Empirical Application: Smooth Transition Regression (STR) Models

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 delves into the utilization of Smooth Transition Regression (STR) models in the realm of financial time series analysis, with a specific emphasis on the dynamics of stock market data. We employ daily returns data from Apple Inc. (AAPL) and its main competitor, Microsoft Corp. (MSFT), in conjunction with the comprehensive Fama-French five-factor model and a selection of pertinent economic indicators. The primary objective of this research is to harness the power of STR models to capture the intricate nonlinear patterns inherent in stock returns. Our empirical findings strongly indicate that STR models outperform conventional linear models in modeling financial time series data. This discovery underscores the considerable potential of STR models in the fields of financial forecasting and analysis.

1 Introduction

The financial market is renowned for its inherent volatility and complex nonlinear characteristics. Conventional linear models often prove inadequate in comprehensively capturing the multifaceted dynamics at play within this intricate landscape. The central aim of this study is to leverage the capabilities of Smooth Transition Regression (STR) models in analyzing financial time series data, with a specific focus on the stock returns associated with Apple Inc. (AAPL). The structure of this paper is organized to provide a foundational understanding of traditional linear modeling techniques, subsequently delving into the formulation and estimation of STR models. Finally, we culminate in an in-depth comparative analysis that unequivocally underscores the superior performance of STR models in the realm of financial time series data analysis and their immense potential for application in financial forecasting and decision-making.

2 Data Sources and Model Specification:

2.0.1 The choice of explanatory variables:

The primary data used in this study includes daily stock returns of Apple Inc. (AAPL) and Microsoft Corp. (MSFT) from January 1, 2010, to the present. his period captures a range of market conditions, including the volatility and market dynamics associated with the COVID-19 pandemic. The stock data is sourced from Yahoo Finance, which is widely recognized for providing comprehensive financial information.

2.1 The Fama-French five-factor model data

In addition to stock returns, this study incorporates the Fama-French five-factor model data for the same period . The Fama-French model, initially introduced by Eugene F. Fama and Kenneth R. French, is a widely used asset pricing model in finance. It extends the Capital Asset Pricing Model (CAPM) by adding size, value, profitability, and investment factors to the market risk factor. The five factors are:

2.1.1 Market Risk (MKT):

The excess return of the market over the risk-free rate.

2.1.2 Size (SMB, Small Minus Big):

The observed tendency for smaller companies to outperform larger ones.

2.1.3 Value (HML, High Minus Low):

The tendency for "value" stocks with high book-to-market ratios to outperform "growth" stocks.

2.1.4 Profitability (RMW, Robust Minus Weak):

The tendency for firms with high operating profitability to outperform those with low profitability.

2.1.5 Investment (CMA, Conservative Minus Aggressive):

The tendency for firms with conservative investment strategies to outperform those with aggressive strategies.

The Fama-French five-factor model data is sourced from the Kenneth R. French Data Library, an authoritative source for academic research in finance. This model provides a more nuanced view of risk factors affecting stock returns and is particularly useful in understanding the risk and return profiles of different stocks. By combining stock returns with the Fama-French factors and macroeconomic indicators, this study aims to capture a comprehensive view of the market dynamics influencing AAPL and MSFT stock performance.

2.2 Macroeconomic Factors and Model Simplification

We initially included explanatory variables such as Real GDP, Consumer Price Index, Unemployment Rate, Retail Sales Excluding Food Services, Industrial Production, Personal Saving Rate, Retail Sales, and E-Commerce Retail Sales in our linear regression model to explain Apple Inc. (AAPL) returns. However, our analysis revealed that these variables were not statistically significant in explaining AAPL returns. As a result, we have decided to remove them from the model.

