PYTHON-DRIVEN ANALYSIS OF EXCHANGE-TRADED FUNDS

By:
Uzma Naeem

TABLE OF CONTENTS

INTRODUCTION

- **Dataset Name:** Exchange Trade Funds (ETFs) and Mutual Funds Composition & Yield Metrics.

- **Source**: Kaggle - Link

- **Number of Columns and Rows:** Columns: 129, Rows: 2,264

- **Description of the data set:**
  This dataset allows us to research, analyze, and compare ETFs using a wide range of data and metrics. These include fund size, expense ratio, asset allocation, sector allocation, geographic distribution, dividend yield, auditing company, legal structure, distribution strategies, and past performance among many others.

- **Why did we pick this data set?**
  Sommer, Jeff. "The Risks Hidden in Public Pension Funds." *The New York Times* indicates how the government invests public funds into private equities and it becomes important to understand the volatility and risk factors involved in these decisions and assessing this data set gives us better insights about the ETF market (private equities).

  In addition, Exchange-traded funds (ETFs) are one of the best choices for a long term investment because compared with the stock market or funds, ETFs offer trading flexibility, diversified portfolios for risk management, cost-effectiveness, and tax advantages.

KEY ANALYTICAL QUESTIONS

Driving Question: What is the best ETF portfolio based on an investor's investment strategy (aggressive, moderate, conservative)?

Questions Guiding Our Analysis:

- How to identify top-performing ETFs?
- How does the domicile country of the ETF affect the investor?
- What is the relationship between volatility and returns of an ETF?
- What are Hedged and Unhedged currencies?

Identifying top-performing ETFs:
The performance of ETFs is mainly deciphered based on the year's returns. Aligning this with the fees incurred and the cost of the ETF, the currency risk factor, the compliance factor of an ETF, and the volume of the ETF provides a picture of the success of the ETF. The previous 5 factors mentioned are crucial in the methodology to identify the top-performing ETFs. In

addition, the best ETFs are categorized based on an investor's investment strategy (aggressive, moderate, conservative) using k-means as a clustering technique.

Domicile countries of ETF:
The domicile countries of ETFs are important in understanding the currency of exposure for the ETF. The currency becomes an important factor for an investor as ETFs can be bought and traded internationally, the risks taken can vary based on the currency it is bought in and the country of origin.

Volatility Vs Returns:
The volatility of an ETF describes the changes it undergoes in value across time periods. The more volatile a fund is; the higher the returns that are possible. But higher returns come with higher risks and understanding them becomes essential while creating a portfolio. So an investor's strategy plays an important role in identifying the volatility rates that might work for them.

Hedged and Unhedged Currencies:
Hedged currencies are the ones that retain their value at the time of investment and unhedged currencies are the ones that vary in value depending on the current market. ETFs can be invested in using either. So, it becomes important for the investor to understand what currency the ETF is bought in to understand if their investment will hold the same value throughout or not. Based on their strategy this can be considered to be good or bad.

METHODOLOGY:

1. **Returns:** In this study, we use the study from Investopedia to understand that portfolios with the following categories of ETF based on their Year-to-Date Return will be the best combination of ETFs

Group	YTDR
A	<4%
B	4%-6%
C	6%-8%
D	8%-14%
E	>14%

2. **Fee:** The second method for categorization is based on the fee incurred in each ETF which is indicated in the 'ter' column. ETFs with fees less than 0.5% of the quote price are considered 'Good' (1 during coding) and if not are considered 'bad' (0 during coding).

3. **Risk Factor:** The third method of categorization is based on the risk factors for a currency. This is studied using the 'hedged' or 'unhedged' description in the 'Currency Risk' column. Hedged currencies are considered risk-free (1 during coding) and unhedged currencies are considered risky (0 during coding).

4. **Compliance:** The fourth method of categorization is based on the compliance factor of an ETF. They are categorized based on the UCITs compliance factor which is indicated in the 'UCIT_Compliance' column. Compliance is 1 during coding and non compliance is 0 during coding.

5. **Volume:** The final method is to categorize based on the volume of the ETF. ETFs with a volume over 10 million are considered reliable (1 during coding) and the ones below that are not (0 during coding).

DATA CLEANING – PREPROCESSING

Data cleaning is a critical phase to ensure accuracy and reliability for meaningful analysis. The cleanliness and accuracy of the dataset were prioritized to lay a strong foundation for subsequent analyses.

Original dataset:

	isin	wkn	name	fundProvider	legalStructure	quote	quote52Low	quote52High	ytdReturnCUR	ter	...	exposureCountry_Argent
0	IE00B0M62Y33	A0HGWF	iShares AEX UCITS ETF	iShares	ETF	73.98	62.81	76.92	0.0541	0.0030	...	
1	IE00BMTX2B82	A2P9XA	iShares AEX UCITS ETF EUR (Acc)	iShares	ETF	6.68	5.80	7.13	0.0503	0.0030	...	
2	NL0009272749	A1JN2C	VanEck AEX UCITS ETF	VanEck	ETF	72.34	63.16	77.27	0.0510	0.0030	...	
3	IE000RN036E0	A3DGK2	First Trust Alerian Disruptive Technology Real...	First Trust	ETF	17.44	17.07	23.80	0.0029	0.0060	...	
4	IE00BKPTXQ89	A2P4PH	HANetf Alerian Midstream Energy Dividend UCITS...	HANetf	ETF	10.7	10.39	13.12	-0.0599	0.0040	...	
...	...	...	...	...	...	...	...	...	...	...	...	
2259	IE00BD6RZW23	A2AS6C	WisdomTree US Equity Income UCITS ETF EUR Hedg...	WisdomTree	ETF	19.69	18.67	22.28	-0.0777	0.0035	...	
2260	IE00BZ56RG20	A2AGPV	WisdomTree US Quality Dividend Growth UCITS ET...	WisdomTree	ETF	31.98	29.23	34.89	0.0019	0.0033	...	
2261	IE00B3Y8D011	A1C1G8	Xtrackers Portfolio Income UCITS ETF 1D	Xtrackers	ETF	11.79	11.41	12.92	0.0164	0.0065	...	
2262	CH0496484640	A22FMC	21Shares Bitcoin Suisse Index ETP	21Shares	ETN	11.88	7.35	20.93	0.6027	0.0250	...	
2263	CH0445689208	A2TT3D	21Shares Crypto Basket Index ETP	21Shares	ETN	7.03	4.39	15.08	0.5900	0.0250	...	

