Hasbro is CLOSED: Forecasting Stock Prices with ARMA-GARCH

Benny Anteneh, Ryan Baker, Riley Clark, Ethan Durham, Hailey Ho, Yaqi Lin

5 December 2025

Introduction

In this paper, we found a forecast of the closing prices of Hasbro stocks using an ARMA-GARCH model. Hasbro is a toy manufacturing and entertainment holding company, and owns smaller brands like G.I. Joe, My Little Pony, Nerf, Play-Doh, Transformers, and Wizards of the Coast. For our forecasting purposes, we studied the closing prices from January 2020 to the end of December 2024. The time series information was drawn from Yahoo Finance. We found that an ARMA(1,1)-GARCH(1,1) model performed well when forecasting across a 30-day period into the future as well as backtesting a 50-day period. Our final prediction declared the January 29, 2025 Hasbro Closing price would be roughly $56, which is 96% accurate compared to the actual price of about $58. Overall, we've constructed a good and useful time series model.

We used the following R-packages: forecast, ggplot2, quantmod, rugarch, tseries.

Data Description

Data resources

We downloaded daily Hasbro (HAS) stock prices from Yahoo Finance using the getSymbols() function in the quantmod package. The dataset covers trading days from January 1, 2020 through December 31, 2024, giving us a complete five-year window of observations suitable for time-series modeling.

Data structure and variable

The data contain one key numeric variable, the daily closing price of HAS stock, recorded for each trading day. The date serves as the index of the series, and each row corresponds to a single trading day between 2020 and 2024. A preview of the dataset confirms that the values are organized chronologically with the expected price information for each date.

Column Name	Type	Description
Date	Date	Represents each trading day from 2020 to 2024.
Close	Numeric	The daily closing price of HAS stock in USD.

Preview of the first few rows :

```None
           Close
2020-01-02 105.33
2020-01-03 105.05
2020-01-06 105.36
2020-01-07 103.39
2020-01-08 104.95
2020-01-09 103.90
```

Data Size

The dataset contains over 1,200 observations, corresponding to all trading days within the 2020–2024 period. After applying na.omit(), no missing values were found.

```
Summary Statistics of Hasbro Closing Prices:
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  42.88   61.02   73.30   75.24   92.24  105.78 

Standard Deviation: 16.93337
```

These values provide a general overview of the distribution of closing prices and indicate substantial variation over the five-year window.

Data Visualization

The following plot shows the movement of HAS closing prices from 2020 to 2024. The series exhibits noticeable fluctuations and changes in volatility over time, which are typical characteristics of financial time-series data and motivate the use of log returns and ARMA-GARCH methods in later sections. Although the time series contains several large movements, these represent normal market variation and are retained for the analysis. No influential points were removed.

As an aside, spikes in the beginning of 2020 line up with the beginning of the Covid pandemic and the real world economic results. In particular, the increase in both the consumption of goods and the boom in investments in the U.S. economy at that time.