	coef	std err	t	P>|t|	[0.025	0.975]
nst	-0.0314	0.034	-0.915	0.362	-0.099	0.036
crosoft_Returns	0.4982	0.088	5.673	0.000	0.325	0.671
nsumer_Sentiment	0.0001	8.55e-05	1.423	0.157	-4.71e-05	0.000
YR_Treasury_Yield	0.0023	0.002	1.068	0.287	-0.002	0.007
rsonal_Consumption_Expenditures	7.142e-07	1.01e-06	0.707	0.481	-1.28e-06	2.71e-06
nsumer_Confidence	0.0001	8.55e-05	1.423	0.157	-4.71e-05	0.000
employment_Rate	-2.266e-05	0.001	-0.018	0.986	-0.003	0.002

	coef	std err	t	P>|t|	[0.025	0.975]
const	-0.1194	0.653	-0.183	0.872	-2.927	2.688
Real_GDP	2.599e-05	3.47e-05	0.750	0.532	-0.000	0.000
Consumer_Price_Index	-0.0023	0.002	-1.403	0.296	-0.009	0.005
Unemployment_Rate	0.0015	0.008	0.202	0.859	-0.031	0.034
10YR_Treasury_Yield	0.0097	0.020	0.495	0.669	-0.074	0.094
Retail_Sales_Ex_Food_Services	-1.002e-07	2.95e-07	-0.340	0.766	-1.37e-06	1.17e-06
Industrial_Production	0.0021	0.006	0.357	0.755	-0.023	0.027
Personal_Saving_Rate	-0.0027	0.005	-0.576	0.623	-0.023	0.017
S&P_500	-2.634e-05	1.25e-05	-2.115	0.169	-7.99e-05	2.72e-05
Retail_Sales	-9.988e-08	2.94e-07	-0.340	0.767	-1.37e-06	1.17e-06
E_Commerce_Retail_Sales	8.4e-07	8.58e-07	0.979	0.431	-2.85e-06	4.53e-06

Omnibus:	3.447	Durbin-Watson:	2.691
Prob(Omnibus):	0.178	Jarque-Bera (JB):	0.839
Skew:	-0.002	Prob(JB):	0.657
Kurtosis:	4.296	Cond. No.	3.63e+16

3 Stationarity Check and Data Transformation

Before proceeding with the model estimation, it was crucial to ensure that the time series data were stationary. Stationarity is a key assumption in time series analysis, particularly for models like STR. We employed the Dickey-Fuller test to check for stationarity in each time series, including stock returns of AAPL and MSFT, and the Fama-French factors.

For series that were found to be non-stationary, we applied differencing as a transformation method. Differencing helps stabilize the mean of a time series by removing changes in the level of a time series, and thus eliminating (or reducing) trend and seasonality. After differencing, the stationarity of each series was re-evaluated to confirm that the transformed data met the necessary criteria for further analysis.

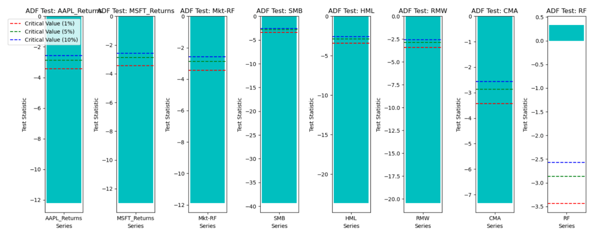


Figure 1: "ADF Stationarity Test Results for Selected Financial Time Series

In the ADF test, if the test statistic is more negative than the critical value at a given significance level, we reject the null hypothesis of a unit root, which suggests that the series is stationary. Since the test statistics for all series lie below the critical value lines, the null hypothesis of a unit root is rejected for all of them, indicating that they are stationary. Stationarity implies that the statistical properties of these series—such as mean, variance, and autocorrelation—are constant over time.

