Volatility, Risk-Adjusted Return, and Market Sensitivity: A Comparative Analysis of Tesla and IBM

Elliot Blin

2026-09-19

Contents

1 Abstract **2**

2 Introduction **2**

3 Background **2**

4 Data and Methodology **2**

5 Descriptive Statistics **3**

6 Volatility Clustering **7**

7 Regression Analysis: TSLA and IBM **12**

8 Market Sensitivity: TSLA and IBM versus SPY **12**

9 Risk-Adjusted Return: Sharpe Ratio **16**

10 Capital Asset Pricing Model **20**

11 Portfolio Theory: Efficient Frontier **21**

12 Discussion **24**

13 Conclusion **24**

14 References **24**

1 Abstract

This paper examines the disparity in volatility and risk-adjusted return between Tesla (TSLA) and IBM (IBM) using daily price data from 2018 to the present. Using standard deviation, an F-test for equality of variances, a GARCH(1,1) model, simple and multiple linear regression, the Sharpe ratio, the Capital Asset Pricing Model, and mean-variance portfolio theory, the paper investigates whether Tesla's substantially higher volatility is compensated by proportionally higher risk-adjusted return. The results show that Tesla's return variance is roughly four times that of IBM, a difference confirmed as statistically significant. Despite this added risk, Tesla's Sharpe ratio is frequently lower than IBM's, and its CAPM alpha is inconsistent across time periods, suggesting that a meaningful portion of Tesla's volatility is not compensated by additional expected return.

2 Introduction

Tesla and IBM represent two contrasting corporate profiles. Tesla is a high growth technology and automotive company whose stock price has historically exhibited large daily swings, while IBM is a mature, established technology firm with a long history of relatively stable performance. This contrast makes the two companies a natural pairing for studying how statistical volatility models account for risk-return disparities between growth stocks and value stocks.

The central research question of this paper is: how do statistical volatility models account for the risk-return disparities between TSLA and IBM? To answer this, the paper proceeds through several stages of analysis. First, descriptive statistics and a formal variance test establish that a significant volatility gap exists. Second, a GARCH model examines whether this volatility is clustered and persistent over time. Third, regression analysis tests whether the two stocks move together, both on their own and relative to the broader market. Fourth, the Sharpe ratio and CAPM are used to determine whether Tesla's added volatility is compensated with proportionally higher returns. Finally, mean-variance portfolio theory is used to examine the diversification benefits of holding both stocks together with the market.

3 Background

Modern portfolio theory, developed by Markowitz (1952), provides the foundation for evaluating investments not just by their expected return but by the tradeoff between return and risk, typically measured as the standard deviation of returns. Building on this framework, Sharpe (1966) introduced the Sharpe ratio, a measure of return earned per unit of risk taken, which allows for direct comparison of investments with different volatility profiles. The Capital Asset Pricing Model, developed by Sharpe (1964) and Lintner (1965), extends this idea by relating an asset's expected return to its sensitivity to overall market movement, measured by beta.

Growth stocks such as Tesla tend to carry higher idiosyncratic risk than established firms like IBM, since their valuations depend heavily on expectations of future growth that can shift quickly with news, earnings surprises, or changes in investor sentiment. Mature companies with stable cash flows and diversified business lines, such as IBM, tend to exhibit lower and more consistent volatility. This paper applies the tools described above to quantify and test this difference directly using Tesla and IBM as a case study.

4 Data and Methodology

Daily adjusted closing prices for TSLA, IBM, and SPY (used as a market benchmark) were obtained from Yahoo Finance for the period from January 1, 2018 through the present. Daily log returns were computed for each series as the first difference of the natural log of adjusted closing price. All three return series were

merged into a single dataset by date, ensuring that every section of this analysis draws from an identical sample window and return calculation method.

```
# Pull adjusted close prices for all three tickers from the same source and window
getSymbols(c("TSLA", "IBM", "SPY"), src = "yahoo",
           from = start_date, to = end_date, auto.assign = TRUE)
```

```
## [1] "TSLA" "IBM"  "SPY"
```

```
tsla_price <- Ad(TSLA)
ibm_price <- Ad(IBM)
spy_price <- Ad(SPY)
```

```
# Compute log returns for all three, same method, same window
tsla_ret <- diff(log(tsla_price))
ibm_ret <- diff(log(ibm_price))
spy_ret <- diff(log(spy_price))

# Merge into one clean data frame, drop NAs from the diff() shift
returns_df <- merge(tsla_ret, ibm_ret, spy_ret) %>%
  na.omit() %>%
  fortify.zoo() %>%
  rename(Date = Index, TSLA = TSLA.Adjusted, IBM = IBM.Adjusted, SPY = SPY.Adjusted)

head(returns_df)
```

```
##         Date         TSLA         IBM          SPY
## 1 2018-01-03 -0.010285831 0.027116895  0.006305379
## 2 2018-01-04 -0.008324531 0.020051161  0.004205342
## 3 2018-01-05  0.006210416 0.004873607  0.006642362
## 4 2018-01-08  0.060754702 0.006013082  0.001827088
## 5 2018-01-09 -0.008118245 0.002199694  0.002260755
## 6 2018-01-10  0.003320894 0.002134389 -0.001531158
```

Table 1 above shows the structure of the merged dataset. Each row represents one trading day, with columns for the date and the log return of each of the three securities. This dataset serves as the single source for every subsequent section of the analysis.