Analysis & Evaluation

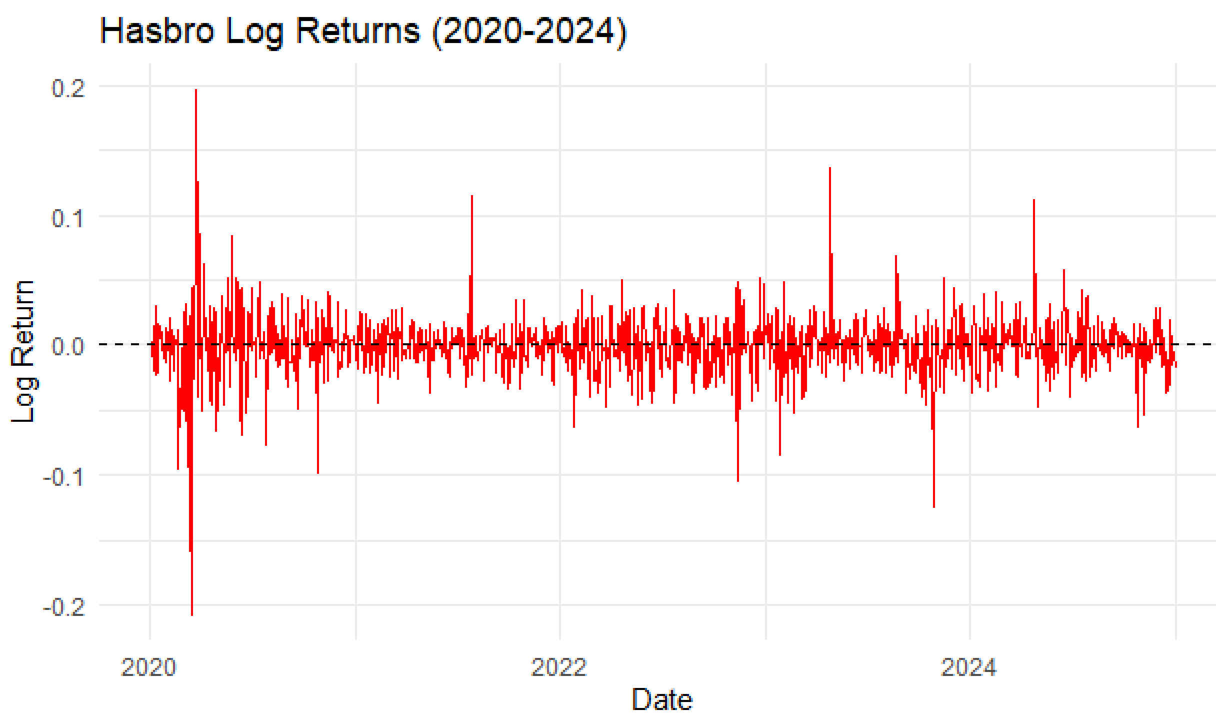
Stationarity

The closing prices are tested with the Augmented Dickey-Fuller test. The resulting p-value is 0.3162, thus the null hypothesis cannot be rejected and so this test concludes it is non-stationary. The KPSS

Test is also used to check for non-stationarity. The test returns a p-value of 0.01. This rejects the null hypothesis that the data is stationary. Together these tests show Hasbro's closing prices are non-stationary.

```
None
adf_test_price <- da adf.test(as.numeric(has))

kpss_test_price <- kpss.test(as.numeric(has))
```

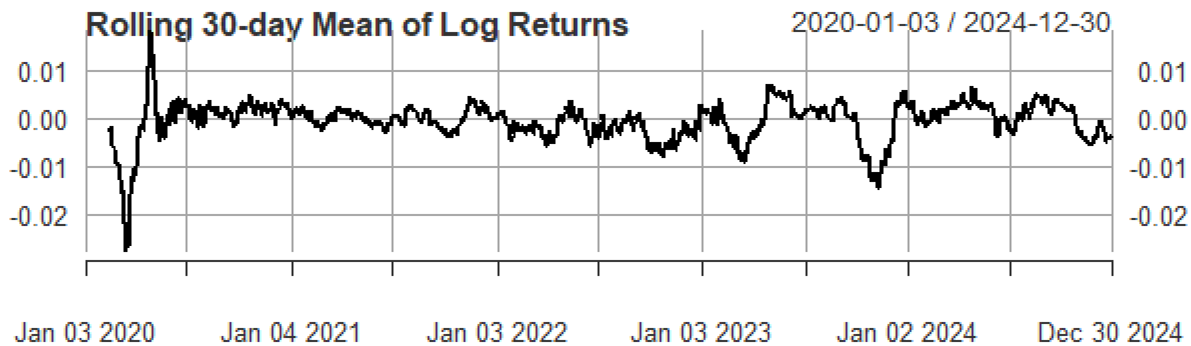
Log Returns for Stationarity

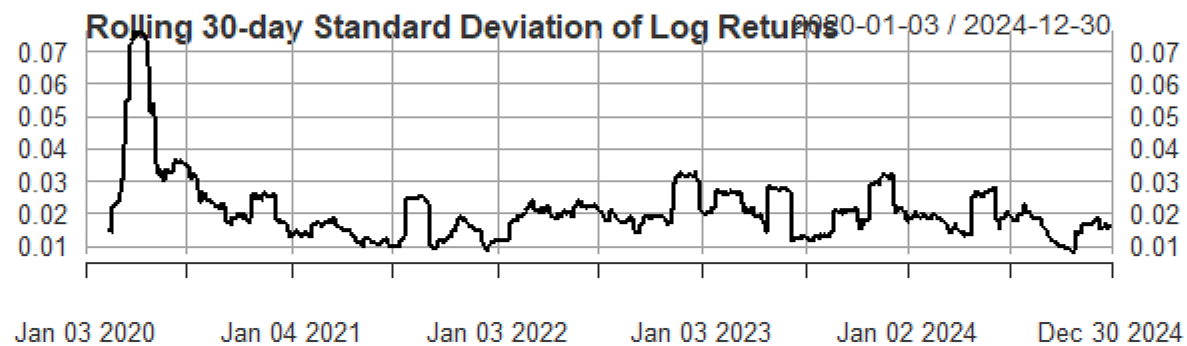


In order to create a data set that is more stationary, we will apply log difference. Applying the log difference method and plotting the log returns immediately shows a more centered data set. To ensure that the new data is now stationary the Augmented Dickey-Fuller Test and KPSS Test are used again.