2264 rows × 129 columns

Enhanced Dataset Quality

1. **Columns Selection:**

   In the initial phase of data cleaning, we strategically curated the dataset by selecting key columns essential for our comprehensive ETF analysis. The chosen columns, including **'inceptionDate,' 'quoteDate,' 'ter'** (Total Expense Ratio), **'quote'** (Quote Price), **'fundSizeMillions,' 'isin'** (International Securities Identification Number),

'ytdReturnCUR' (Year-to-Date Return in Currency), **'yearVolatilityCUR'** (Year Volatility in Currency), **'currencyRisk,' 'UCITSCompliance,' 'domicileCountry,' 'fundCurrency,'** and **'name,'** were meticulously chosen to capture crucial aspects of ETF characteristics.

These columns collectively provide insights into the fund's operational timeline, cost structure, market performance, risk metrics, compliance with regulations, geographic exposure, and other fundamental attributes. This strategic column selection aligns with our project's objectives, enabling a thorough and nuanced analysis of ETFs, and fostering a holistic understanding of the market landscape.

In [205]:
```
columns_to_keep = [
    'inceptionDate', 'quoteDate', 'ter', 'quote',
    'fundSizeMillions', 'isin', 'ytdReturnCUR', 'yearVolatilityCUR',
    'currencyRisk','UCITSCompliance','domicileCountry','fundCurrency','name']
```

2. Data Conversion:

In the second phase of our comprehensive data cleaning, a detailed approach was applied to fortify the accuracy and uniformity of the dataset tailored for the ETF analysis project. The provided code outlines key transformations, including the conversion of **'inceptionDate'** and **'quoteDate'** to **datetime** objects for standardized temporal analyses. Additionally, crucial financial metrics like **'ter,' 'quote,'** and **'fundSizeMillions'** underwent **numeric conversion**, ensuring precise mathematical computations. These conversions collectively enhance the dataset's integrity, laying a robust foundation for subsequent analyses. This detailed and systematic approach aligns with our commitment to conducting a thorough, reliable exploration of ETF market dynamics.

```
#Data conversion

etfFiles['inceptionDate'] = pd.to_datetime(etfFiles['inceptionDate'], errors='coerce')
etfFiles['quoteDate'] = pd.to_datetime(etfFiles['quoteDate'])
etfFiles['ter'] = pd.to_numeric(etfFiles['ter'], errors='coerce')
etfFiles['quote']=pd.to_numeric(etfFiles['quote'],errors='coerce')
etfFiles['fundSizeMillions']=pd.to_numeric(etfFiles['fundSizeMillions'],errors='coerce')
```

3. Data Filtering:

In the third step of our meticulous data cleaning process, a strategic decision was taken to filter the ETF dataset based on the **'quoteDate,'** aligning with the core objectives of our ETF analysis project. This approach ensures temporal relevance by focusing on March 2023, enhancing the precision of our analyses, and facilitating a concentrated investigation. By isolating data within this defined period, we have established a foundation for a more accurate, relevant, and focused exploration of ETF market trends, adhering to the high standards set in our comprehensive analysis project.

In [6]:
```
# Filter quoteDate of the ETFs - March 2023

filtered_data = etfFiles[(etfFiles['quoteDate'] >= '2023-03-01') & (etfFiles['quoteDate'] < '2023-04-01')]
filtered_data

#filtered_data['quoteDate'].unique()
```

4. **Indexing:**

In the fourth step of our data cleaning process, the decision to designate the **'isin'** column values as the index was strategic and rooted in enhancing accessibility and ensuring consistency. This indexing approach improves the efficiency of data retrieval, providing a streamlined method for precise ETF identification. By maintaining consistency in data handling, we have established a solid foundation for a coherent and precise examination of the ETF market, aligning with the robust standards set in our comprehensive analysis project.

```
In [7]: # index values
        filtered_data.index = filtered_data['isin'].values
        #filtered_data.set_index('isin',inplace=True)
```

5. **Handling Missing Values:**

In the fifth step of the data cleaning process, missing values in vital metrics such as **'ytdReturnCUR'** and **'yearVolatilityCUR'** were addressed. This meticulous approach aimed to preserve data integrity, focusing on key indicators essential for the ETF analysis. By systematically removing rows with incomplete data, the goal was to ensure statistical robustness in subsequent analyses, enhancing the reliability and credibility of the findings. This strategic decision underscores the commitment to maintaining high data quality standards throughout the project.

```
In [8]: # Delete values where the ytdReturnCUR  or yearVolatilityCUR is null
        filtered_data.dropna(subset=['ytdReturnCUR'],inplace=True)
        filtered_data.dropna(subset=['yearVolatilityCUR'],inplace=True)
        filtered_data
```

DATA TRANSFORMATION

To transform the data as described in the methodology above the following steps are undertaken.