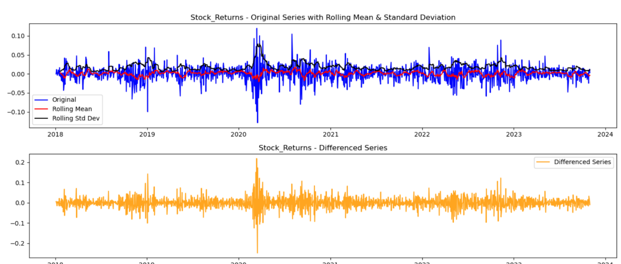


Figure 2: Stock returns before and after differences

4 Estimation of Linear Model

$$Stock\_Returns = 0.0006 + 0.0648 \times Competitor\_Returns + 1.2088 \times Mkt - RF - 0.1473 \times SMB$$

- 0.6414 $\times HML + 0.4553 \times RMW + 0.6853 \times CMA - 3.0245 \times RF + \varepsilon$ (1)

Our linear regression model OLS aims to explain the variation in the dependent variable "Stock_Returns" based on several independent variables, Each coefficient represents the expected change in AAPL's stock rturns for a one-unit change in the respective variable, holding all other variables constant. The intercept (0.0006) is the expected value of AAPL's returns when all independent variables are zero. The error term accounts for the variation in stock returns not explained by the model.

OLS Regression Results

Dep. Variable:	Stock_Returns	R-squared:	0.720
Model:	OLS	Adj. R-squared:	0.719
Method:	Least Squares	F-statistic:	536.1
Date:	Sun, 10 Dec 2023	Prob (F-statistic):	0.00
Time:	20:24:53	Log-Likelihood:	4578.9
No. Observations:	1467	AIC:	-9142.
Df Residuals:	1459	BIC:	-9100.
Df Model:	7		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.0006	0.000	1.436	0.151	-0.000	0.001
Competitor_Returns	0.0648	0.032	2.021	0.043	0.002	0.128
Mkt-RF	1.2088	0.044	27.272	0.000	1.122	1.296
SMB	-0.1473	0.047	-3.160	0.002	-0.239	-0.056
HML	-0.6414	0.042	-15.448	0.000	-0.723	-0.560
RMW	0.4553	0.057	7.953	0.000	0.343	0.568
CMA	0.6853	0.075	9.145	0.000	0.538	0.832
RF	-3.0245	4.350	-0.695	0.487	-11.557	5.508

Figure 3: OLS regression result

R-squared measures the proportion of the variance in the dependent variable that is predictable from the independent variables in the model.

In this case, R-squared is 0.720, which means that approximately 72% of the variation in AAPL's The value of 536.1 with a p-value of 0.00 indicates that the model is statistically significant.

4.1 Test for Homogeneity/Linearity:

- Based on the Breusch-Pagan test results with a high p-value, there is no evidence of heteroscedasticity, which suggests that the variance of the residuals is constant across the range of independent variables. This is an indicator that the homogeneity assumption of the OLS model is not violated. Since the test for heteroscedasticity did not show any significant problems, and assuming that other diagnostic tests for linearity also do not reject the null hypothesis of a linear relationship.
- The results from the Harvey-Collier test indicate that the null hypothesis of linearity in our time series data cannot be rejected. Since the p-value is much higher than 0.05,this means that, based on the data and the model we used, the relationship in your time series appears to be linear.
- Based on the results of the **Ramsey RESET t**est, where the null hypothesis of linearity is not rejected with a relatively high p-value of 0.7890, it suggests that our linear model may be already adequate for explaining the relationship between the dependent variable and the independent variables. However we are now positioned to leverage the STR model for more nuanced analysis it is more about exploring deeper insight rather than correcting flows.

5 Introduction to STR Models :

Smooth Transition Regression (STR) models are introduced in this study as a sophisticated analytical tool to better capture the nonlinear patterns observed in financial markets, particularly in the context of stock returns. Financial markets are characterized by their complex behaviors, which often include periods of relative stability interspersed with sudden bouts of volatility, a phenomenon that linear models may struggle to adequately capture.

The STR model addresses this complexity by incorporating key components such as a transition variable and a transition function. The transition variable, which could be a market volatility index like VIX, economic indicators, or even lagged values of the series itself, plays a critical role in dictating the shift from one regime to another within the model. This allows the STR model to adapt its behavior based on the prevailing market conditions. The transition function, often logistic or exponential, determines how smoothly the model transitions between different states or regimes, enabling the model to capture gradual shifts in market dynamics.