For the Dickey-Fuller Test, the p-value is 0.01 meaning we reject the null hypothesis and this test concludes the data is stationary. For the KPSS Test, the p-value is still 0.1 and so the null hypothesis cannot be rejected. The log returns are thus deemed more stationary by the aforementioned tests. The degree to how stationary the data is can be further analyzed with rolling statistics.

```
None
adf_test_returns <- adf.test(as.numeric(log_returns))

kpss_test_returns <- kpss.test(as.numeric(log_returns))
```

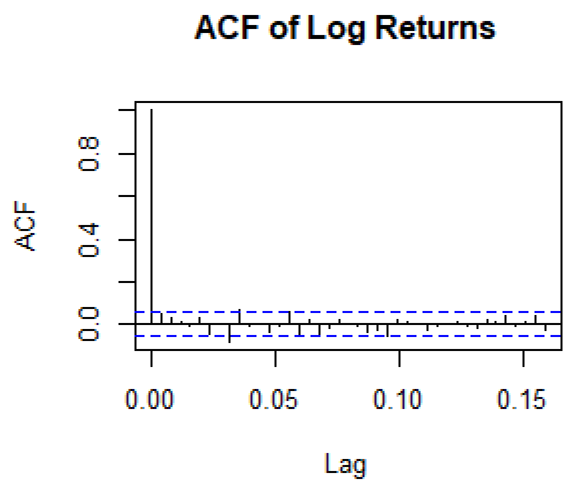


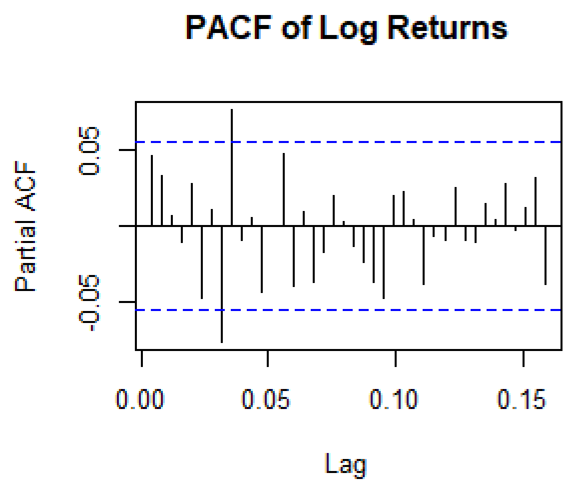


To reinforce the assumption that the data set is stationary, we plot a 30-day rolling statistic of the data. The rolling mean fluctuates about 0 without any sort of long term trend, and the rolling standard deviation is fairly stable after the initial spike in 2020. The constant overall mean and the roughly stable standard deviation suggests that this stationarity is actually weak. Although the conditions for stationary are satisfied, the degree to which they are is not necessarily strong.

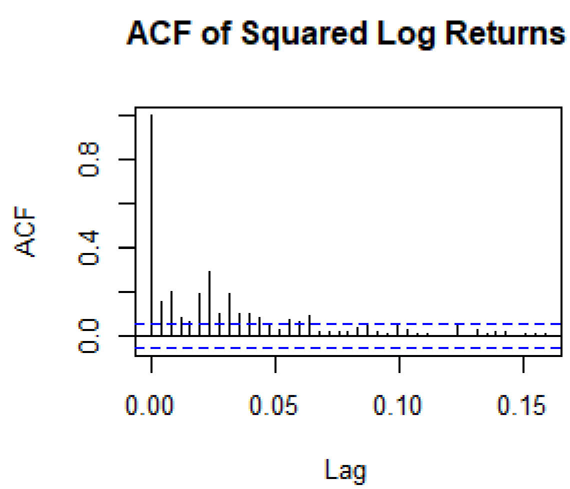
Autocorrelation

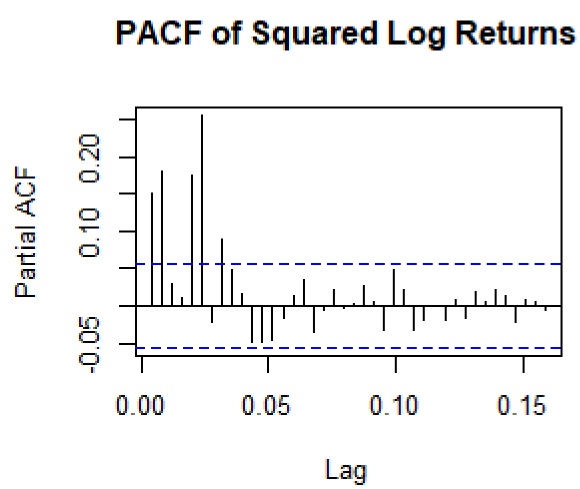
Once the log returns were confirmed to be stationary, the next step was to examine whether any meaningful autocorrelation remained in the series. To do this, the log returns were converted into a time-series object with a trading-year frequency of 252, which allows us to apply standard ARMA identification tools. We then plotted the ACF and PACF of the log returns. Aside from a single small spike at lag 1, both plots stay within the significance bands, which suggests that the mean of the return series behaves almost like white noise. This is typical for financial assets, where past returns offer little direct predictive power.