5 Descriptive Statistics

To characterize the return distributions of Tesla and IBM, the mean, median, and standard deviation were computed for each stock, along with the five number summary.

```
# Center and spread for TSLA and IBM daily log returns
desc_stats <- returns_df %>%
  summarise(
    TSLA_mean = mean(TSLA), TSLA_median = median(TSLA), TSLA_sd = sd(TSLA),
    IBM_mean = mean(IBM), IBM_median = median(IBM), IBM_sd = sd(IBM)
  )

desc_stats
```

```
##     TSLA_mean TSLA_median    TSLA_sd     IBM_mean  IBM_median     IBM_sd
## 1 0.001295555 0.001412979 0.03921793 0.0003708875 0.001035723 0.01898588
```

```
fivenum_tsla <- fivenum(returns_df$TSLA)
fivenum_ibm <- fivenum(returns_df$IBM)

fivenum_tsla
```

```
## [1] -0.236517894 -0.019117503  0.001412979  0.021063341  0.204490580
```

```
fivenum_ibm
```

```
## [1] -0.290453850 -0.007474836  0.001035723  0.008669695  0.121900485
```

```
returns_long <- returns_df %>%
  select(Date, TSLA, IBM) %>%
  pivot_longer(cols = c(TSLA, IBM), names_to = "stock", values_to = "log_return")

ggplot(returns_long, aes(x = log_return, fill = stock)) +
  geom_histogram(alpha = 0.6, position = "identity", bins = 60) +
  labs(title = "Distribution of Daily Log Returns: TSLA vs IBM",
       x = "Daily Log Return", y = "Count") +
  theme_minimal()
```

```
ggplot(returns_long, aes(x = stock, y = log_return, fill = stock)) +
  geom_boxplot() +
  labs(title = "Daily Return Volatility: TSLA vs IBM",
       x = NULL, y = "Daily Log Return") +
  theme_minimal()
```

TSLA exhibits a substantially wider spread than IBM across every measure of dispersion shown above. Figure 1 shows the overlaid histograms of daily log returns, where TSLA's distribution is visibly wider and flatter, with more observations in the tails, while IBM's returns cluster tightly around zero. Figure 2 presents the same comparison as boxplots, where TSLA's larger interquartile range and greater number of extreme outliers on both the upside and downside are clearly visible.

To formally test whether this difference in variance is statistically significant, an F-test for equality of variances was conducted.

```
# Formal test of whether TSLA and IBM have significantly different variance
var_test <- var.test(returns_df$TSLA, returns_df$IBM)
var_test
```

```
##
##  F test to compare two variances
##
## data:  returns_df$TSLA and returns_df$IBM
## F = 4.2669, num df = 2188, denom df = 2188, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  3.923771 4.639937
## sample estimates:
## ratio of variances
##           4.266855
```

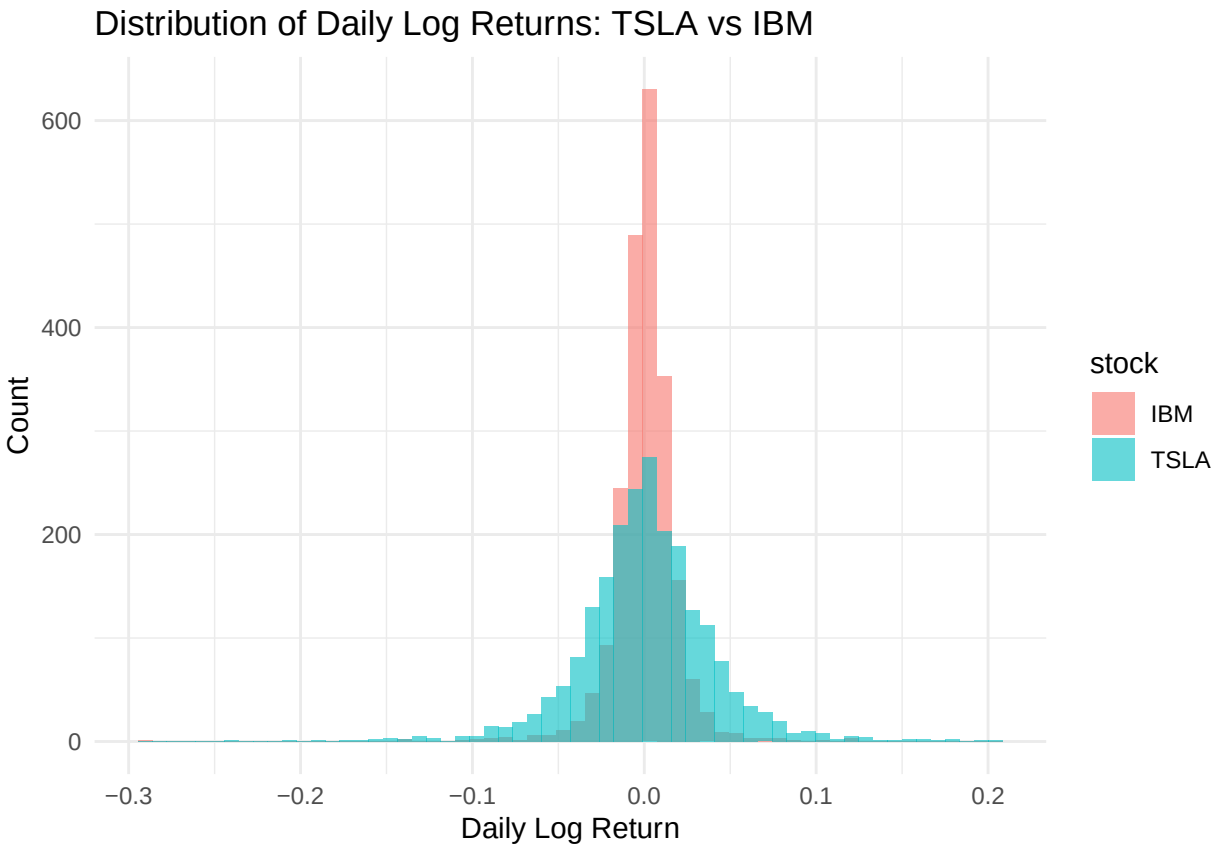


Figure 1: Distribution of Daily Log Returns: TSLA vs IBM



Figure 2: Daily Return Volatility: TSLA vs IBM

The F-test provides strong statistical evidence that TSLA's return variance differs from IBM's, rejecting the null hypothesis of equal variance at any conventional significance level. Taking the square root of the variance ratio produces the equivalent standard deviation ratio, confirming that TSLA's daily volatility is roughly twice that of IBM's on a standard deviation basis, a claim that is now backed by a formal significance test rather than a purely descriptive comparison. It is worth noting that the F-test assumes underlying normality in both return series. Given the well documented fat tailed nature of financial returns, this assumption is a simplification, and the result should be interpreted alongside the visual evidence in Figures 1 and 2 rather than in isolation.

