocks s
SET u.stock_id = s.stock_id
WHERE u.Name = s.name;
```

You can see my main queries below:

1. Add a new column USO_Open_Close_Diff (shows the difference between Open and Close Price of USO ETF)

```
ALTER TABLE uso_etf ADD COLUMN USO_Open_Close_Diff FLOAT;

UPDATE uso_etf SET USO_Open_Close_Diff = USO_Open - USO_Close;
```

2. Select the maximum and minimum difference between Open and Close Price

```
SELECT MIN(USO_Open_Close_Diff) AS Min_Diff FROM uso_etf;

SELECT MAX(USO_Open_Close_Diff) AS Max_Diff FROM uso_etf;
```

3. Find the average volume for each of the four securities:

```
SELECT 'XOM' AS Security, AVG(XOM_Volume) AS Avg_Volume

FROM exxon_mobil

UNION ALL

SELECT 'TTE' AS Security, AVG(TTE_Volume) AS Avg_Volume

FROM total_energies

UNION ALL

SELECT 'PBS' AS Security, AVG(PBR_Volume) AS Avg_Volume

FROM petrobras
```

```
UNION ALL

SELECT 'CVX' AS Security, AVG(CVX_Volume) AS Avg_Volume

FROM chevron

UNION ALL

SELECT 'SHEL' AS Security, AVG(SHEL_Volume) AS Avg_Volume

FROM shell;
```

4. Find the dates when USO ETF had the highest closing price:

```
SELECT

    Date,

    USO_Close

FROM uso_etf

WHERE USO_Close = (SELECT MAX(USO_Close) FROM uso_etf);
```

5. Find the average adjusted closed price for USO for each year:

```
SELECT

    YEAR(Date) AS Year,

    AVG(USO_Adj_Close) AS Avg_USO_Adj_Close

FROM uso_etf
```

```
GROUP BY YEAR(Date);
```

5. I used the following script to concatenate the tables:

```
SELECT

        exxon_mobil.Date, exxon_mobil.XOM_Open, exxon_mobil.XOM_Close, exxon_mobil.XOM_High, exxon_mobil.XOM_Low,
exxon_mobil.XOM_Adj_Close, exxon_mobil.XOM_Volume,

    bz_crude_oil.BZ_Open, bz_crude_oil.BZ_Close, bz_crude_oil.BZ_High, bz_crude_oil.BZ_Low, bz_crude_oil.BZ_Volume,

    cl_crude_oil.CL_Open, cl_crude_oil.CL_Close, cl_crude_oil.CL_High, cl_crude_oil.CL_Low, cl_crude_oil.CL_Volume,

     chevron.CVX_Open, chevron.CVX_Close, chevron.CVX_High, chevron.CVX_Low, chevron.CVX_Adj_Close, chevron.CVX_Volume,

    dow_jones.DJIA_Open, dow_jones.DJIA_Close, dow_jones.DJIA_High, dow_jones.DJIA_Low,

    petrobras.PBR_Open, petrobras.PBR_Close, petrobras.PBR_High, petrobras.PBR_Low, petrobras.PBR_Adj_Close,
petrobras.PBR_Volume,

    shell.SHEL_Open, shell.SHEL_Close, shell.SHEL_High, shell.SHEL_Low, shell.SHEL_Adj_Close, shell.SHEL_Volume,

    spx.SPX_Open, spx.SPX_Close, spx.SPX_High, spx.SPX_Low,

    total_energies.TTE_Open, total_energies.TTE_Close, total_energies.TTE_High, total_energies.TTE_Low,
total_energies.TTE_Adj_Close, total_energies.TTE_Volume,

    usd_eur.USDEUR_Open, usd_eur.USDEUR_Close, usd_eur.USDEUR_High, usd_eur.USDEUR_Low, usd_eur.USDEUR_Adj_Close,

    usdi.USDI_Price, usdi.USDI_Open, usdi.USDI_High, usdi.USDI_Low,

    uso_etf.USO_Open, uso_etf.USO_Close, uso_etf.USO_High, uso_etf.USO_Low, uso_etf.USO_Adj_Close, uso_etf.USO_Volume

    FROM exxon_mobil
```

```
JOIN bz_crude_oil ON exxon_mobil.Date = bz_crude_oil.Date

JOIN cl_crude_oil ON exxon_mobil.Date = cl_crude_oil.Date

JOIN chevron ON exxon_mobil.Date = chevron.Date

JOIN dow_jones ON exxon_mobil.Date = dow_jones.Date

JOIN petrobras ON exxon_mobil.Date = petrobras.Date

JOIN shell ON exxon_mobil.Date = shell.Date

JOIN spx ON exxon_mobil.Date = spx.Date

JOIN total_energies ON exxon_mobil.Date = total_energies.Date

JOIN usd_eur ON exxon_mobil.Date = usd_eur.Date

JOIN usdi ON exxon_mobil.Date = usdi.Date

JOIN uso_etf ON exxon_mobil.Date = uso_etf.Date;
```

8. Conclusion

Analysing the impact of different factors on the performance on the The United States Oil Fund performance I came to the conclusion that there is a positive correlation between the prices of USO ETF, XOM (Exxon Mobil Corp) and SHEL (SHELL PLC) which makes sense because these companies operate in the same industry. However the prices of USO ETF, PBR (Petrobras SA), TTE (Total Energies SE) and CVX (Chevron Corp) are much less correlated. The performance of USO ETF also depends a lot on Crude Oil prices.

I saw that the prices of USO ETF and three major energy stocks prices had similar trend in 2016 - 2020. I found out that there is a negative correlation between the prices of USO ETF and market indexes (S&P 500 and Dow Jones Industrial Average) performance. It can be explained by the nature of the oil industry and its relationship with the overall economy. When the economy is performing well, demand for oil increases, and this leads to higher oil prices. However, higher oil prices can have a

negative impact on other sectors of the economy, such as manufacturing and transportation, which can cause a decline in stock prices. There is also a negative correlation between the prices USO ETF and WTI Crude Oil trading volume may be due to the relationship between trading volume and market volatility. Typically, higher trading volume is associated with higher market volatility, which means that prices may fluctuate more rapidly in response to changes in supply and demand or other market events.

Apart from all of the above, performance of the United States Oil Fund can be impacted by several other factors, including global oil supply and demand, geopolitical tensions, OPEC's decisions, etc. Investors in the USO should consider these factors and other risks associated with this ETF before making investment decisions.

The next step of this project is to implement Machine Learning Models to predict the performance of USO ETF. The target variable will be USO ETF adjusted close price.

9. Links

https://github.com/katerinakovaleva201/Final_Project_USO_ETF_Performance_Analysis.git