To check for volatility clustering, the idea that large shocks tend to be followed by more large shocks, we repeated the process for the squared log returns. In contrast to raw returns, the squared return ACF decays more gradually, and the PACF shows several early spikes. This behavior is a classic signal of heteroskedasticity and indicates that a GARCH-type model is appropriate for capturing changing variance over time.





ARMA-GARCH Model Selection

To support the identification step, we also used auto.arima() to automatically select the ARMA structure for the mean equation. As expected from the ACF and PACF, the algorithm settled on a low-order ARMA model (3,0,2), confirming that only minimal structure exists in the mean.

```
None
auto_model <- auto.arima(returns_ts, seasonal = FALSE, stepwise = FALSE,
approximation = FALSE)
```

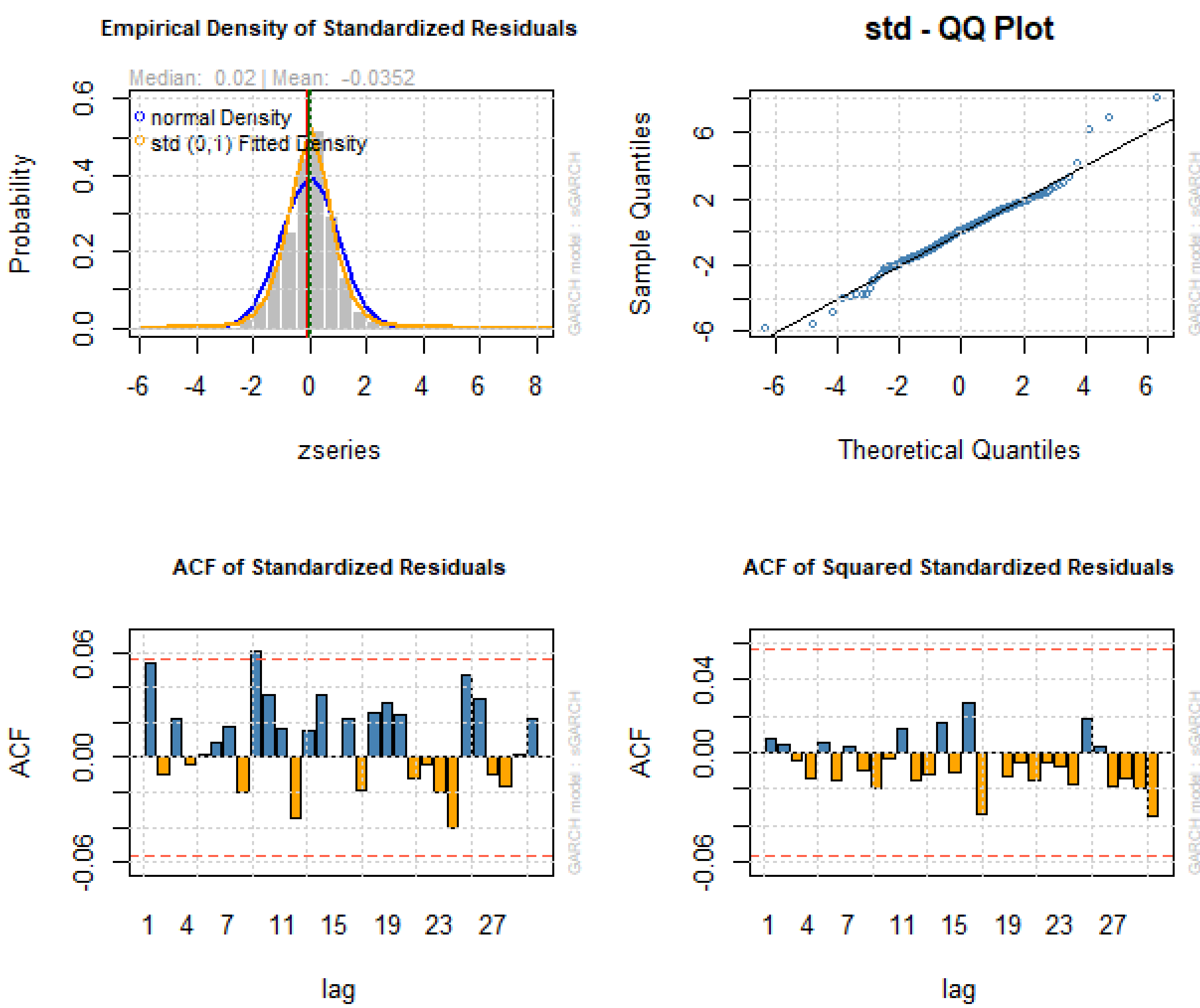
Based on these results, we selected an ARMA(1,1)-GARCH(1,1) model. This specification is widely used in financial econometrics because it captures both the slight autocorrelation in returns and the persistent volatility patterns seen in the squared-return plots. We also used a Student-t distribution for the innovations to account for the heavy tails that are common in financial data.