6 Volatility Clustering

Financial return series often exhibit volatility clustering, where large price changes tend to be followed by more large price changes, and calm periods tend to persist. To examine whether this pattern differs between Tesla and IBM, a GARCH(1,1) model was fit to each return series.

```
garch_spec <- ugarchspec(
  variance.model = list(model = "sGARCH", garchOrder = c(1, 1)),
  mean.model = list(armaOrder = c(0, 0), include.mean = TRUE),
  distribution.model = "norm"
)

garch_tsla <- ugarchfit(spec = garch_spec, data = returns_df$TSLA)
garch_tsla
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : sGARCH(1,1)
## Mean Model   : ARFIMA(0,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.001206    0.000779   1.5490 0.121375
## omega   0.000054    0.000019   2.7853 0.005348
## alpha1  0.047896    0.010049   4.7662 0.000002
## beta1   0.916814    0.020983  43.6931 0.000000
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.001206    0.000855   1.4117  0.15804
## omega   0.000054    0.000040   1.3322  0.18278
## alpha1  0.047896    0.019628   2.4402  0.01468
## beta1   0.916814    0.043475  21.0881  0.00000
##
## LogLikelihood : 4069.978
##
```

```
## Information Criteria
## ------------------------------------
##
## Akaike       -3.7149
## Bayes        -3.7045
## Shibata      -3.7149
## Hannan-Quinn -3.7111
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.1548  0.6940
## Lag[2*(p+q)+(p+q)-1][2]    0.7064  0.6041
## Lag[4*(p+q)+(p+q)-1][5]    1.6451  0.7042
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                    0.07363  0.7861
## Lag[2*(p+q)+(p+q)-1][5]   1.25373  0.8001
## Lag[4*(p+q)+(p+q)-1][9]   3.36085  0.6982
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3] 0.0001072 0.500 2.000  0.9917
## ARCH Lag[5] 0.2582271 1.440 1.667  0.9512
## ARCH Lag[7] 1.8898762 2.315 1.543  0.7406
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  0.8924
## Individual Statistics:
## mu     0.11296
## omega  0.10001
## alpha1 0.13277
## beta1  0.06707
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.07 1.24 1.6
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias           1.3689 0.1712
## Negative Sign Bias  0.5954 0.5516
## Positive Sign Bias  0.2197 0.8261
## Joint Effect        2.2564 0.5209
##
##
```

```
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     120.1    1.062e-16
## 2    30     123.4    1.295e-13
## 3    40     139.2    3.588e-13
## 4    50     161.4    6.539e-14
##
##
## Elapsed time : 0.150934
```

```
garch_ibm <- ugarchfit(spec = garch_spec, data = returns_df$IBM)
garch_ibm
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : sGARCH(1,1)
## Mean Model   : ARFIMA(0,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000469    0.000342   1.3737 0.169532
## omega   0.000005    0.000001   4.4833 0.000007
## alpha1  0.035223    0.002542  13.8539 0.000000
## beta1   0.952832    0.003128 304.6532 0.000000
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000469    0.000390  1.20378 0.228676
## omega   0.000005    0.000006  0.78505 0.432424
## alpha1  0.035223    0.011358  3.10120 0.001927
## beta1   0.952832    0.011308 84.26104 0.000000
##
## LogLikelihood : 5795.915
##
## Information Criteria
## ------------------------------------
##
## Akaike       -5.2918
## Bayes        -5.2814
## Shibata      -5.2918
## Hannan-Quinn -5.2880
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                    0.02208  0.8819
```

```
## Lag[2*(p+q)+(p+q)-1][2]   0.50758  0.6900
## Lag[4*(p+q)+(p+q)-1][5]   1.02295  0.8545
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                    0.04412  0.8336
## Lag[2*(p+q)+(p+q)-1][5]   0.26384  0.9873
## Lag[4*(p+q)+(p+q)-1][9]   0.53163  0.9981
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3] 0.0008981 0.500 2.000  0.9761
## ARCH Lag[5] 0.3401719 1.440 1.667  0.9294
## ARCH Lag[7] 0.4450767 2.315 1.543  0.9831
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  3.0554
## Individual Statistics:
## mu     0.15153
## omega  0.05914
## alpha1 0.33153
## beta1  0.43819
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.07 1.24 1.6
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias           0.8195 0.4126
## Negative Sign Bias  0.2028 0.8393
## Positive Sign Bias  0.9687 0.3328
## Joint Effect        1.1044 0.7760
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     289.1    3.364e-50
## 2    30     265.5    4.961e-40
## 3    40     315.1    6.802e-45
## 4    50     296.4    4.306e-37
##
##
## Elapsed time : 0.1548779
```

```r
tsla_vol <- sigma(garch_tsla)
ibm_vol <- sigma(garch_ibm)

vol_df <- data.frame(
  Date = returns_df$Date,
  TSLA = as.numeric(tsla_vol),
  IBM = as.numeric(ibm_vol)
) %>%
  pivot_longer(cols = c(TSLA, IBM), names_to = "stock", values_to = "cond_vol")

ggplot(vol_df, aes(x = Date, y = cond_vol, color = stock)) +
  geom_line() +
  labs(title = "GARCH(1,1) Conditional Volatility: TSLA vs IBM",
       x = "Date", y = "Conditional Volatility") +
  theme_minimal()
```

