Empirical Application: Markov Switching Model

Charaf ZGUIOUAR, Majd RABBAJ, Meriem BENDIR
Université Panthéon-Sorbonne Paris 1 - Ecole de l'Economie - Master II Finance, Technology, Data

07th of December 2023

Abstract

This study extends the exploration of stock return predictability by employing a Markov Switching Model. Unlike traditional linear models, Markov Switching Models allow for changes in regime, capturing the dynamic behavior of financial time series data. Our analysis focuses on Microsoft's stock returns, considering various macroeconomic factors . The goal is to identify and characterize different market states, such as bull and bear conditions, and to understand their impacts on return dynamics. .

1 Introduction

The financial markets are known for their volatility and the transitory nature of their states, often influenced by a myriad of factors including investor sentiment, economic cycles, and global events. Traditional linear models often fall short in capturing this complexity, leading to a misrepresentation of risk and return dynamics. The Markov Switching Model presents a robust alternative, accommodating regime changes and providing a more nuanced understanding of market behavior.

2 Data Sources and Model Specification:

2.0.1 The choice of explanatory variables:

The study utilizes historical stock return data for Microsoft (MSFT) obtained from Yahoo Finance and macroeconomic indicators from the Federal Reserve Economic Data (FRED) from January 2010.

Google's returns are considered as a competitor return series to provide a comparative basis within the same industry. By including Google's returns, we aim to capture any competitive effects that might exist between the two tech companies, under the assumption that they may face similar market conditions and economic factors that impact their stock performance.

2.1 Macroeconomic Factors Influencing Stock Returns

2.2 Labor Market Conditions

2.2.1 Unemployment Rate (UNRATE)

The unemployment rate serves as an indicator of labor market health. A rising unemployment rate often signals weakening economic growth and reduced consumer spending, both of which can adversely impact stock returns.

2.3 Inflation and Price Levels

2.3.1 Consumer Price Index (CPIAUCSL)

The CPI measures inflation by tracking the price changes of a basket of consumer goods and services. Inflation influences the discount rates in stock valuation models, affecting stock prices.

2.4 Economic Growth

2.4.1 Gross Domestic Product (GDP)

GDP represents the total domestic production and is a key indicator of economic health. Robust GDP growth can indicate higher potential earnings for companies, which is positive for the stock prices of companies like Microsoft and Google.

2.5 Monetary Policy

2.5.1 Federal Funds Rate (FEDFUNDS)

The federal funds rate is the interest rate for overnight loans between depository institutions. Variations in this rate can affect other interest rates, modifying the cost of borrowing and the overall investment climate.

2.6 Investor Confidence

2.6.1 10-Year Treasury Yield (GS10)

The yield on 10-year U.S. Treasury notes is often regarded as a benchmark for long-term interest rates, reflecting investor confidence and affecting the equity risk premium.

2.7 Energy Market Dynamics

2.7.1 West Texas Intermediate (WTI) Oil Price (DCOILWTICO)

WTI Oil Price is a major benchmark for crude oil prices. Changes in oil prices can significantly influence the operational costs of businesses and the spending power of consumers, thereby affecting the broader economy and stock market.

2.8 Market Volatility

2.8.1 The VIX:

Often known as the "fear gauge," the VIX index measures the market's expectation of volatility. Higher VIX values typically correspond to increased market risk and investor concern, which can result in lower stock market performance.

3 Stationary Check and Data Transformation

A fundamental step in our analysis was to verify the stationarity of the time series data, which is a critical assumption in time series analysis and particularly pertinent for models like the OLS and Markov Switching. We utilized the Augmented Dickey-Fuller (ADF) test to assess stationarity in each of the time series, including the stock returns of Microsoft (MSFT), Google (GOOGL), and various economic indicators.