To confirm that this was the best choice among other options, we compared the AIC and BIC of this model to similar ones, primarily changing the ARMA parameters to (3,2), (2,2), and (1,2) due to our previous automatic ARMA selection. In every instance, the ARMA(1,1)-GARCH(1,1) model had a better Akaike and Bayes criterion (just over -5 for each).

```
None
spec <- ugarchspec(
  variance.model = list(model = "sGARCH", garchOrder = c(1, 1)),
  mean.model = list(armaOrder = c(1, 1), include.mean = TRUE),
  distribution.model = "std"
)

fit <- ugarchfit(spec = spec, data = returns_ts, out.sample = 50)
```

Once the model is estimated, we checked the residual diagnostics to confirm that it captured the autocorrelation and volatility dynamics appropriately. The ACF of the standardized residuals showed no significant spikes, and the squared-residual ACF similarly stayed within the bounds. This indicates that the model removed both linear dependence and any remaining ARMA effects. The QQ-plot also showed that the Student-t distribution was a reasonable fit, with only minor deviations in the tails.

Final Model



Taken together, these results show that the ARMA(1,1)-GARCH(1,1) model provides a strong fit for the HAS return series. The mean equation is appropriately simple, and the GARCH component captures the volatility clustering evident in the data. With these diagnostics satisfied, the model is reliable enough to use for forecasting in the next section.