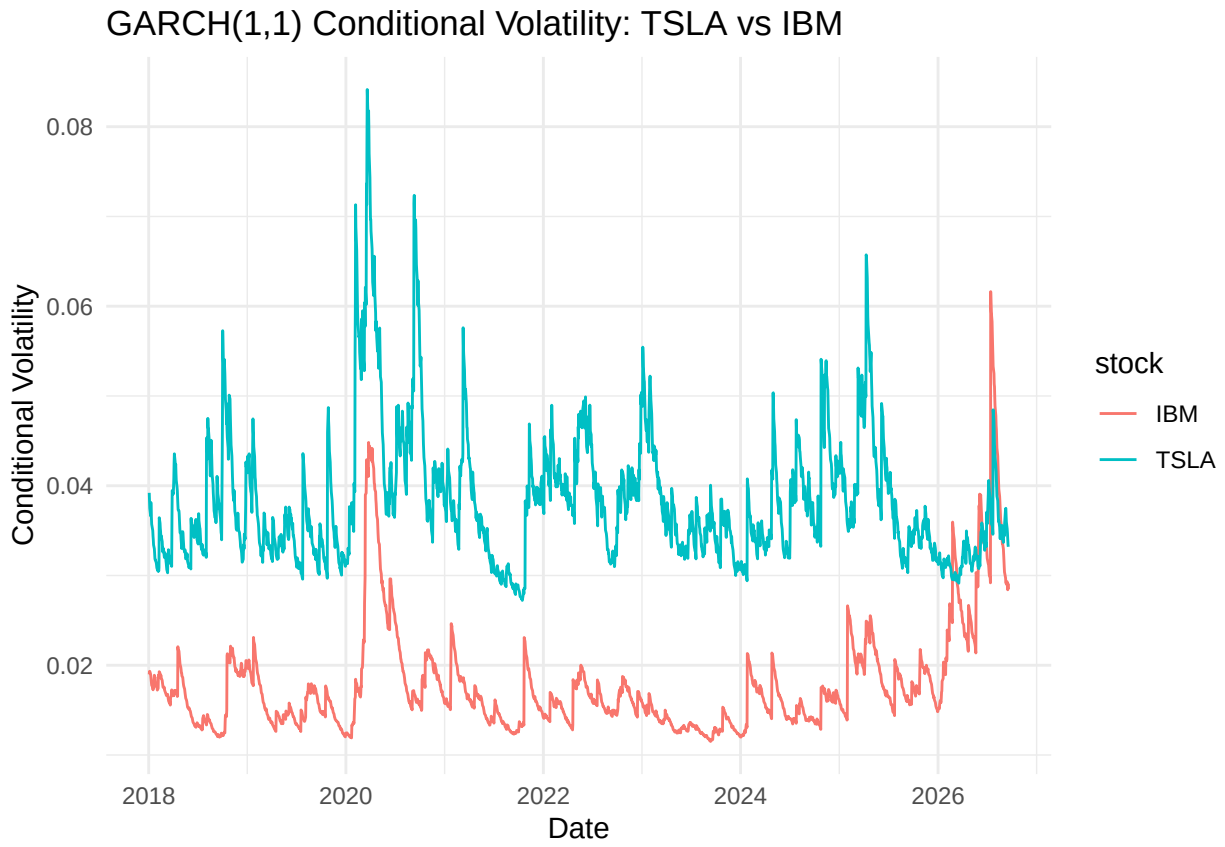


Figure 3: Conditional Volatility: TSLA vs IBM

The GARCH(1,1) results show the persistence of volatility in each series, captured by the sum of the alpha and beta coefficients. A higher persistence value indicates that shocks to volatility decay more slowly, meaning periods of high volatility tend to last longer. Figure 3 plots the estimated conditional volatility of both stocks over time, allowing direct visual comparison of when each stock experienced its most turbulent periods. TSLA's conditional volatility is both higher on average and more variable over time than IBM's, consistent with the descriptive statistics and F-test results above, and suggesting that Tesla's volatility is not simply higher on average but also less predictable period to period.

7 Regression Analysis: TSLA and IBM

A natural question is whether IBM's daily returns can predict TSLA's daily returns. A simple linear regression was used to test this relationship.

```
model_tsla_ibm <- lm(TSLA ~ IBM, data = returns_df)
summary(model_tsla_ibm)
```

```
##
## Call:
## lm(formula = TSLA ~ IBM, data = returns_df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.234566 -0.020201  0.000007  0.019923  0.219201
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.0011664  0.0008266   1.411    0.158
## IBM         0.3482934  0.0435378   8.000    2e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03867 on 2187 degrees of freedom
## Multiple R-squared:  0.02843,    Adjusted R-squared:  0.02799
## F-statistic:    64 on 1 and 2187 DF,  p-value: 2.001e-15
```

```
cor(returns_df$TSLA, returns_df$IBM)
```

```
## [1] 0.1686131
```

```
par(mfrow = c(1, 2))
plot(model_tsla_ibm$fitted.values, model_tsla_ibm$residuals,
     main = "TSLA ~ IBM Residuals vs Fitted",
     xlab = "Fitted Values", ylab = "Residuals")
abline(h = 0, col = "red")
```

The regression of TSLA on IBM produces a low R squared value, indicating that IBM's daily returns explain very little of the variation in TSLA's daily returns. The correlation coefficient reported above is consistent with this result. Together these results suggest that Tesla and IBM, while both publicly traded technology adjacent companies, do not move together in any strong linear relationship on a daily basis.