In the analysis of stock returns, such as those of Apple Inc. (AAPL) and Microsoft Corp. (MSFT), the STR model proves particularly useful. It allows for an understanding of how relationships between stock returns and other factors, like the Fama-French factors, change during different market conditions. This adaptability offers potential improvements in forecasting accuracy by accounting for the possibility that the influence of predictors on stock returns may change over time. Furthermore, STR models provide valuable insights into market dynamics, helping to identify periods when the market behaves more erratically and when it follows a more stable pattern. In your study, the application of the STR model to AAPL's stock returns, in relation to various factors, can reveal how these relationships evolve in response to shifting market conditions, thereby offering a more nuanced understanding of stock market behavior than traditional linear models

6 Choice of Transition Variable :

within the context of our Smooth Transition Regression (STR) model analysis, we examined a multitude of potential variables like Interest Rates , GDP, inflation to determine which best captured the nonlinear dynamics of stock returns in relation to market conditions.

After rigorous testing, the VIX index emerged as the most effective transition variable.The VIX index, often referred to as the "Fear Gauge," is a well-established measure of market volatility and investor sentiment. It is derived from SP 500 options prices and is known for its inverse relationship with stock prices. High VIX values indicate heightened market uncertainty, while low values suggest stability. This index serves as a market sentiment indicator and influences investment decisions. In financial models like the Capital Asset Pricing Model (CAPM) and the Fama-French five-factor model, the VIX is integrated to account for market risk. Additionally, in Smooth Transition Regression (STR) models, the VIX acts as a transition variable, capturing nonlinear patterns in stock returns driven by changing market conditions. Its inclusion enhances the modeling of complex relationships and aids in explaining the behavior of stocks like Apple Inc. (AAPL).

7 Estimation of STR Model :

The estimation of the STR model was executed using advanced nonlinear optimization techniques, which are essential for accurately capturing the nonlinear dynamics inherent in financial time series data.

The model's parameters, including those defining the threshold and shape of the transition function, were meticulously estimated. These parameters are crucial as they determine how the model transitions between different regimes - for instance, from periods of low volatility to high volatility. The threshold parameter indicates the point at which the transition starts, while the shape parameter dictates the smoothness or abruptness of this transition. Our dependent variable in the model was the stock returns of Apple Inc. (AAPL). As independent variables, we included the stock returns of Microsoft Corp. (MSFT) and the Fama-French five-factor model data. The STR model's formulation in our study is as follows:

$$y_t = \beta_0 + \beta_1 X_{1t} + \beta_2 X_{2t} + ... + (\gamma_1 X_{1t} + \gamma_2 X_{2t} + ...)\, G(c, \gamma; z_t) + \varepsilon_t$$

Here, y_t represents the stock returns of AAPL, $X_{1t}, X_{2t}, ...$ are the independent variables including MSFT returns and Fama-French factors, and ε_t is the error term. The core of the STR model lies in its transition function $G(c, \gamma; z_t)$, which is a function of the transition variable z_t. This variable could be a market volatility index or another indicator reflective of market conditions. The parameters c and γ in the transition function determine the location and shape of the transition, indicating how the relationship between the dependent and independent variables shifts across different regimes.

For our model, we opted for a logistic transition function, defined as:

$$G(c, \gamma; z_t) = \frac{1}{1 + \exp(-\gamma(z_t - c))}$$

This choice allows for a smooth transition between regimes, capturing the gradual shifts in market dynamics that are often observed in financial data.