Model Forecasting

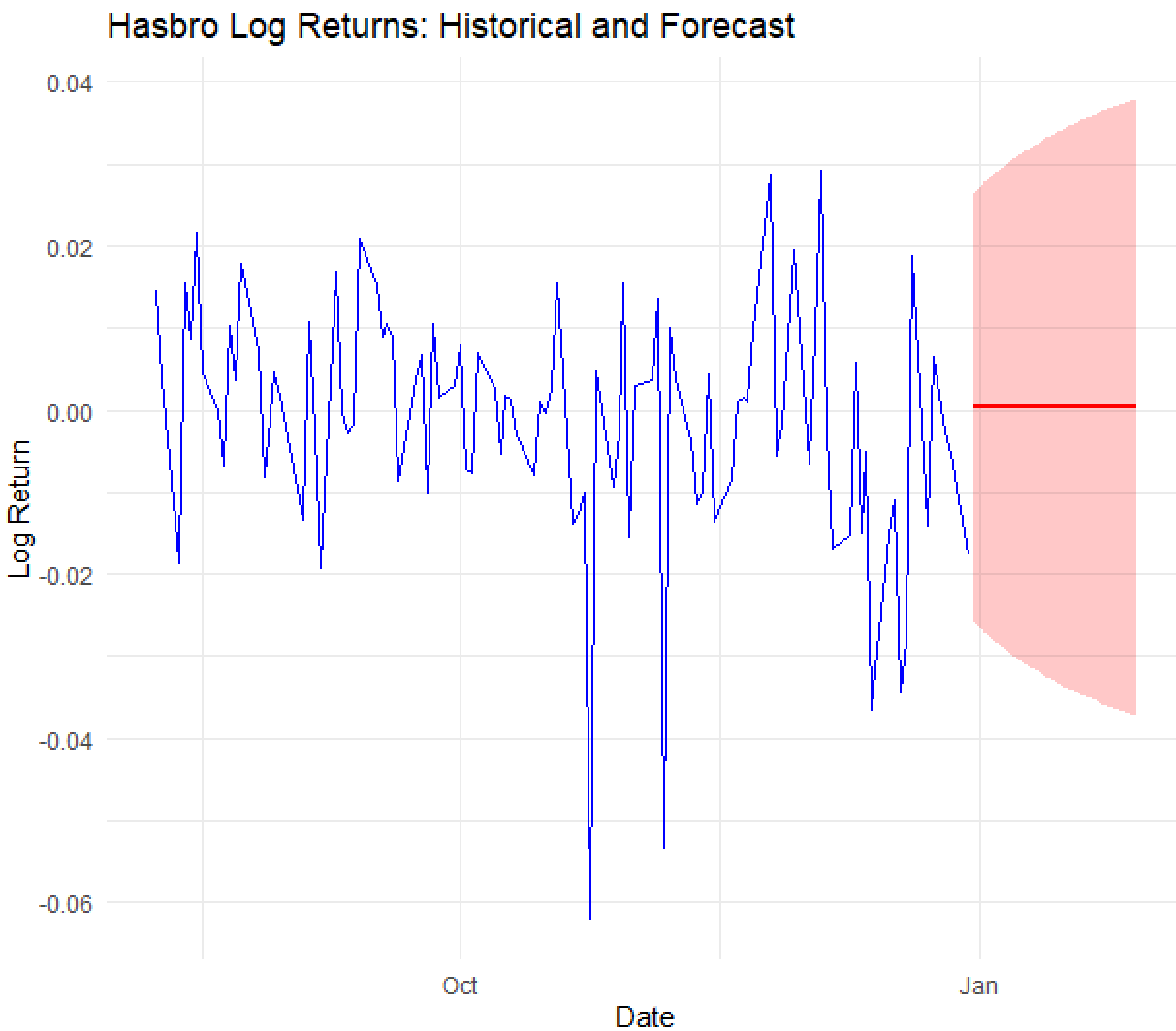
Forecast Log Returns

This section forecasts the hasbro log returns over the next 30 days using our final ARMA-GARCH model.

```
None
forecast_horizon <- 30
forecast_result <- ugarchforecast(fit, n.ahead = forecast_horizon, n.roll = 0)

forecast_returns <- fitted(forecast_result)
forecast_variance <- sigma(forecast_result)^2

forecast_dates <- seq.Date(from = as.Date(tail(index(log_returns), 1)) + 1,
                           by = "day", length.out = forecast_horizon)
forecast_df <- data.frame(
  Date = forecast_dates,
  LogReturn_Forecast = as.numeric(forecast_returns),
  Variance_Forecast = as.numeric(forecast_variance),
  Lower_CI = as.numeric(forecast_returns) - 1.96 *
sqrt(as.numeric(forecast_variance)),
  Upper_CI = as.numeric(forecast_returns) + 1.96 *
sqrt(as.numeric(forecast_variance))
)
```

Plot of Forecasted Log Returns

The plot shown below represents the historical log returns of Hasbro as well as a forecast. The blue line represents the historical returns, while the red line represents the 30 day forecast. The shaded red region is the confidence interval of the prediction. This model is created from the data made by the ARMA-GARCH model.



Convert Forecasted Log Returns to Price Forecast

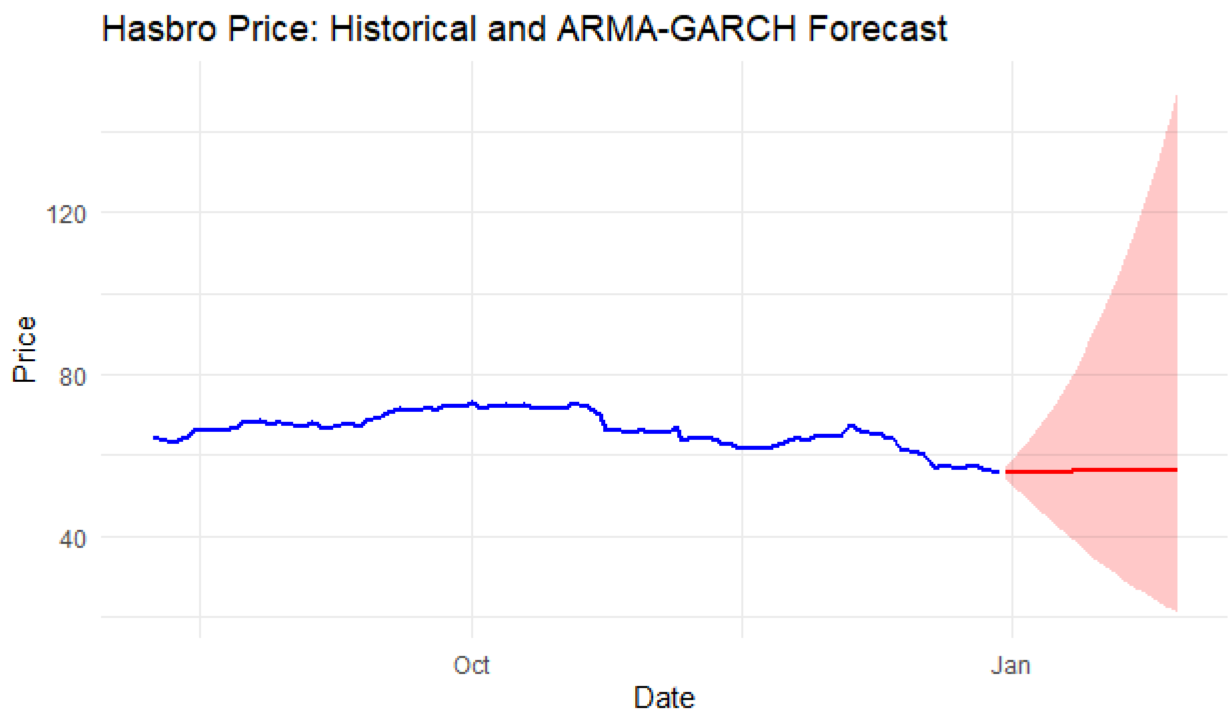
The forecast model of the log returns is then converted to an actual stock price forecast.

```
None
last_price <- as.numeric(tail(has, 1))
price_forecast <- last_price * exp(cumsum(forecast_df$LogReturn_Forecast))

price_lower <- last_price * exp(cumsum(forecast_df$Lower_CI))
price_upper <- last_price * exp(cumsum(forecast_df$Upper_CI))

final_forecast <- data.frame(
  Date = forecast_dates,
```

```
  Price_Forecast = price_forecast,
  Price_Lower = price_lower,
  Price_Upper = price_upper
)
```

Final Price Forecast Plot

The historical stock price and 30 day forecast are graphed below. The blue line represents the historical price from the last 100 days, while the red line is the forecast. The shaded red region represents the confidence interval.