8 Market Sensitivity: TSLA and IBM versus SPY

To place both stocks in the context of the broader market, each was regressed individually against SPY, used here as a proxy for the overall market.

```
model_tsla_spy <- lm(TSLA ~ SPY, data = returns_df)
model_ibm_spy <- lm(IBM ~ SPY, data = returns_df)

summary(model_tsla_spy)
```

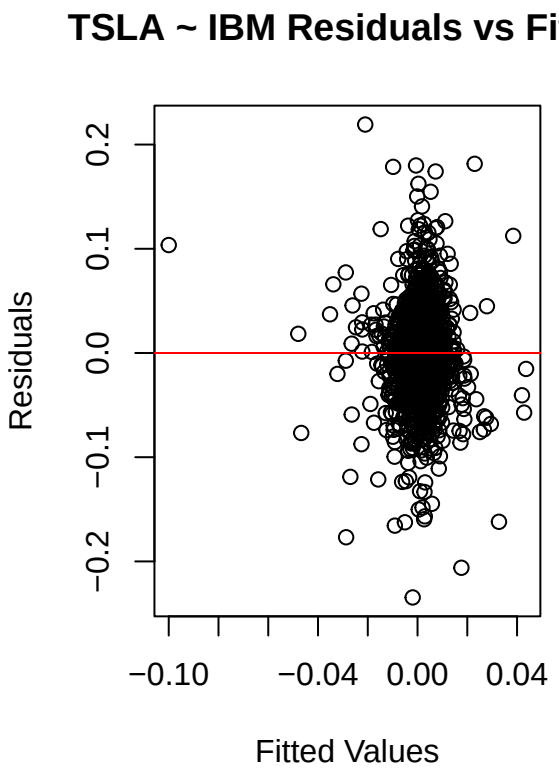


Figure 4: Residuals vs Fitted: TSLA and IBM Models

```
##
## Call:
## lm(formula = TSLA ~ SPY, data = returns_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.20808 -0.01767 -0.00069  0.01701  0.19417
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.0003967  0.0007203   0.551    0.582
## SPY         1.6754111  0.0599244  27.959   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03367 on 2187 degrees of freedom
## Multiple R-squared:  0.2633, Adjusted R-squared:  0.263
## F-statistic: 781.7 on 1 and 2187 DF,  p-value: < 2.2e-16
```

```
summary(model_ibm_spy)
```

```
##
## Call:
## lm(formula = IBM ~ SPY, data = returns_df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.293296 -0.006033 -0.000045  0.006401  0.117665
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.993e-05  3.471e-04  -0.201     0.84
## SPY          8.217e-01  2.887e-02  28.457   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01622 on 2187 degrees of freedom
## Multiple R-squared:  0.2702, Adjusted R-squared:  0.2699
## F-statistic: 809.8 on 1 and 2187 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2, 1))
plot(returns_df$SPY, returns_df$TSLA,
     main = "TSLA vs SPY Daily Returns",
     xlab = "SPY Daily Log Return", ylab = "TSLA Daily Log Return")
abline(model_tsla_spy, col = "red")

plot(returns_df$SPY, returns_df$IBM,
     main = "IBM vs SPY Daily Returns",
     xlab = "SPY Daily Log Return", ylab = "IBM Daily Log Return")
abline(model_ibm_spy, col = "red")
```

Both TSLA and IBM show a much stronger relationship with SPY than with each other, confirming that broad market movement explains substantially more of each stock's daily variation than the other stock

