📈 Dividend Discount Model Data Collection and Export

This guide provides instructions for collecting data for the Dividend Discount Model (DDM) using Python libraries. The steps focus on retrieving, cleaning, and exporting historical dividend data for analysis.

---

1. Setting Up Data Collection

Initialize Data Structures

1. **Create an Empty DataFrame**:

   - Start by creating an empty DataFrame, `all_data`, to store combined data for all stock tickers.

2. **Define Stock Tickers**:

   - Prepare a list of stable dividend-paying stocks for analysis. Here's an example of stocks known for reliable dividends:
     - **IBM**: International Business Machines Corp.
     - **NEE**: NextEra Energy
     - **CAT**: Caterpillar Inc.
     - **O**: Realty Income Corporation
     - **ALB**: Albemarle Corporation
     - **ESS**: Essex Property Trust
     - **BRO**: Brown & Brown, Inc.
     - **WST**: West Pharmaceutical Services, Inc.
     - **ECL**: Ecolab Inc.
     - **CVX**: Chevron Corporation

---

2. Data Collection Steps

Library Imports

The following libraries are essential for handling and processing data:

- **yfinance**: Retrieves historical stock data, including prices and dividends.
- **pandas**: Manages data structures like DataFrames for handling large datasets.
- **datetime**: Sets up date ranges for data retrieval.

- **numpy**: Provides support for numerical operations on arrays.

Set Date Range for Data Retrieval

Define the start and end dates for the analysis:

- **End Date**: Set to today's date.
- **Start Date**: Typically set 10 years prior to today for a comprehensive analysis period.

Download and Filter Data

1. **Loop through Each Ticker**:
    - For each stock ticker in the list, download historical data for the specified date range.
2. **Download and Format Data**:
    - Use the `yfinance` library to download historical stock data, including prices and dividends.
    - Format the date index to display only the date (removing timestamps) for easier reading in the final dataset.
3. **Filter for Dividend Payments**:
    - Include only dates where dividends were paid (i.e., rows where the `Dividends` column is greater than zero).
    - Select essential columns, such as `Adj Close`, `Close`, and `Dividends`, to focus on relevant data for dividend analysis.
4. **Handle Missing Data**:
    - Remove any rows with NaN values to ensure that the dataset is clean and ready for analysis.
5. **Display and Verify Data**:
    - Display the dividend data for each stock ticker to ensure accuracy before merging it into the `all_data` DataFrame.

---

3. Export Data to Excel

1. **Save Dividend Data to Excel**:
    - Create an Excel file and add a separate sheet for each ticker's dividend data.
2. **Save Latest Closing Prices**:
    - Save the latest closing prices in a separate sheet within the Excel file for reference. This provides a snapshot of each stock's latest close price alongside historical

dividend data.

Output:

- The resulting file, `Dividend Discount Model.xlsx`, will contain:
  - Separate sheets for each stock's dividend data.
  - A sheet titled "Close Price Today" with the latest closing prices for each ticker.

---

4. Conclusion

This guide walks you through the data collection and export process for the Dividend Discount Model. With this data in Excel, you’re ready to proceed to DDM analysis and calculate additional metrics for portfolio optimization.

📊 Dividend Discount Model (DDM) Project

This project covers the Dividend Discount Model (DDM) step-by-step, including an example calculation and instructions for implementing it in Excel.

---

1. Understanding the DDM Concept

The **Dividend Discount Model (DDM)** predicts the price of a company's stock based on the sum of the present value of its expected dividends.

1. **Objective**: Determine if a stock is undervalued or overvalued by comparing the calculated intrinsic value to the market price.
2. **Required Rate of Return**: Set a required rate of return based on the stock's risk. For this example, we use **8%**.

---

2. Steps to Calculate DDM Value

Define Expected Dividends and Stock Price

1. Set a timeline, (T = 0) as today, and forecast expected dividends and stock price at future intervals:
   - **Year 1**: $8 dividend
   - **Year 2**: $10 dividend
   - **Year 3**: $6 dividend
   - **Year 4**: $12 dividend + expected stock price of$100

Apply the DDM Formula

Use the formula to discount each dividend and final stock price to present value:

$$\text{Intrinsic Value} = \frac{D_1}{(1+r)^1} + \frac{D_2}{(1+r)^2} + \cdots + \frac{D_n + P_n}{(1+r)^n}$$

where:

- $$(D_1, D_2, \ldots, D_n) = Dividendsateachtimepoint$$
- $$(P_n) = Expectedstockpriceatthefinalperiod$$
- $$(r) = Requiredrateofreturn(e.g.,8$$

Example Calculation

1. **Year 1**: Discount $8 at 8% → (\frac{8}{1.08} = 7.41)
2. **Year 2**: Discount $10 at 8% → (\frac{10}{(1.08)^2} = 8.57)
3. Continue this process and sum the discounted values.

Result: The total discounted cash flow is the **intrinsic value** of the stock.

---

3. Implementing DDM in Excel

1. **Get Historical Data**:
   - Go to Yahoo Finance, search for the stock (e.g., INTC), and download historical dividend data.
2. **Calculate Dividend Growth Rate**:
   - Import quarterly dividends, annualize them, and calculate the growth rate for each year.
3. **Calculate Intrinsic Price Using DDM**:
   - Use the DDM formula in Excel, inputting the required rate of return and expected dividend growth.

---

4. Using Excel Solver for Required Return

To match the intrinsic value with the actual market price, adjust the required rate of return using **Excel Solver**:

1. Open **Solver** under the Data tab.
2. Set the **DDM Price** cell equal to the **Actual Price** cell.

3. Change the **Required Rate of Return** cell until both values match.

Is the Stock Overvalued or Undervalued?

- **If Intrinsic Price > Market Price**: The stock is undervalued and may be a good buy.
- **If Intrinsic Price < Market Price**: The stock is overvalued.

```
In [ ]:
```