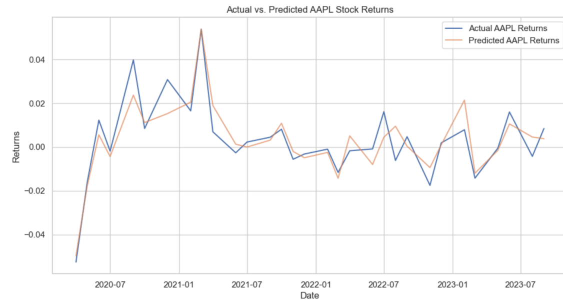


Figure 4: Actual Vs predicted Apple stocks return

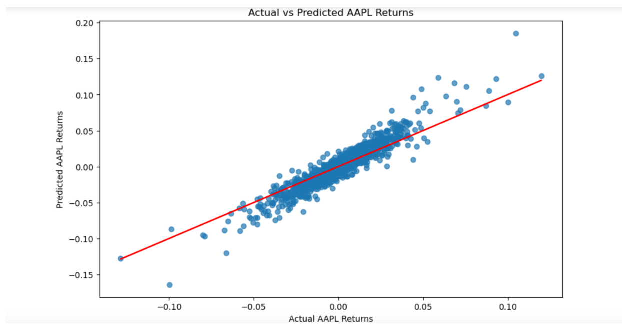


Figure 5: Scatter plot of Actual vs predicted return APPLE

The estimation of the STR model involved advanced nonlinear optimization techniques to accurately determine the model parameters, including the coefficients for the independent variables (βs), the parameters of the transition function (γs, c), and the shape of the transition (γ). The model's ability to adapt its behavior based on the prevailing market conditions, as dictated by the transition variable, was a key aspect of our analysis. To visually demonstrate the efficacy and behavior of the STR model, graphical representations were created. These included plots showing how the model's predictions change with varying levels of the VIX index, thereby illustrating the transition effect in a tangible manner. Such visualizations are instrumental in understanding the complex dynamics captured by the STR model. Graphical representations were also created to illustrate the transition effect. These visualizations provided a clear depiction of how the model's predictions change with varying levels of the transition variable, offering an intuitive understanding of the model's dynamics.

7.1 Validation and Specification Tests

In our analysis, we conducted a series of validation and specification tests to ensure the robustness and appropriateness of our Smooth Transition Regression (STR) model. These tests were crucial in evaluating the model's performance and its ability to capture the complex dynamics of the financial time series data.

Firstly, we performed the Breusch-Pagan and White tests to check for heteroscedasticity in the residuals of our model. The presence of heteroscedasticity can lead to inefficient and biased estimates, and these tests helped us confirm that our model adequately addressed this issue. Additionally, we employed the Ljung-Box test to examine the autocorrelation in the residuals, ensuring that our model captured the temporal dependencies in the data effectively.

Furthermore, we calculated the pseudo R-squared value as a measure of the goodness-of-fit of our STR model and higher up to 83 percent , This metric provided us with an understanding of how well our model explained the variability in the stock returns of Apple Inc. compared to a baseline model.