1. Group the ETFs into 4 categories - A, B, C, and D with 4,6,8 & 14% yearly returns respectively.

```
oups of ETF according to the ytdReturnCUR

tered_data['Group']='A'
tered_data.loc[(filtered_data['ytdReturnCUR']>=0.04)&(filtered_data['ytdReturnCUR']<0.06),'Group']='B'
tered_data.loc[(filtered_data['ytdReturnCUR']>=0.06)&(filtered_data['ytdReturnCUR']<0.08),'Group']='C'
tered_data.loc[(filtered_data['ytdReturnCUR']>=0.08)&(filtered_data['ytdReturnCUR']<0.14),'Group']='D'
tered_data.loc[(filtered_data['ytdReturnCUR']>0.14),'Group']='E'
```

2. Fee Comparison: 1 if the fee is less than 0.5% of the quoted price; else 0.

```
In [11]: # What is the equivalency of ter on the quote
         filtered_data['Fee_comparison']=(filtered_data['ter']<0.0005*filtered_data['quote']).astype(int)
```

3. Currency Risk: 1 if the currency is hedged; else 0 (un-hedged)

```
In [12]: # column "Currency_Risk". If the colum " currencyRisk" is hedge is good=1 else 0
filtered_data['Currency_Risk'] = filtered_data['currencyRisk'].apply(lambda x: 1 if x == 'Currency hedged' else 0)

#filtered_data['Currency_Risk'].unique()
#filtered_data['currencyRisk'].unique()
```

4. Compliance Evaluation: 1 if ETF is compliant; else 0.

```
In [13]: #"Compliance" . If the column "UCITSCompliance" is yes then 1 else 0
filtered_data['Compliance'] = filtered_data['UCITSCompliance'].apply(lambda x: 1 if x == 'Yes' else 0)
#filtered_data['Compliance'].unique()
#filtered_data['UCITSCompliance'].unique()
```

5. 'Volume Assessment: 1 if ETF has a volume greater than 10 million; else 0.

```
In [14]: #Volume" if the column "fundSizeMillions" is more than 10 Million then  good=1 else 0
filtered_data['Volume'] = filtered_data['fundSizeMillions'].apply(lambda x:1 if x > 10 else 0)
#filtered_data['fundSizeMillions'].info()
#filtered_data['Volume'].unique()
```

6. Risk Groups using K-means:

K-Means, an unsupervised machine learning clustering technique, partitions data into distinct groups or clusters within a dataset. Each object belongs to a single cluster, ensuring exclusivity and every cluster contains at least one object. These clusters are formed by considering the mathematical distance between each data point, to minimize the sum of distances within each cluster. In the context of financial risk assessment, for instance, 3 clusters represent 3 Risk Groups: Conservative, Moderate, and Aggressive, based on Year-to-Date Volatility. The next table represents the summary of the year today's return per each risk group.

Risk Group	# ETFs	Year Today Return		
		Min	Max	Mean
Conservative	999	0.001	0.183	0.126
Moderate	978	0.184	0.519	0.241
Aggressive	59	0.581	1.995	0.838

The Python code that makes the clusters for the risk groups is:

```
# K-means to get 3 risk groups - Conservative, Moderate and Aggressive

kmeans = KMeans(n_clusters=3, random_state=0).fit(filtered_data[['yearVolatilityCUR']])
filtered_data['Risk_Group'] = kmeans.labels_
```

```
# check the values of the distribution to assign the risk group

filtered_data.groupby('Risk_Group').yearVolatilityCUR.aggregate([np.count_nonzero, np.min, np.max, np.mean])
```

```
# assign the risk group

filtered_data['Risk_Group']= filtered_data['Risk_Group'].map({0: 'Moderate', 1: 'Conservative', 2: 'Aggressive'})
```

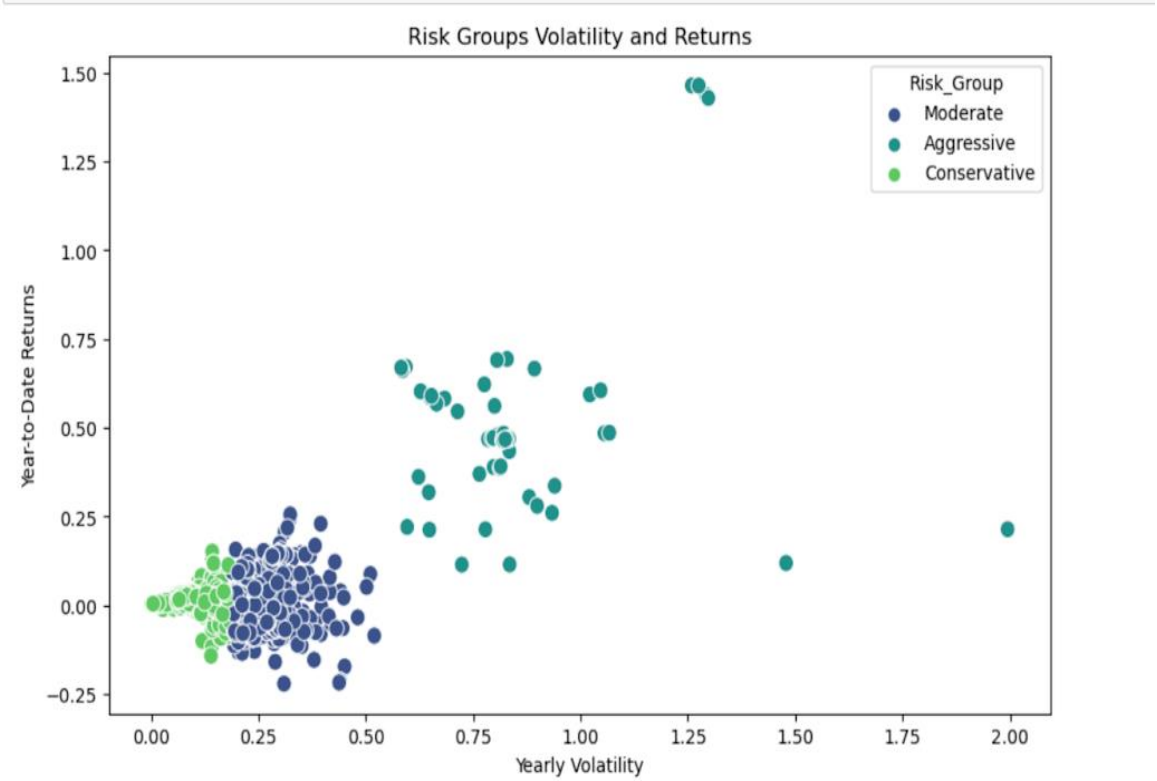
```
# check values of risk group

filtered_data.groupby('Risk_Group').yearVolatilityCUR.aggregate([np.count_nonzero, np.min, np.max, np.mean])
```