In instances where time series were identified as non-stationary, we employed differencing as a method of transformation. Differencing is effective in stabilizing the mean of a time series, thereby mitigating or eliminating trends and seasonality. This process ensures that the transformed data adhere to the stationarity prerequisite essential for accurate model estimation. The blue bars show that the ADF statistic for each time series is below the 5% critical value, which means that for each of these time series, the null hypothesis of a

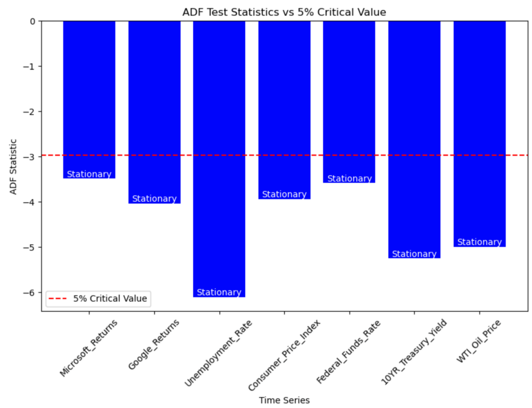

Figure 1: Stationarity in all series

unit root can be rejected at the 5% significance level. This implies that each of the time series is considered stationary based on this test.

4 Estimation of Linear Model:

OLS Regression Results

Dep. Variable:	Microsoft_Returns	R-squared:	0.829
Model:	OLS	Adj. R-squared:	0.761
Method:	Least Squares	F-statistic:	12.15
Date:	Wed, 13 Dec 2023	Prob (F-statistic):	3.60e-06
Time:	00:28:25	Log-Likelihood:	113.80
No. Observations:	29	AIC:	-209.6
Df Residuals:	20	BIC:	-197.3
Df Model:	8		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.1670	0.071	2.361	0.028	0.019	0.314
Google_Returns	0.5854	0.107	5.495	0.000	0.363	0.808
VIX	0.0008	0.000	2.400	0.026	0.000	0.002
Unemployment_Rate	-0.0021	0.001	-1.991	0.060	-0.004	0.000
Consumer_Price_Index	-0.0015	0.001	-2.568	0.018	-0.003	-0.000
Gross_Domestic_Product	1.02e-05	3.84e-06	2.656	0.015	2.19e-06	1.82e-05
Federal_Funds_Rate	-0.0032	0.002	-1.652	0.114	-0.007	0.001
10YR_Treasury_Yield	0.0037	0.002	1.529	0.142	-0.001	0.009
WTI_Oil_Price	9.5e-05	9.98e-05	0.952	0.352	-0.000	0.000

Omnibus:	1.969	Durbin-Watson:	2.837
Prob(Omnibus):	0.374	Jarque-Bera (JB):	1.570
Skew:	-0.559	Prob(JB):	0.456

Figure 2: OLS results

The Ordinary Least Squares (OLS) regression model developed to understand the relationship between Microsoft's stock returns and various explanatory variables including Google's returns, VIX, and selected macroeconomic indicators, is represented by the following linear equation:

$$Microsoft\_Returns = \beta_0 + \beta_1 \times Google\_Returns + \beta_2 \times VIX + \beta_3 \times Unemployment\_Rate + \ldots + \epsilon \quad (1)$$

where:

- β_0 is the intercept of the model.
- $\beta_1, \beta_2, \beta_3, \ldots$ are the coefficients representing the impact of Google's returns, VIX, unemployment rate, and other factors on Microsoft's returns.
- ϵ is the error term, capturing the variation in Microsoft's returns not explained by the model.

The coefficients $\beta_1, \beta_2, \beta_3, \ldots$ are estimated through the OLS regression analysis and provide insights into the strength and nature of the relationships between the dependent variable (Microsoft's returns) and the independent variables (Google's returns, VIX, etc.).

The OLS regression results indicate that the model explains a significant portion of the variability in Microsoft's stock returns, with an R-squared value , implying that approximately 82.9

The F-statistic of 12.15 and its associated Prob (F-statistic) of 3.60e-06 suggest that the model is statistically significant at conventional levels. This means that the independent variables, collectively, have a significant impact on Microsoft's returns.