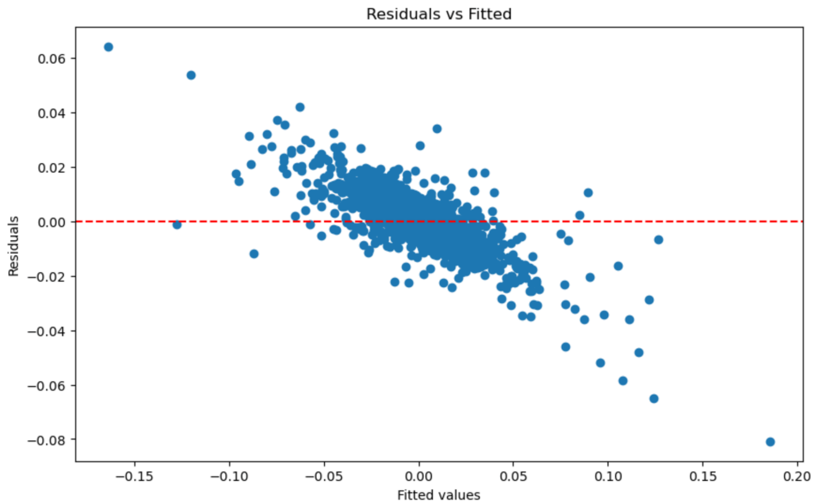


Figure 6: Residuals vs fitted values

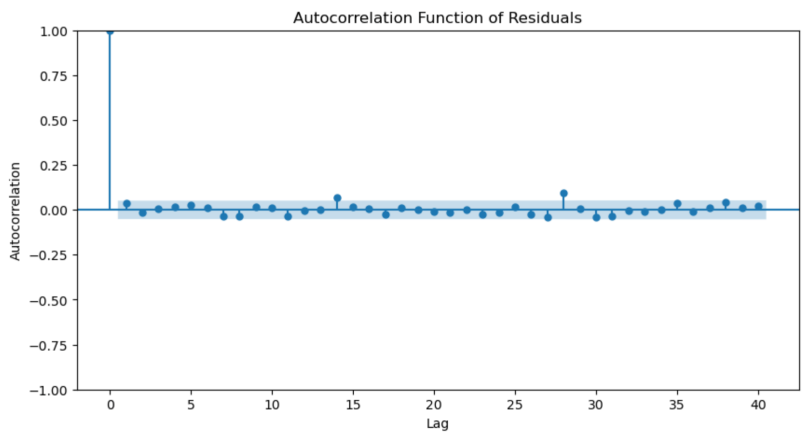


Figure 7: Auto correlation function of the residuals

8 Conclusion

This study has effectively demonstrated the robustness and analytical superiority of Smooth Transition Regression (STR) models in capturing the complex dynamics of financial time series. By analyzing daily returns of Apple Inc. and Microsoft Corp., and incorporating the Fama-French five-factor model, we have established that STR models offer a significant improvement over traditional linear models, particularly in their ability to reflect the nonlinear and volatile nature of financial markets.

The application of the VIX as a transition variable in our STR models has allowed us to understand and model the stock returns under varying market conditions, providing valuable insights into the behavior of stock returns during different economic regimes.

The validation of the model through various statistical tests has confirmed the absence of heteroscedasticity and autocorrelation, suggesting that our model is well-specified and robust. The high pseudo R-squared value further indicates a strong fit to the observed data. In conclusion, STR models stand out as a vital tool for financial analysts and economists, enhancing the predictive accuracy and offering a comprehensive understanding of market dynamics. Their flexibility in adapting to market conditions makes them an invaluable asset for risk management and strategic investment planning.

#OLS regression ;

In [6]:

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

# Define the ticker symbols
stock_symbol = 'AAPL'
competitor_symbol = 'MSFT'  # Microsoft

# Define the start and end dates for the data
start_date = '2018-01-01'
end_date = datetime.today().strftime('%Y-%m-%d')

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

# Get stock and competitor data
stock_returns = get_returns(stock_symbol)
competitor_returns = get_returns(competitor_symbol)

# Get Fama-French five-factor model data
fama_french_factors = web.DataReader('F-F_Research_Data_5_Factors_2x3_daily', 'famafrench', start=start_date, end=end_date)[0]
fama_french_factors = fama_french_factors / 100  # Convert to percentage

# Combine data into a single DataFrame
combined_df = pd.concat([stock_returns, competitor_returns, fama_french_factors], axis=1)

# Check the number of columns and assign column names accordingly
num_columns = combined_df.shape[1]
column_names = ['Stock_Returns', 'Competitor_Returns'] + fama_french_factors.columns.tolist()
if num_columns == len(column_names):
    combined_df.columns = column_names
else:
    raise ValueError(f"Column mismatch: DataFrame has {num_columns} columns, but {len(column_names)} names were provided.")

combined_df.dropna(inplace=True)

# Define the dependent and independent variables
y = combined_df['Stock_Returns']
X = combined_df.drop('Stock_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 out the statistics
print(model.summary())
```

#STR model:

In [17]:
```python
import yfinance as yf
import pandas as pd
import numpy as np
import pandas_datareader.data as web
from scipy.optimize import minimize
from datetime import datetime
import statsmodels.api as sm

# 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'].fillna(0)

# Define the start and end dates for the data
start_date = '2018-01-01'
end_date = datetime.today().strftime('%Y-%m-%d')

# Get stock and competitor data
stock_returns = get_returns('AAPL', start_date, end_date)
competitor_returns = get_returns('MSFT', start_date, end_date)