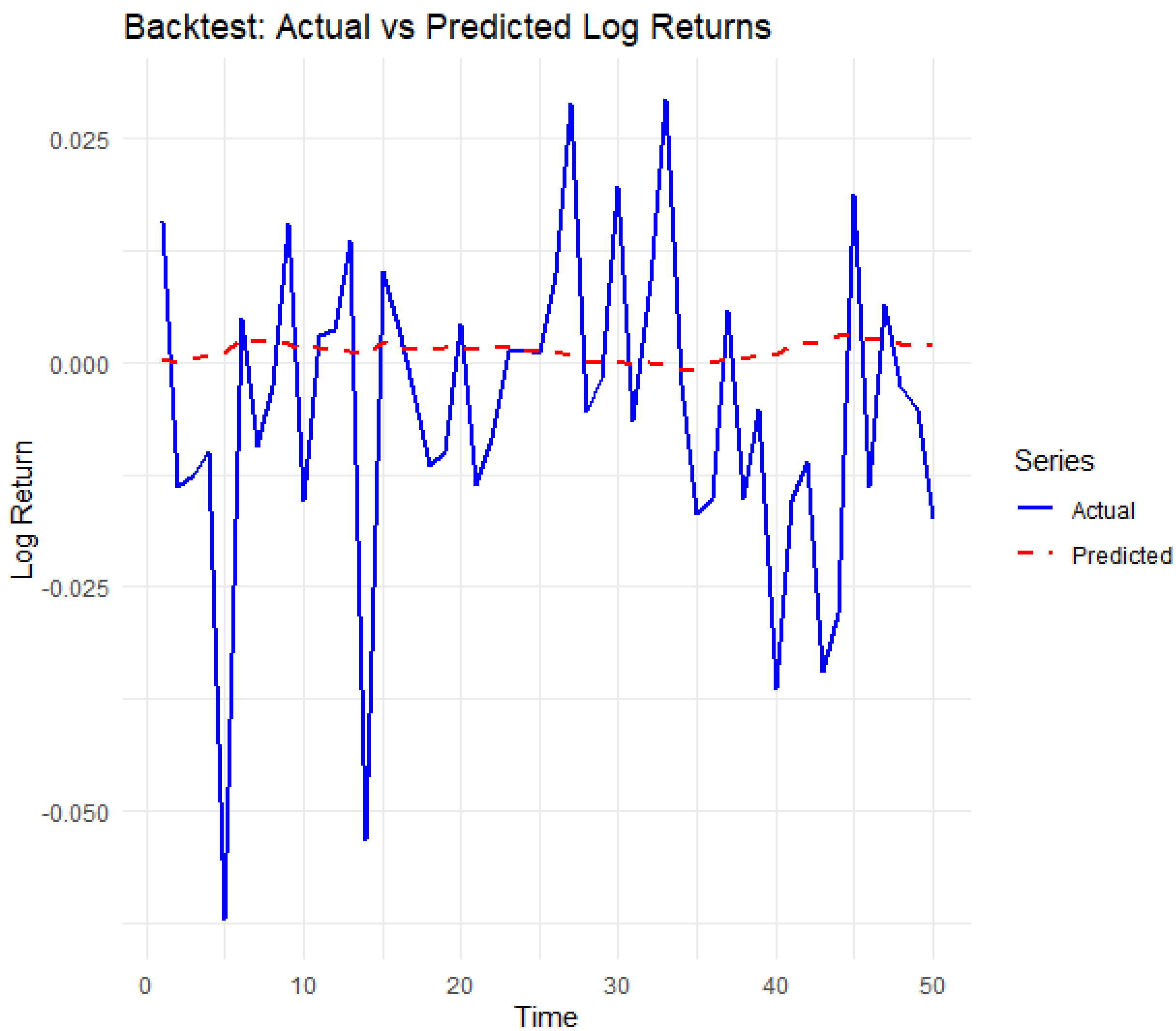


Final Forecast Results

We can now examine the summary of the final price forecasts across January 2025. The predictions are fairly low in variance, ranging from 55.73 up to 56.33 across the entire 30-day period. There are no fluctuations, though the lower and upper bounds increase greatly as time goes on.

```None
Final Price Forecast Summary:
         Date Price_Forecast Price_Lower Price_Upper
1  2024-12-31       55.73337    54.30130    57.20320
2  2025-01-01       55.75618    52.89168    58.77581
3  2025-01-02       55.77853    51.48492    60.43020
4  2025-01-03       55.80048    50.08441    62.16893
5  2025-01-04       55.82212    48.69316    63.99481
6  2025-01-05       55.84349    47.31385    65.91084
7  2025-01-06       55.86464    45.94887    67.92023
8  2025-01-07       55.88560    44.60030    70.02644
9  2025-01-08       55.90641    43.27000    72.23311
10 2025-01-09       55.92709    41.95959    74.54410
11 2025-01-10       55.94767    40.67046    76.96351
12 2025-01-11       55.96816    39.40385    79.49566
13 2025-01-12       55.98858    38.16077    82.14511
14 2025-01-13       56.00894    36.94212    84.91666
15 2025-01-14       56.02925    35.74861    87.81536
16 2025-01-15       56.04953    34.58085    90.84652
17 2025-01-16       56.06978    33.43930    94.01573
18 2025-01-17       56.09000    32.32431    97.32883
19 2025-01-18       56.11020    31.23616   100.79198
20 2025-01-19       56.13038    30.17499   104.41164
21 2025-01-20       56.15056    29.14089   108.19456
22 2025-01-21       56.17072    28.13385   112.14784
23 2025-01-22       56.19089    27.15381   116.27891
24 2025-01-23       56.21104    26.20064   120.59557
25 2025-01-24       56.23120    25.27416   125.10597
26 2025-01-25       56.25135    24.37411   129.81867
27 2025-01-26       56.27151    23.50023   134.74261
28 2025-01-27       56.29167    22.65220   139.88717
29 2025-01-28       56.31183    21.82965   145.26218
30 2025-01-29       56.33199    21.03219   150.87791
```

Model Validation



Backtesting lets us review the results of our model across the last 50 days of Closing prices. We can measure accuracy and error this way. The results of the backtesting can be summarized as follows:

```None
Last actual price: 55.71
Forecasted price in 30 days: 56.33199
Expected return: 1.116484 %

Backtesting Results (Last 50 days):
Mean Absolute Error (MAE): 0.01343775
Root Mean Square Error (RMSE): 0.01858904
```

```
Mean Absolute Percentage Error (MAPE): 96.56453 %