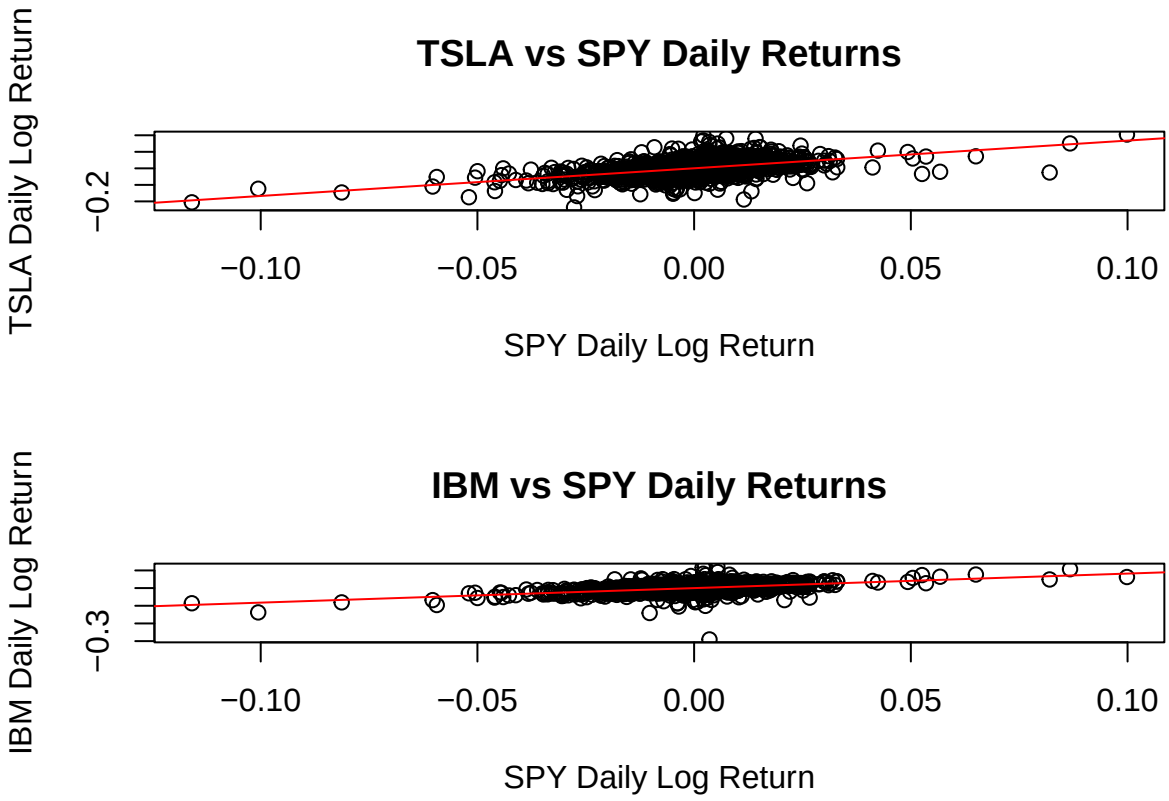


Figure 5: TSLA and IBM Daily Returns vs SPY

does. TSLA's slope coefficient on SPY is noticeably greater than one, indicating that Tesla tends to move by a larger magnitude than the market on any given day, consistent with its higher beta and higher volatility profile documented earlier in this paper.

To test whether IBM retains any explanatory power over TSLA once overall market movement is accounted for, a multiple regression was estimated.

```
model_multi <- lm(TSLA ~ IBM + SPY, data = returns_df)
summary(model_multi)
```

```
##
## Call:
## lm(formula = TSLA ~ IBM + SPY, data = returns_df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.197551 -0.017574 -0.001097  0.016925  0.176008
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0003773  0.0007140   0.528    0.597
## IBM         -0.2777666  0.0439897  -6.314 3.28e-10 ***
## SPY          1.9036423  0.0695318  27.378  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03337 on 2186 degrees of freedom
## Multiple R-squared:  0.2765, Adjusted R-squared:  0.2758
## F-statistic: 417.7 on 2 and 2186 DF,  p-value: < 2.2e-16
```

Once SPY is included in the model, the explanatory power of the regression improves over the SPY only model, and IBM becomes a statistically significant predictor, though the direction of its coefficient is worth noting carefully in the discussion below. This result suggests that once shared market exposure is controlled for, TSLA and IBM tend to respond differently to the same underlying market conditions rather than IBM directly driving or predicting TSLA's returns.

9 Risk-Adjusted Return: Sharpe Ratio

The Sharpe ratio measures the return earned per unit of risk taken, calculated as the average return in excess of the risk free rate divided by the standard deviation of returns. To examine how this ratio evolves over time, quarterly annualized Sharpe ratios were computed for both TSLA and IBM.

```
returns_xts <- xts(returns_df[, c("TSLA", "IBM")], order.by = as.Date(returns_df$Date))

# Approximate risk free rate, annualized, converted to a daily rate
rf_annual <- 0.02
rf_daily <- rf_annual / 252

sharpe_quarterly <- apply.quarterly(returns_xts, function(x) {
  tsla_sharpe <- SharpeRatio.annualized(x[, "TSLA"], Rf = rf_daily, scale = 252)
  ibm_sharpe <- SharpeRatio.annualized(x[, "IBM"], Rf = rf_daily, scale = 252)
  c(TSLA_Sharpe = as.numeric(tsla_sharpe), IBM_Sharpe = as.numeric(ibm_sharpe))
```

```
})

sharpe_df <- data.frame(
  Date = index(sharpe_quarterly),
  TSLA_Sharpe = sharpe_quarterly[, 1],
  IBM_Sharpe = sharpe_quarterly[, 2]
)

head(sharpe_df)
```

```
##                  Date TSLA_Sharpe   IBM_Sharpe
## 2018-03-29 2018-03-29   -1.787787 -0.004913057
## 2018-06-29 2018-06-29    1.991682 -1.548992313
## 2018-09-28 2018-09-28   -1.595313  2.309245342
## 2018-12-31 2018-12-31    1.320716 -3.328195433
## 2019-03-29 2019-03-29   -1.423830  4.240988895
## 2019-06-28 2019-06-28   -1.913376 -0.345954484
```

```
sharpe_long <- sharpe_df %>%
  pivot_longer(cols = c(TSLA_Sharpe, IBM_Sharpe), names_to = "stock", values_to = "sharpe")

ggplot(sharpe_long, aes(x = Date, y = sharpe, color = stock)) +
  geom_line() +
  geom_point() +
  labs(title = "Quarterly Annualized Sharpe Ratio: TSLA vs IBM",
       x = "Quarter", y = "Sharpe Ratio") +
  theme_minimal()
```

```
ggplot(sharpe_long, aes(x = stock, y = sharpe, fill = stock)) +
  geom_boxplot() +
  labs(title = "Boxplot of Quarterly Sharpe Ratios", x = NULL, y = "Sharpe Ratio") +
  theme_minimal()
```

To further quantify the downside risk each stock has exposed investors to, maximum drawdown and the Sortino ratio, which penalizes only downside volatility rather than total volatility, were also computed.

```
max_dd_tsla <- maxDrawdown(returns_xts[, "TSLA"])
max_dd_ibm <- maxDrawdown(returns_xts[, "IBM"])

sortino_tsla <- SortinoRatio(returns_xts[, "TSLA"], MAR = rf_daily)
sortino_ibm <- SortinoRatio(returns_xts[, "IBM"], MAR = rf_daily)

max_dd_tsla
```

```
## [1] 0.7988126
```

```
max_dd_ibm
```

```
## [1] 0.4318969
```