The Durbin-Watson statistic of 2.837 indicates there is no autocorrelation in the residuals.

5 Markov switching model:

In our analysis, we applied the Markov Switching Model to the stock returns of Microsoft (MSFT) to identify different regimes in the stock's performance over time. The Markov Switching Model is particularly adept at capturing the dynamic nature of financial time series, which often exhibit different behaviors under varying market conditions. The model is fitted to the daily returns of Microsoft stock from January 1, 2010, to April 30, 2023. We ensure that NaN values are dropped from the return series before fitting the model.

5.1 Model Specification:

- The `MarkovRegression` model from the `statsmodels` library was employed.
- We defined two regimes (`k_regimes=2`) to represent different states in the stock market, such as bull and bear markets.
- The model included a constant term (`trend='c'`) to account for the average effect in each regime.
- We enabled `switching_variance=True` to allow the model to capture changes in volatility between different market regimes.
- We ensured the data was free from NaN values by applying the `dropna` method.

5.2 Model Estimation:

Upon fitting the Markov Switching Model to the data, the estimated parameters provided insights into the behavior of Microsoft stock returns in different regimes. The model's summary output included:

- The coefficients for each regime, indicating the mean return in each state. - The variance of the returns in each regime, reflecting the risk or volatility associated with each state. - Transition probabilities, offering a view of the likelihood of moving from one regime to another.

5.2.1 Markov Switching Model Equation:

The general equation for our Markov Switching Model is given by:

$$y_t = \mu_{S_t} + \epsilon_t, \quad \epsilon_t \sim N(0, \sigma^2_{S_t})$$

where:

- y_t represents the stock returns of Microsoft at time t.
- μ_{S_t} is the mean return for the regime S_t at time t.
- $\sigma^2_{S_t}$ is the variance of returns in regime S_t.
- ϵ_t is the error term, following a normal distribution with mean 0 and variance $\sigma^2_{S_t}$.
- S_t is the state or regime at time t, which can take a value of 1 or 2 in our two-regime model.

The model thus allows for different mean returns and variances in each regime, capturing the varying characteristics of the stock returns in different market conditions. The transition probabilities, not explicitly shown in the equation, are crucial in determining the likelihood of switching from one regime to another over time.

6 Empirical result:

```
                        Markov Switching Model Results
==============================================================================
Dep. Variable:                Returns   No. Observations:                 3353
Model:               MarkovRegression   Log Likelihood                9414.448
Date:                Tue, 05 Dec 2023   AIC                         -18816.895
Time:                        14:34:14   BIC                         -18780.190
Sample:                             0   HQIC                        -18803.767
                               - 3353
Covariance Type:               approx
                             Regime 0 parameters
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.0011      0.000      4.573      0.000       0.001       0.002
sigma2         0.0001   6.32e-06     17.507      0.000    9.83e-05       0.000
                             Regime 1 parameters
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const        7.83e-05      0.001      0.081      0.935      -0.002       0.002
sigma2         0.0007   6.07e-05     11.941      0.000       0.001       0.001
                         Regime transition parameters
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
p[0->0]        0.9565      0.008    124.771      0.000       0.942       0.972
p[1->0]        0.1224      0.025      4.992      0.000       0.074       0.171
==============================================================================
```

Figure 3: Markov switching model results

Based on the Markov Switching Regression results for Microsoft's stock returns, we can infer the following:

1. Regime 0 (Possibly Low Volatility Regime): The constant term is statistically significant with a very low p-value, suggesting that the returns in this regime have a distinct mean level that is different from zero. This regime appears to have lower variance (given the magnitude of sigma2), which might indicate a period of less volatility or more stable market conditions.

2. Regime 1 (Possibly High Volatility Regime): The constant term is not statistically significant, indicating that the mean return for this regime is not significantly different from zero. However, the significant variance (sigma2) for this regime indicates that while the mean return doesn't change, the volatility is higher compared to Regime 0. This could be associated with periods of market stress or uncertainty.