DATA VISUALIZATION AND ANALYSIS

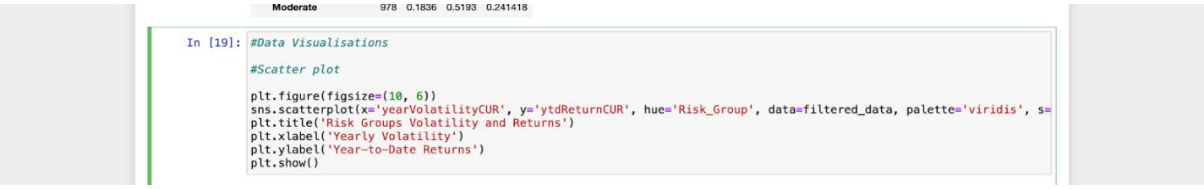
What is the relationship between volatility and return?

The primary objective of this analysis is to examine how Volatility influences Year-to-Date Returns across different Risk Groups.

Scatter plot:



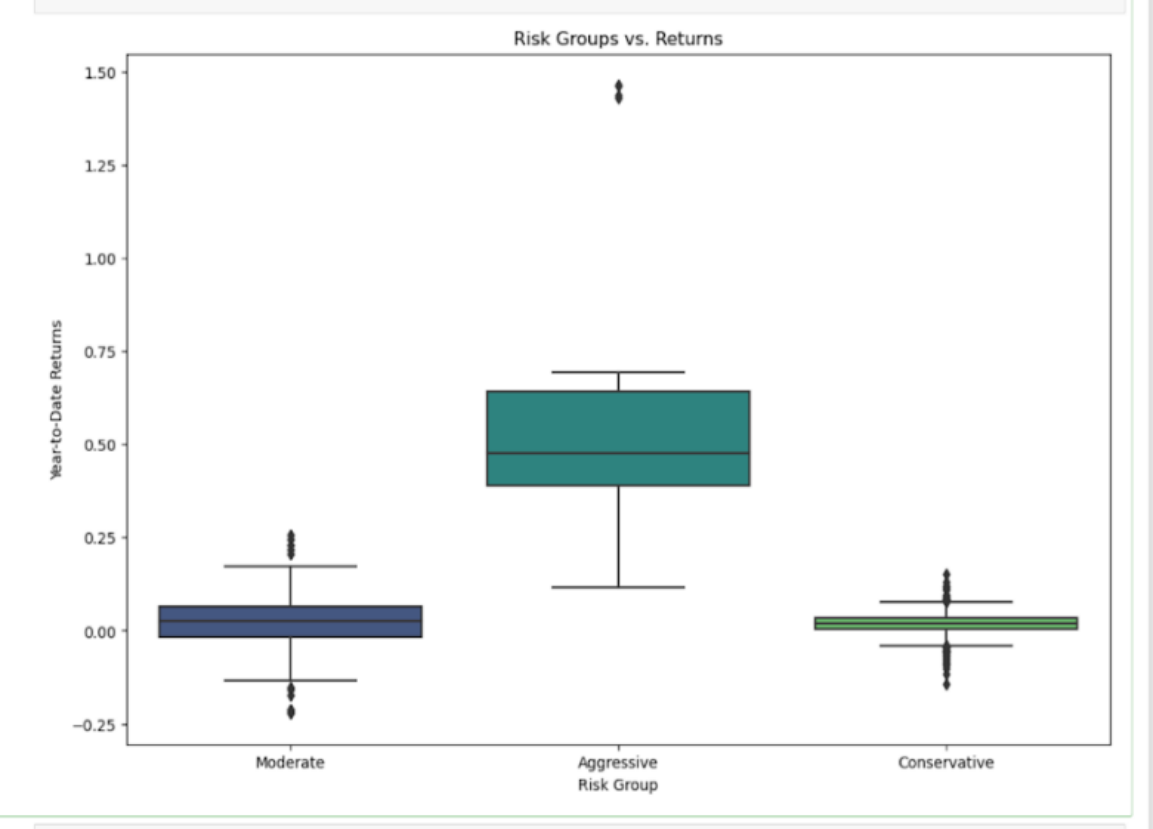
Code used to plot the scatter plot:



```
Moderate  978  0.1836  0.5193  0.241418

In [19]: #Data Visualisations

         #Scatter plot

         plt.figure(figsize=(10, 6))
         sns.scatterplot(x='yearVolatilityCUR', y='ytdReturnCUR', hue='Risk_Group', data=filtered_data, palette='viridis', s=
         plt.title('Risk Groups Volatility and Returns')
         plt.xlabel('Yearly Volatility')
         plt.ylabel('Year-to-Date Returns')
         plt.show()
```

- The graph presented is a scatter plot, visualizing data points based on the 'Yearly Volatility' (on the x-axis) and 'Year-to-Date Returns' (on the y-axis).
- Each data point is color-coded according to its respective Risk Group, providing a comprehensive view of the interplay between volatility, and returns.