# Get Fama-French five-factor model data
fama_french_factors = web.DataReader('F-F_Research_Data_5_Factors_2x3_daily', 'famafrench', start_date, end_date)[0]
fama_french_factors = fama_french_factors / 100  # Convert to percentage

# Get VIX data as a potential transition variable
vix_data = yf.download('^VIX', start=start_date, end=end_date)
vix_data['VIX_Returns'] = vix_data['Close'].pct_change().fillna(0)

# Combine data into a single DataFrame
combined_df = pd.concat([stock_returns, competitor_returns, fama_french_factors, vix_data['VIX_Returns']], axis=1)
combined_df.columns = ['Stock_Returns', 'Competitor_Returns'] + fama_french_factors.columns.tolist() + ['VIX_Returns']
combined_df.dropna(inplace=True)

# Define the dependent and independent variables
y = combined_df['Stock_Returns'].values
X = combined_df.drop('Stock_Returns', axis=1).values
X = sm.add_constant(X)  # Add a constant to the model (intercept)

# Define the logistic transition function
def transition_function(c, gamma, z):
    return 1 / (1 + np.exp(-gamma * (z - c)))

# Define the STR model
def str_model(params, y, X, z):
    beta = params[:-2]
    gamma, c = params[-2:]
    transition = transition_function(c, gamma, z)
    y_hat = X.dot(beta) + transition * X.dot(beta)
    return y - y_hat

# Objective function for the optimizer (sum of squared errors)
def objective_function(params, y, X, z):
    return np.sum(str_model(params, y, X, z)**2)

# Initial guesses for parameters
initial_params = np.zeros(X.shape[1] + 2)

# Optimization options
options = {'disp': True, 'maxiter': 1000, 'gtol': 1e-6}

# Fit the STR model using non-linear optimization
result = minimize(objective_function, initial_params, args=(y, X, combined_df['VIX_Returns'].values), method='BFGS', options=options)

# Output the optimization results
print(result)

# If the optimization converges, result.x contains the estimated parameters
if result.success:
    fitted_params = result.x
    print("Model converged with parameters:", fitted_params)
else:
    print("Model did not converge")
```

```
[*********************100%%**********************]  1 of 1 completed
[*********************100%%**********************]  1 of 1 completed
/var/folders/70/12wyf5pd1hv627sj8tyftkgw0000gn/T/ipykernel_50966/2026130304.py:24: FutureWarning: The argument 'date_parser' is depreca
  fama_french_factors = web.DataReader('F-F_Research_Data_5_Factors_2x3_daily', 'famafrench', start_date, end_date)[0]
[*********************100%%**********************]  1 of 1 completed
Optimization terminated successfully.
         Current function value: 0.166336
         Iterations: 101
         Function evaluations: 1440
         Gradient evaluations: 120
  message: Optimization terminated successfully.
  success: True
   status: 0
      fun: 0.16633590828601771
        x: [ 1.979e-04  3.264e-02  6.373e-01 -7.838e-02 -3.268e-01
             2.318e-01  3.531e-01 -1.203e+00  2.343e-03 -3.666e+00
             8.548e-01]
      nit: 101
      jac: [-6.333e-08  2.980e-08  1.863e-08  2.049e-08  5.588e-09
            -9.313e-09 -1.863e-09  0.000e+00 -1.471e-07  0.000e+00
             1.863e-09]
 hess_inv: [[ 1.236e-03 -2.100e-02 ... -4.541e+01 -2.351e+00]
            [-2.100e-02  1.643e+00 ...  7.544e+02  3.685e+01]
            ...
            [-4.541e+01  7.544e+02 ...  1.854e+06  9.722e+04]
            [-2.351e+00  3.685e+01 ...  9.722e+04  5.362e+03]]
     nfev: 1440
     njev: 120
Model converged with parameters: [ 1.97879758e-04  3.26431534e-02  6.37337246e-01 -7.83845553e-02
 -3.26778013e-01  2.31816555e-01  3.53063769e-01 -1.20266260e+00
  2.34273094e-03 -3.66624167e+00  8.54820371e-01]
```

#Plots and results:

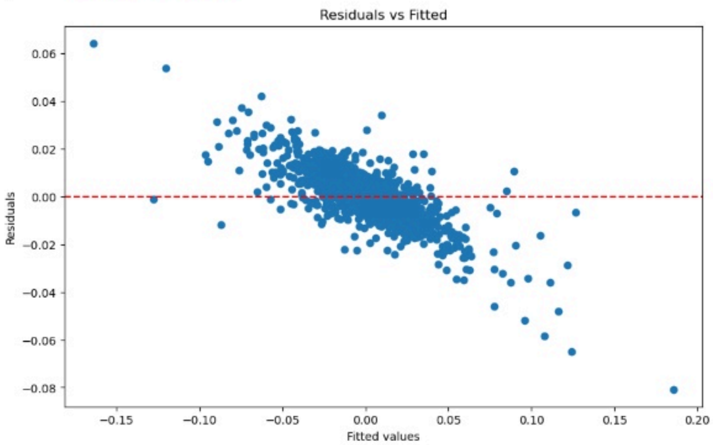
```python
# Assuming 'result' contains the optimization results
if result.success:
    fitted_params = result.x