3. Regime Persistence and Transition: The transition probabilities demonstrate that there is a high likelihood (around 96.55%) of the market remaining in Regime 0 once it is in that state. This suggests that stable periods tend to persist once they occur. On the other hand, there is a smaller probability (around 12.24%) of switching from Regime 1 to Regime 0, suggesting that once the market enters a volatile phase, it's less likely but still possible to transition back to stability.

From these observations it seems that Microsoft's stock returns exhibit two different behaviors or regimes which can be characterized by their levels of volatility. One regime is more stable, possibly representing normal trading conditions, and the other is more volatile, which may represent times of market stress or downturns. The significant variance in both regimes indicates that volatility is an important factor in modeling Microsoft's stock returns, and the model has successfully captured these dynamics.

Investors and analysts could use this information to adjust their strategies based on the regime the market is currently in. For instance, they might adopt a more conservative approach during the volatile regime and a more aggressive strategy in the stable regime, depending on their risk tolerance and investment horizon.

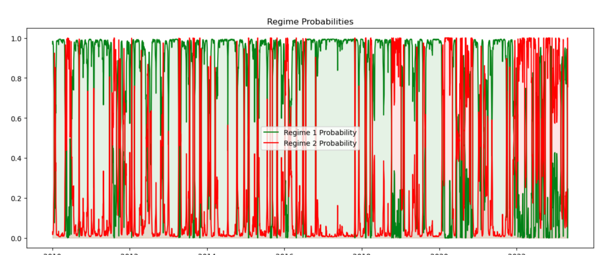

Figure 4: Regime probabilities

Figure 5: Regime 1 and 2 mean returns

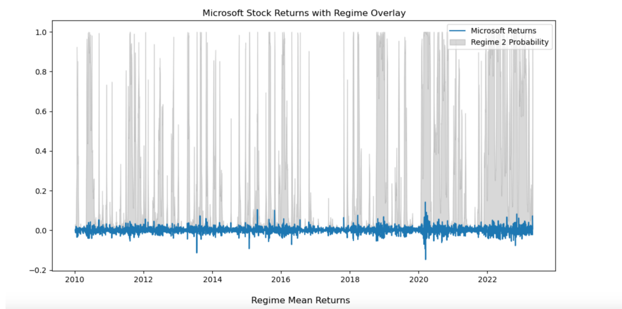

Figure 6: Microsoft return with regime overlay

6.1 Validation and Specification Tests

Regression Output Analysis (Regime-Switching Model):

- The significant constant term in the first regime implies a meaningful intercept.
- The variance in both regimes is significantly different from zero, indicating variability in the data is well-captured by the model.
- Transition probabilities suggest a high likelihood of staying in regime 0 and a lower probability of switching from regime 1 to regime 0, indicating distinct behavioral states in the data.

A key part of this preliminary analysis involved assessing the presence of heteroscedasticity, a common feature in financial datasets where the variance of the errors may not be constant across observations.To this end, we employed the Breusch-Pagan test, a widely used statistical method for detecting heteroscedasticity. The test's primary objective is to ascertain whether the variance of the residuals from a regression model is dependent on the values of the independent variables. Durbin-Watson Test for Autocorrelation: A statistic of 2.837 indicates slight negative autocorrelation, but it's generally not a significant concern as it's close to the ideal value of 2. shapiro-Wilk Test for Normality:The p-value of 0.473 suggests that the residuals are normally distributed, meeting a key assumption of many regression models.

```
In [31]: from scipy.stats import shapiro
         stat, p = shapiro(residuals)
         print('Statistics=%.3f, p=%.3f' % (stat, p))

         Statistics=0.967, p=0.473

In [32]: from statsmodels.stats.stattools import durbin_watson
         dw = durbin_watson(residuals)
         print('Durbin-Watson:', dw)

         Durbin-Watson: 2.837130209738209
```

Figure 7: Enter Caption

7 Conclusion:

This study has successfully demonstrated the effectiveness of the Markov Switching Model in capturing the dynamic nature of financial markets, particularly in the context of Microsoft's stock returns. Our analysis reveals that Microsoft's stock performance is significantly influenced by various macroeconomic factors and exhibits distinct behaviors in different market regimes. The model has identified two primary regimes: one characterized by lower volatility, likely representing stable market conditions, and another by higher volatility, possibly indicative of turbulent market phases.