Box Plot:



Code used for the Box Plot:

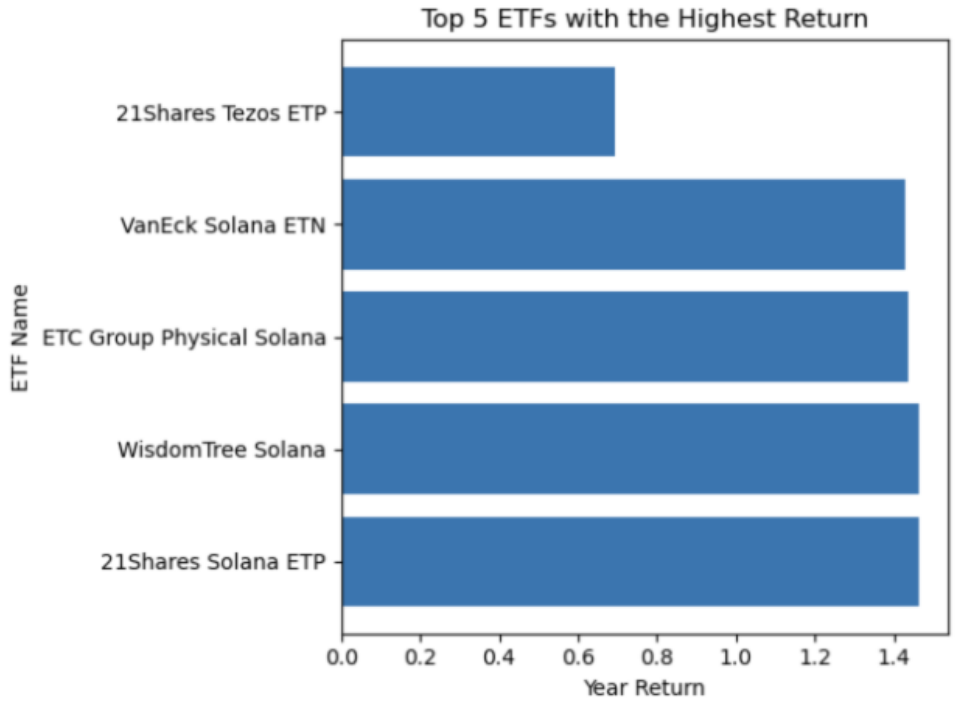
```
In [20]: #Boxplot

         plt.figure(figsize=(12, 8))
         sns.boxplot(x='Risk_Group', y='ytdReturnCUR', data=filtered_data, palette='viridis')
         plt.title('Risk Groups vs. Returns')
         plt.xlabel('Risk Group')
         plt.ylabel('Year-to-Date Returns')
         plt.show()
```

- The box plot encapsulates the distribution of returns, offering insights into central tendencies, variability, and potential outliers.
- The line within each box represents the median YTD Returns for the respective Risk Group. The box itself spans the Interquartile Range, indicating the middle 50% of Year-to-Date Returns.
- The length of the box reflects the variability within each Risk Group, offering a glimpse into the consistency of performance.
- The whiskers extend from the box to the minimum and maximum values within a certain range, typically 1.5 times the IQR (interquartile range).
- Outliers beyond the whiskers are represented as individual data points.

What are the top 5 ETFs with the best return?

Generally, the stock market tends to appreciate over the long term, while some companies exceed market performance and others fall behind. Consequently, index funds, known for their

low costs, typically offer substantial returns, representing a great investment option for any investor. It is important to emphasize the top 5 ETFs with the most attractive returns for future investors, this analysis could impact the decisions taken. The next figure represents the top 5 ETFs with the best return.

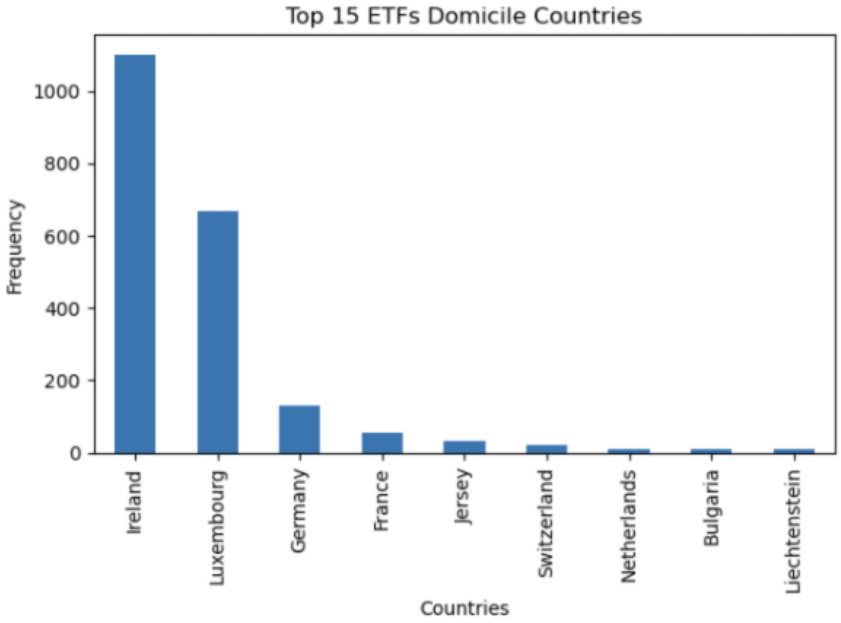


The Python code realized to get the previous plot is:

```
top_5_etfs = filtered_data.sort_values(by='ytdReturnCUR', ascending=False).head(5)

# Create a clustered bar chart
plt.barh(top_5_etfs['name'], top_5_etfs['ytdReturnCUR'],)

# Adding labels and title
plt.title('Top 5 ETFs with the Highest Return')
plt.xlabel('Year Return')
plt.ylabel('ETF Name')

# Save the plot as an image
plt.tight_layout()
plt.savefig('Top_5_ETFs_Highest_Return.png')
```

What is the concentration of ETFs per Domicile Country?