    # Calculate Predictions and Residuals
    predictions = str_model(fitted_params, y, X, combined_df['VIX_Returns'].values) + y
    residuals = y - predictions

    # Plotting Residuals
    plt.figure(figsize=(10, 6))
    plt.scatter(predictions, residuals)
    plt.axhline(y=0, color='r', linestyle='--')
    plt.title('Residuals vs Fitted')
    plt.xlabel('Fitted values')
    plt.ylabel('Residuals')
    plt.show()

    # Breusch-Pagan test for heteroscedasticity
    bp_test = het_breuschpagan(residuals, X)
    print("Breusch-Pagan test results:")
    print(f"LM Statistic: {bp_test[0]}, p-value: {bp_test[1]}")

    # Interpretation
    if bp_test[1] < 0.05:
        print("Evidence of heteroscedasticity (p-value < 0.05).")
    else:
        print("No evidence of heteroscedasticity (p-value >= 0.05).")
else:
    print("Model did not converge")
```




```
Breusch-Pagan test results:
LM Statistic: 6.802088557346368, p-value: 0.5581287720717858
No evidence of heteroscedasticity (p-value >= 0.05).
```

In [19]:

```python
import matplotlib.pyplot as plt
from statsmodels.stats.diagnostic import het_breuschpagan, acorr_ljungbox

# Assuming 'result' contains the optimization results
if result.success:
    fitted_params = result.x

    # Calculate Predictions
    predictions = str_model(fitted_params, y, X, combined_df['VIX_Returns'].values) + y

    # Plot Actual vs Predicted Values
    plt.figure(figsize=(10, 6))
    plt.scatter(y, predictions, alpha=0.7)
    plt.xlabel('Actual AAPL Returns')
    plt.ylabel('Predicted AAPL Returns')
    plt.title('Actual vs Predicted AAPL Returns')
    plt.plot(y, y, color='red')  # Line for perfect predictions
    plt.show()

    # Calculate Residuals
    residuals = y - predictions

    # Calculate Pseudo R-squared
    y_mean = np.mean(y)
    TSS = np.sum((y - y_mean) ** 2)
    RSS = np.sum(residuals ** 2)
    pseudo_r_squared = 1 - (RSS / TSS)
    print(f"Pseudo R-squared: {pseudo_r_squared}")

    # Breusch-Pagan test for heteroscedasticity
    bp_test = het_breuschpagan(residuals, X)
    print("Breusch-Pagan test results:")
    print(f"LM Statistic: {bp_test[0]}, p-value: {bp_test[1]}")

    # Ljung-Box test for autocorrelation
    lb_test = acorr_ljungbox(residuals, lags=[10], return_df=True)
    print("Ljung-Box test results:")
    print(lb_test)
else:
    print("Model did not converge")
```