```

We can see that the last actual Close price was 55.71, and the 30-day forecast is 56.33. If we compare this to the actual recorded Close price on January 29, 2025, we find that the actual price was 58.55, which is about 96.21% accurate. This falls in line with our Mean Absolute Percentage Error above, 96.56%, and indicates that our forecasting model is very good.

```
None
backtest_forecasts <- ugarchforecast(fit, n.ahead = 1, n.roll = 49)

backtest_fitted <- as.numeric(fitted(backtest_forecasts))
actual_values <- as.numeric(tail(returns_ts, 50))

forecast_errors <- actual_values - backtest_fitted
```

Conclusion

The ARMA-GARCH model creates a strong analysis of the Hasbro stocks from January 2020 to December 2024. Overall, the stock prices had a lot of volatility clustering. However, these are natural market tendencies, and especially given COVID, these numbers are not abnormal. Overall, no outlying data was removed.

The first tests performed on the dataset showed that the data was non-stationary. Thus, we applied the log difference method to create a more centralized dataset which passed the Augmented Dickey-Fuller test for stationarity, but not the KPSS test. The next tests performed showed that the dataset had slight correlation and persistent volatility, so the ARMA-GARCH model would be the strongest fit.

Using an ARMA(1,1)-GARCH(1,1) model, it was predicted that the expected return in the next 30 days would be around 1.116% with a stock price of $56.33. The backtest showed that the predicted log returns were much less volatile than the actual log returns, however, the numerical difference is not significant. When compared to the actual Close price in January, our forecast was found to be 96% accurate, which is exactly the accuracy predicted by backtesting. This suggests that the model we've created is a good one.

References

Ghani, I M Md, and H A Rahim. *Modeling and Forecasting of Volatility Using ARMA-GARCH: Case Study on Malaysia Natural Rubber Prices*, 2019, www.researchgate.net/publication/335431614_Modeling_and_Forecasting_of_Volatility_using_ARMA-GARCH_Case_Study_on_Malaysia_Natural_Rubber_Prices.

Hasan, Mahmud. "Micrsoft Stock Price Analysis Using GARCH Model in R." *RPubs*, RStudio, 6 Jan. 2021, rpubs.com/Mahmud_Hasan/778532.

"Hasbro, Inc. (Has) Stock Price, News, Quote & History." *Yahoo! Finance*, Yahoo!, 4 Dec. 2025, finance.yahoo.com/quote/HAS/.

Lian, Jinguo. "Math 456: Mathematical Modeling." Nov. 2025, Amherst, Lederle Tower.