For international or global ETFs, it's crucial to consider the economic fundamentals and the currency's creditworthiness of the country they track. The success of an ETF investing in a specific country or region is significantly influenced by economic and social stability. These aspects are essential to consider when evaluating the potential success of an ETF. In addition, if a future investor is aiming to invest in more than one ETF, the domicile country of the ETF takes more relevance because investing all the ETFs from the same country will increase the concentration and risk in the portfolio. The next figure shows the top 15 domicile countries from the ETFS in the Dataset. The countries with the most ETFs are Ireland, Luxembourg, Germany, and France.



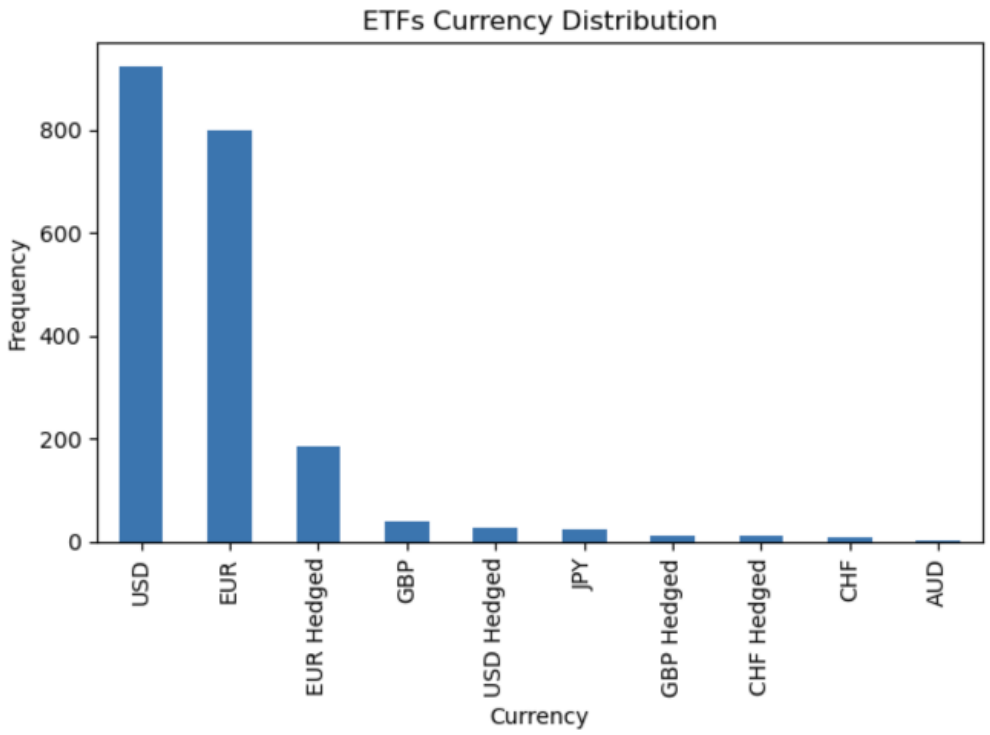
The Python code that made possible the previous graph is:

```
# Top Domicile Countries

# We are counting how many time each country is appearing
filtered_data['domicileCountry'].value_counts().head(15).sort_values(ascending=False).plot(kind='bar')

# Adding labels and title
plt.title('Top 15 ETFs Domicile Countries')
plt.xlabel('Countries')
plt.ylabel('Frequency')

# Save the plot as an image
plt.tight_layout()
plt.savefig('Top_15_ETFs_Countries.png')

plt.show()
```

What is the concentration of ETFs per currency?

Investing in ETFs with foreign assets carries currency risk, where exchange rate shifts impact investment value. For example, if a UK investor's US equities decrease in value due to a weakening dollar against the pound, the investment's pound value drops by 5%. Conversely, if the dollar strengthens by 5%, the investment's pound value increases by an equivalent percentage. These currency fluctuations can significantly affect the actual returns of overseas investments, highlighting the importance of considering currency risk in international portfolios.

Investment duration is key to currency risk exposure. Short-term investors, especially those needing liquidity, face greater risk from exchange rate swings, like that post-Brexit. For long-term investors, this risk diminishes and can be beneficial, as currency values tend to stabilize over time, offsetting volatility. Diverse global investments can inherently balance out, as gains in one currency can counteract losses in another. The next figure represents the distribution of the ETF currency distribution, so a future investor could be aware of possible currency risk. Notably, USD is the most frequent currency among the ETFs.