The empirical results from the Markov Switching Model provide valuable insights for investors and financial analysts. They underscore the importance of considering market regimes in investment strategies. The distinct characteristics of each regime, as revealed by the model, allow for more tailored investment approaches. For instance, a more conservative strategy might be preferable during high volatility periods, while a more aggressive stance could be advantageous in stable conditions.

Moreover, the study highlights the limitations of traditional linear models in financial market analysis. The Markov Switching Model, with its ability to adapt to regime changes, offers a more nuanced and accurate representation of stock market dynamics. This approach is particularly relevant in today's rapidly evolving financial landscape, where understanding the implications of economic indicators and market sentiments is crucial.

In conclusion, the application of the Markov Switching Model to Microsoft's stock returns provides a robust framework for analyzing stock market behavior. It offers a comprehensive view of how different market conditions impact stock performance, enabling more informed and strategic investment decisions. Future research could expand on this approach by incorporating additional variables or exploring other sectors, further enriching our understanding of market dynamics.

In [115…

```python
import yfinance as yf
import pandas as pd
import statsmodels.api as sm
from datetime import datetime
import pandas_datareader.data as web

# Define the start and end dates for the data
start_date = '2010-01-01'
end_date = datetime.today().strftime('%Y-%m-%d')

# Define the ticker symbols for Microsoft and Alphabet (Google)
microsoft_symbol = 'MSFT'
google_symbol = 'GOOGL'

# Function to get returns
def get_returns(ticker, start_date, end_date):
    data = yf.download(ticker, start=start_date, end=end_date)
    data['Returns'] = data['Close'].pct_change()
    return data['Returns']

# Get stock data
microsoft_returns = get_returns(microsoft_symbol, start_date, end_date)
google_returns = get_returns(google_symbol, start_date, end_date)

# Define the selected series IDs from FRED
selected_series_ids = {
    'Unemployment_Rate': 'UNRATE',
    'Consumer_Price_Index': 'CPIAUCSL',
    'Gross_Domestic_Product': 'GDP',
    'Federal_Funds_Rate': 'FEDFUNDS',
    '10YR_Treasury_Yield': 'GS10',
    'WTI_Oil_Price': 'DCOILWTICO'
}

# Download the selected economic data
economic_data = {}
for series_name, series_id in selected_series_ids.items():
    economic_data[series_name] = web.DataReader(series_id, 'fred', start_date, end_date)

# Get VIX data
vix_data = yf.download('^VIX', start=start_date, end=end_date)['Close']

# Combine stock returns, VIX, and economic data into a single DataFrame
combined_df = pd.concat([microsoft_returns, google_returns, vix_data] + list(economic_data.values()), axis=1)
combined_df.columns = ['Microsoft_Returns', 'Google_Returns', 'VIX'] + list(economic_data.keys())
combined_df.dropna(inplace=True)

# Define the dependent variable and independent variables
y = combined_df['Microsoft_Returns']
X = combined_df.drop('Microsoft_Returns', axis=1)

# Add a constant to the model (intercept)
X = sm.add_constant(X)

# Fit the model
model = sm.OLS(y, X).fit()

# Print the summary of the model
print(model.summary())

from statsmodels.tsa.stattools import adfuller



# Function to perform ADF test
def adf_test(series, name=''):
    result = adfuller(series.dropna())  # dropna() handles differenced data
    print(f'ADF Statistic for {name}: {result[0]}')
    print(f'p-value for {name}: {result[1]}')