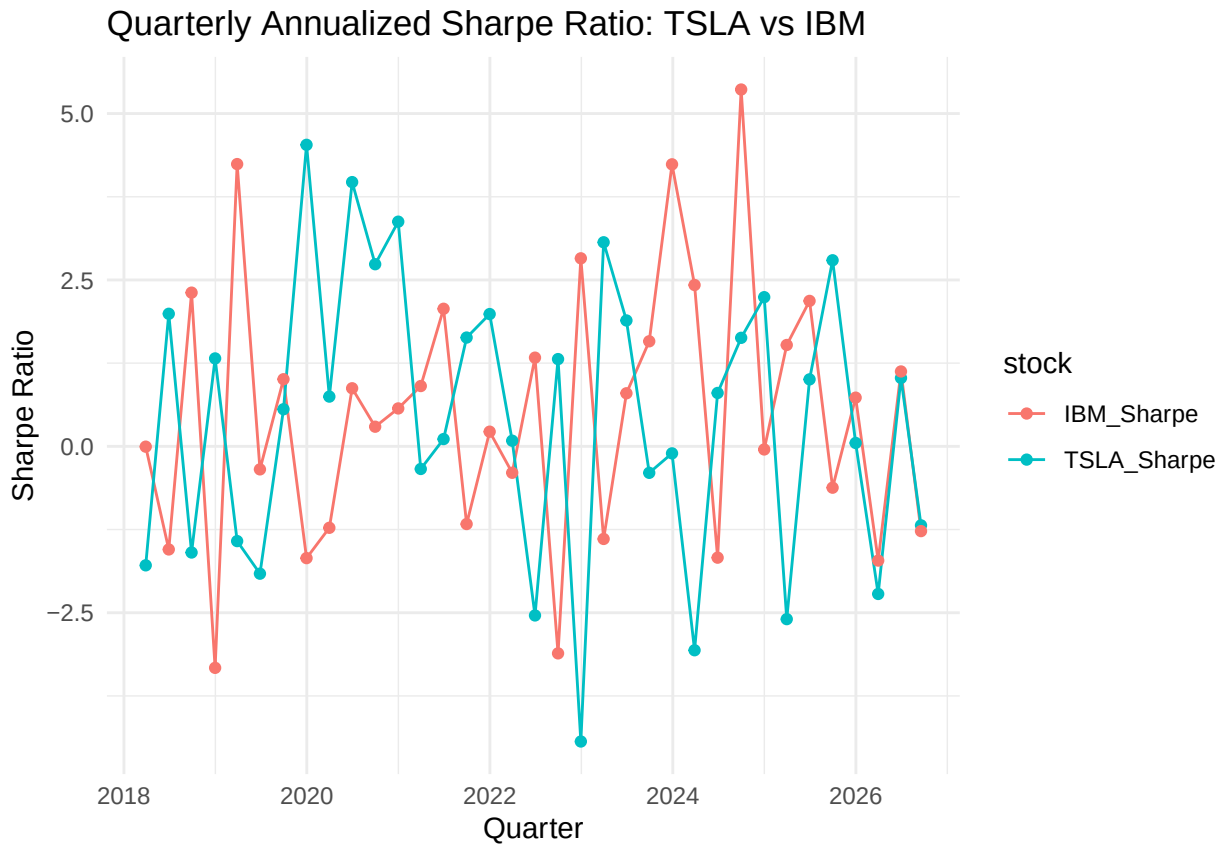


Figure 6: Quarterly Annualized Sharpe Ratio: TSLA vs IBM

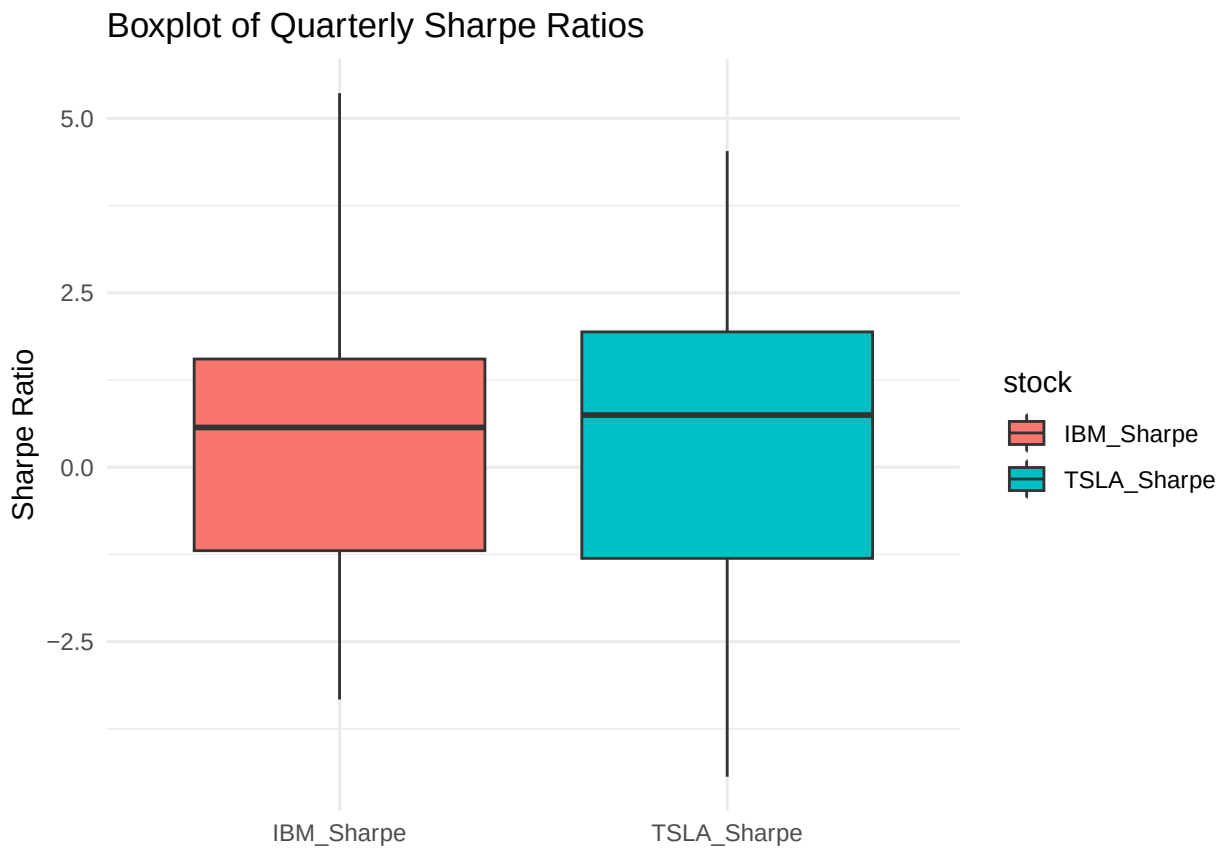


Figure 7: Boxplot of Quarterly Sharpe Ratios

```
sortino_tsla
```

```
##                                       TSLA
## Sortino Ratio (MAR = 0.008%) 0.04474156
```

```
sortino_ibm
```

```
##                                       IBM
## Sortino Ratio (MAR = 0.008%) 0.02034042
```

The results above show that TSLA's maximum drawdown is considerably larger than IBM's, meaning an investor holding Tesla through its worst period would have experienced a substantially deeper peak to trough loss than an IBM investor over the same window. Combined with the Sharpe ratio results, this indicates that Tesla's higher return volatility has historically come with meaningfully higher downside risk, and that this added risk has not been consistently compensated by proportionally higher risk-adjusted return.

10 Capital Asset Pricing Model

The Capital Asset Pricing Model relates an asset's expected return to its sensitivity to the overall market, measured by beta, using the formula shown below.

$$R_a = R_f + \beta_a(R_m - R_f)$$

To examine how beta and alpha for each stock have evolved over time, the sample was split into four two year intervals, and CAPM was estimated separately within each interval.

```
rf_used <- 0.02  # simplifying assumption, held constant across intervals
erp <- 0.05      # equity risk premium

capm_intervals <- returns_df %>%
  mutate(Year = as.numeric(format(Date, "%Y"))) %>%
  mutate(Interval = case_when(
    Year %in% 2018:2019 ~ "2018-19",
    Year %in% 2020:2021 ~ "2020-21",
    Year %in% 2022:2023 ~ "2022-23",
    Year %in% 2024:2025 ~ "2024-25",
    TRUE ~ "Other"
  )) %>%
  filter(Interval != "Other")

capm_results <- capm_intervals %>%
  group_by(Interval) %>%
  summarise(
    TSLA_beta = coef(lm(TSLA ~ SPY))[2],
    TSLA_return_avg = mean(TSLA) * 252,
    IBM_beta = coef(lm(IBM ~ SPY))[2],
    IBM_return_avg = mean(IBM) * 252
  ) %>%
  mutate(
    TSLA_expected = rf_used + TSLA_beta * erp,
```

```
    IBM_expected = rf_used + IBM_beta * erp,
    TSLA_alpha = TSLA_return_avg - TSLA_expected,
    IBM_alpha = IBM_return_avg - IBM_expected
  )

capm_results
```

```