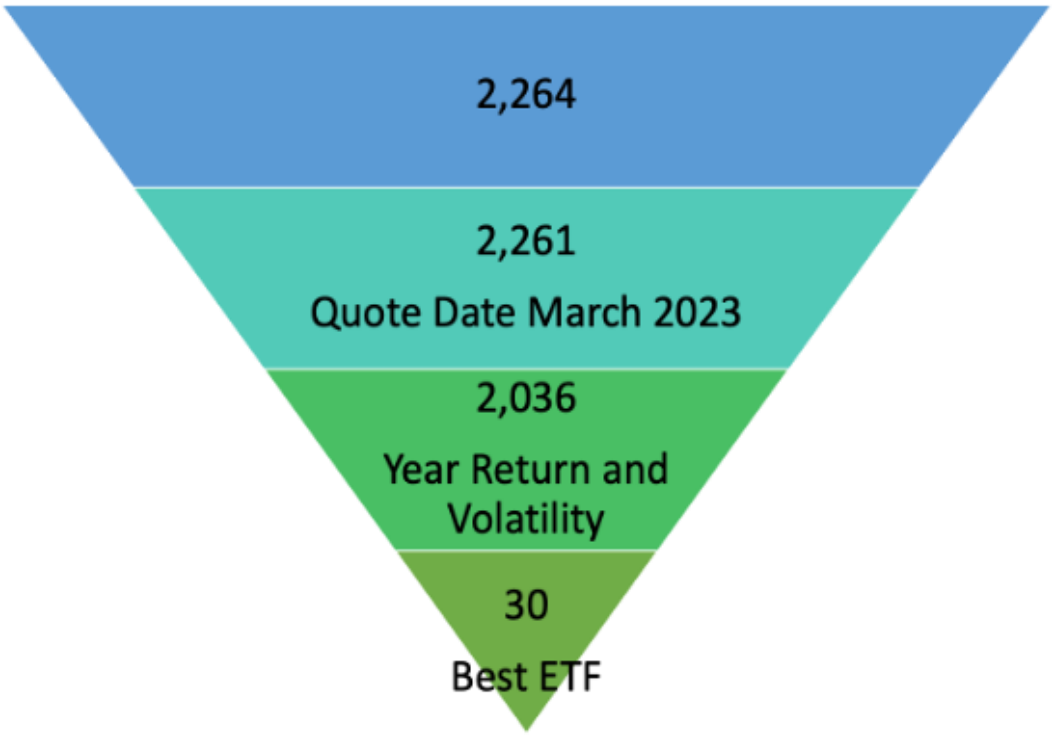


The code created to generate the previous graph is the next one:

```
# We are counting how many time each currency  is appearing

filtered_data['fundCurrency'].value_counts().head(10).sort_values(ascending=False).plot(kind='bar')

plt.title('ETFs Currency Distribution')
plt.xlabel('Currency')
plt.ylabel('Frequency')

# Save the plot as an image
plt.tight_layout()
plt.savefig('ETFs_Currency_Distribution.png')

plt.show()
```

THE BEST ETFs

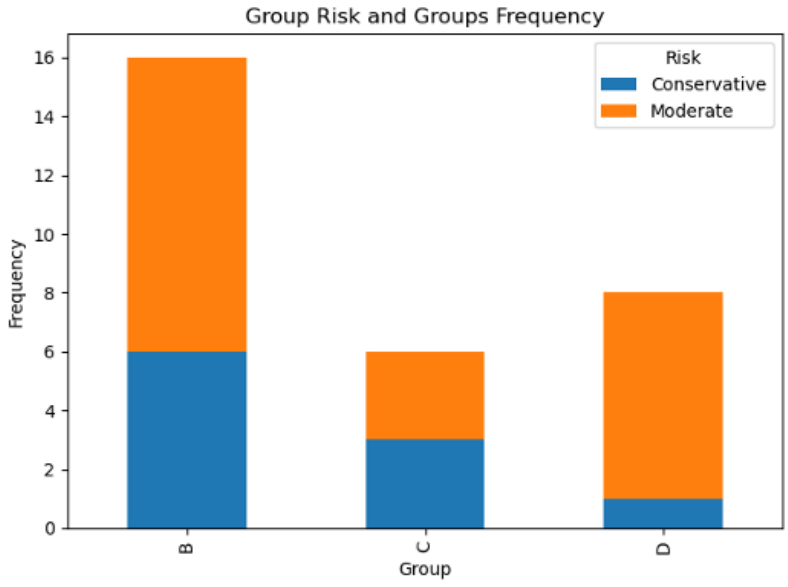
After filtering the ETFs that satisfy the different characteristics of a good ETF from the methodology explained previously, 30 ETFs met it. However, there are no ETFs that accomplish the filters for aggressive risk profiles.



The code that allows to filter the best ETFs is:

```
# best ETF that qualify the logic parameters in volume, currency among others

best_etf = filtered_data[(filtered_data['Group']!='A') &
(filtered_data['Volume']==1) & (filtered_data['Compliance']==1)&
(filtered_data['Currency_Risk']==1)& (filtered_data['Fee_comparison']==1)]
best_etf
```

The next visualization shows the distribution by risk group of the best 30 ETFs. 20 of the 30 best ETFs have a moderate risk level. In addition, group B (year return 4%-6%) includes 16 out of the best ETFs.



The code that performs the previous plot is:

```
grouped_data = best_etf.groupby(['Group', 'Risk_Group']).size().unstack(fill_value=0)

# Plotting the grouped bar chart
grouped_data.plot(kind='bar', stacked=True)

# Adding labels and title
plt.title('Group Risk and Groups Frequency')
plt.xlabel('Group')
plt.ylabel('Frequency')
plt.legend(title='Risk')


# Save the plot as an image
plt.tight_layout()
plt.savefig('Group_Risk_Groups_Frequency.png')

# Display the grouped bar chart
plt.show()
```

DATA MODELLING

Our goal with this modeling project is to evaluate how the year volatility of the top ETFs affected their return (**ytdreturn**). The data set post-transactions had 30 observations in different risk groups. We used a basic linear regression to create the model to predict the **ytdreturn** using the volatility.

- **Code for the Model:**

```
In [26]: #We want to analyze how the total bill influence the tip.
         model = smf.ols(formula = 'ytdReturnCUR ~ yearVolatilityCUR', data = best_etf)

In [27]: # The fit() method allows to fit the data to the model
         results= model.fit()

In [28]: # The summary() method allows to display the results

         print(results.summary())
```

- **The best equation:** Y(YearReturn) = 0.0011 + 0.3126 YearVolatility.