    for key, value in result[4].items():
        print(f'Critical Values ({key}): {value}')

# Perform ADF test for each series
print("\nTesting for Stationarity:")
adf_test(combined_df['Microsoft_Returns'], 'Microsoft Returns')
adf_test(combined_df['Google_Returns'], 'Google Returns')
adf_test(combined_df['VIX'], 'VIX')
for key in economic_data.keys():
    adf_test(combined_df[key], key)

# Fit the model
model = sm.OLS(y, X).fit()

# Print the summary of the model
print(model.summary())
```

```python
# Create differenced series for Microsoft Returns
combined_df['Microsoft_Returns'] = combined_df['Microsoft_Returns'].diff().dropna()

# Create differenced series for Google Returns
combined_df['Google_Returns'] = combined_df['Google_Returns'].diff().dropna()

# Create differenced series for Unemployment Rate
combined_df['Unemployment_Rate'] = combined_df['Unemployment_Rate'].diff().dropna()

# Create differenced series for Consumer Price Index (CPI)
combined_df['Consumer_Price_Index'] = combined_df['Consumer_Price_Index'].diff().dropna()

# Create differenced series for Federal Funds Rate
combined_df['Federal_Funds_Rate'] = combined_df['Federal_Funds_Rate'].diff().dropna()

# Create differenced series for 10-Year Treasury Yield
combined_df['10YR_Treasury_Yield'] = combined_df['10YR_Treasury_Yield'].diff().dropna()

# Create differenced series for WTI Oil Price
combined_df['WTI_Oil_Price'] = combined_df['WTI_Oil_Price'].diff().dropna()
```

```python
from statsmodels.tsa.stattools import adfuller

# Function to perform ADF test
def adf_test(series, name=''):
    result = adfuller(series.dropna())  # dropna() handles differenced data
    print(f'ADF Statistic for {name}: {result[0]}')
    print(f'p-value for {name}: {result[1]}')
    for key, value in result[4].items():
        print(f'Critical Values ({key}): {value}')

# Perform ADF test for each series
adf_test(combined_df['Microsoft_Returns'], 'Microsoft Returns')
adf_test(combined_df['Google_Returns'], 'Google Returns')
adf_test(combined_df['Unemployment_Rate'], 'Unemployment Rate')
adf_test(combined_df['Consumer_Price_Index'], 'Consumer Price Index (CPI)')
adf_test(combined_df['Federal_Funds_Rate'], 'Federal Funds Rate')
adf_test(combined_df['10YR_Treasury_Yield'], '10-Year Treasury Yield')
adf_test(combined_df['WTI_Oil_Price'], 'WTI Oil Price')
```

```python
from statsmodels.stats.diagnostic import het_breuschpagan
bp_test = het_breuschpagan(model.resid, model.model.exog)
print("Breusch-Pagan Test Statistic:", bp_test[0])
print("Breusch-Pagan Test p-value:", bp_test[1])
```

```
Breusch-Pagan Test Statistic: 6.3334211828678155
Breusch-Pagan Test p-value: 0.6099377898317198
```

```python
import statsmodels.api as sm
import yfinance as yf
import pandas as pd
from statsmodels.tsa.regime_switching.markov_regression import MarkovRegression

# Assuming you have already defined the function get_returns and the date range
start_date = '2010-01-01'
end_date = '2023-04-30'  # Replace with your end date
microsoft_symbol = 'MSFT'

# Get Microsoft stock returns
microsoft_returns = get_returns(microsoft_symbol, start_date, end_date)

# Drop NaN values
microsoft_returns.dropna(inplace=True)

# Define the Markov Switching model
# Here, k_regimes defines the number of regimes you want to consider
# You can adjust the order and other parameters based on your specific needs
ms_model = MarkovRegression(microsoft_returns, k_regimes=2, trend='c', switching_variance=True)

# Fit the model
ms_results = ms_model.fit()

# Print the summary of the model
print(ms_results.summary())
```

```
[*********************100%***********************]  1 of 1 completed
```