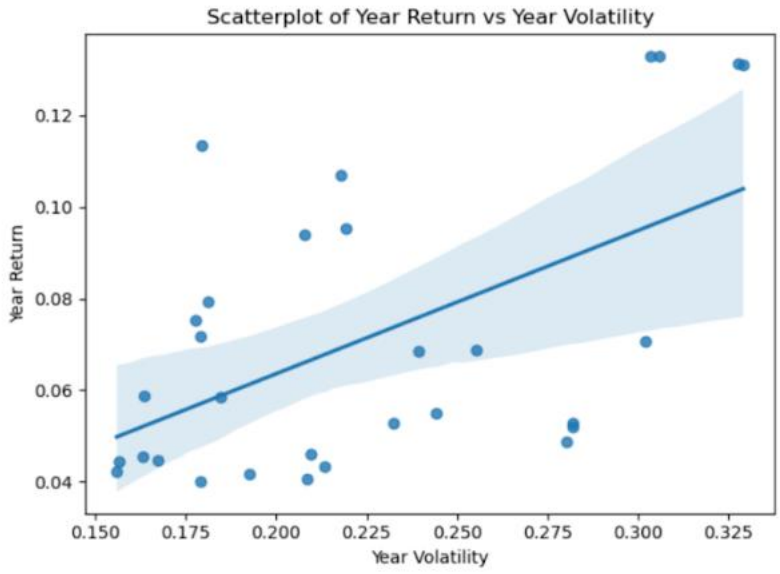
- Know the Slope and intercept of the model.

```
In [29]: #If we just want the coefficients, we can use the attribute params on results
         print(results.params)

Intercept           0.001096
yearVolatilityCUR   0.312577
dtype: float64
```

- Scatter plot to show how the Year Return is scattered with respect to the volatility.

```
In [30]: #Option 2: Create an Axes object using regplot

         reg,ax = plt.subplots()

         #use fit_reg = False if you do not want the regression line
         sns.regplot(x='yearVolatilityCUR', y='ytdReturnCUR', data=best_etf, ax =ax)

         ax.set_title('Scatterplot of Year Return vs Year Volatility')
         ax.set_xlabel('Year Volatility')
         ax.set_ylabel('Year Return')

         # Save the plot as an image
         plt.tight_layout()
         plt.savefig('linear_regression_plot.png')

         plt.show()
```

- The final model shows a linear relationship between Year Volatility and Year Return, with Y(YearReturn) = 0.0011 + 0.3126 YearVolatility.
- The coefficient of 0.3126 indicates that the Year Return is anticipated to increase by 0.3126 times for every unit increase in Year Volatility.
- Although its practical significance might be restricted, the intercept, 0.0011, reflects the lowest projected Year Return when Year Volatility is zero.
- The regression equation essentially offers a framework for prediction, showing that an increase in Year Volatility corresponds to a corresponding rise in projected Year Return.



MODEL VALIDATION

- The Summary stats:

```
In [28]: # The summary() method allows to display the results

         print(results.summary())
```

OLS Regression Results

Dep. Variable:	ytdReturnCUR	R-squared:	0.296
Model:	OLS	Adj. R-squared:	0.271
Method:	Least Squares	F-statistic:	11.76
Date:	Mon, 04 Dec 2023	Prob (F-statistic):	0.00189
Time:	23:58:40	Log-Likelihood:	67.161
No. Observations:	30	AIC:	−130.3
Df Residuals:	28	BIC:	−127.5
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	0.0011	0.021	0.052	0.959	−0.042	0.044
yearVolatilityCUR	0.3126	0.091	3.429	0.002	0.126	0.499

Omnibus:	3.129	Durbin-Watson:	0.910
Prob(Omnibus):	0.209	Jarque-Bera (JB):	1.805
Skew:	0.340	Prob(JB):	0.405
Kurtosis:	2.009	Cond. No.	19.6

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

- The F-statistic was used to assess the model's statistical significance; the result was a value of 11.76 with a matching p-value of 0.00189.
- The model is considered statistically significant at a 5% alpha level, meaning that the YearVolatility predictor's slope is not zero.
- YearVolatility accounts for 29.6% of the variance in YearReturn, according to the R-squared value of 0.296. This suggests a considerable degree of explanatory power, but the model is still not a very good return predictor.
- Taking this into account, we see that adding more variables to the model has the potential to improve forecast accuracy.
- The presented results further our knowledge of the connection between year volatility and returns, paving the way for the model to be further improved and expanded to include more variables for a more complete and reliable forecasting framework.

FINAL DATASET

	inceptionDate	quoteDate	ter	quote	fundSizeMillions	isin	ytdReturnCUR	yearVolatilityCUR	currencyRisk	UCITSCompliance	domicileCountry	fundCurren
482	2015-01-30	2023-03-21	0.0020	185.96	53.0	FR0012399772	0.1070	0.2180	Currency hedged	Yes	France	GBP Hedg
483	2015-01-07	2023-03-21	0.0020	184.85	32.0	FR0012399806	0.0953	0.2194	Currency hedged	Yes	France	USD Hedg
821	2017-10-24	2023-03-17	0.0010	3.73	130.0	IE00BD8PH067	0.0688	0.2552	Currency hedged	Yes	Ireland	CHF Hedg

CONCLUSION: BENEFITS AND CHALLENGES

Our in-depth analysis of Exchange-Traded Funds (ETFs) has provided valuable insights into the dynamics of this investment landscape. As we conclude our exploration, it's essential to highlight the benefits and challenges uncovered during our project.

Benefits

Our analysis of Exchange-Traded Funds (ETFs) provides valuable insights into top-performing options, aiding investors in strategic portfolio construction. The identification of global diversification opportunities and a nuanced understanding of the risk-return relationship empower data-driven decision-making for enhanced investment outcomes.

Challenges

Despite the benefits, our ETF analysis project faces challenges, including limited data granularity, the intricacies of market dynamics, opportunities for model improvement, and susceptibility to external economic factors. Acknowledging these challenges underscores the need for continuous refinement and adaptability in navigating the dynamic ETF landscape.

Learnings

Data cleaning is a vital process in any database project, ensuring the accuracy, consistency, and integrity of the information. By eliminating errors, duplicates, null values, and inconsistencies,

clean data enhances the quality of insights, facilitates efficient operations, and promotes trust among users and stakeholders. It is an essential step for, efficient decision-making, and the prevention of errors that could compromise the success of the overall project.