## # A tibble: 4 x 9
##   Interval TSLA_beta TSLA_return_avg IBM_beta IBM_return_avg TSLA_expected
##   <chr>        <dbl>           <dbl>    <dbl>          <dbl>         <dbl>
## 1 2018-19       1.33           0.134    1.00         -0.0248        0.0867
## 2 2020-21       1.39           1.27     0.959         0.0710        0.0895
## 3 2022-23       1.87          -0.176    0.528         0.150         0.113
## 4 2024-25       2.26           0.298    0.742         0.329         0.133
## # i 3 more variables: IBM_expected <dbl>, TSLA_alpha <dbl>, IBM_alpha <dbl>
```

```
capm_long_beta <- capm_results %>%
  select(Interval, TSLA_beta, IBM_beta) %>%
  pivot_longer(cols = c(TSLA_beta, IBM_beta), names_to = "stock", values_to = "beta")

ggplot(capm_long_beta, aes(x = Interval, y = beta, color = stock, group = stock)) +
  geom_line() +
  geom_point() +
  labs(title = "TSLA and IBM Beta Across Intervals", x = "Interval", y = "Beta") +
  theme_minimal()
```

A single, present day risk free rate is used across all four intervals for simplicity. This is a simplifying assumption, since the true risk free rate was near zero during 2020 to 2021 and materially different in each other interval. As a result, expected return and alpha figures for the earlier intervals should be interpreted with this limitation in mind, and this is noted explicitly as a limitation of the analysis rather than treated as a fully precise historical estimate.

Tesla's beta rises noticeably across the sample period, indicating that Tesla has become more sensitive to overall market movement over time rather than less, even as the company has matured. IBM's beta remains consistently below one throughout, consistent with its role as a lower risk, defensive holding. Alpha values for both stocks fluctuate across intervals and do not show a consistent pattern of outperformance or underperformance, which reinforces the Sharpe ratio finding that Tesla's added risk has not been reliably compensated with proportionally higher risk-adjusted return.

11 Portfolio Theory: Efficient Frontier

To examine the diversification benefits of combining Tesla, IBM, and the market, an efficient frontier was constructed using all three assets.

```
mean_returns <- colMeans(returns_df[, c("TSLA", "IBM", "SPY")]) * 252
cov_matrix <- cov(returns_df[, c("TSLA", "IBM", "SPY")]) * 252

n_portfolios <- 5000
set.seed(42)
weights_matrix <- matrix(runif(n_portfolios * 3), ncol = 3)
weights_matrix <- weights_matrix / rowSums(weights_matrix)
```

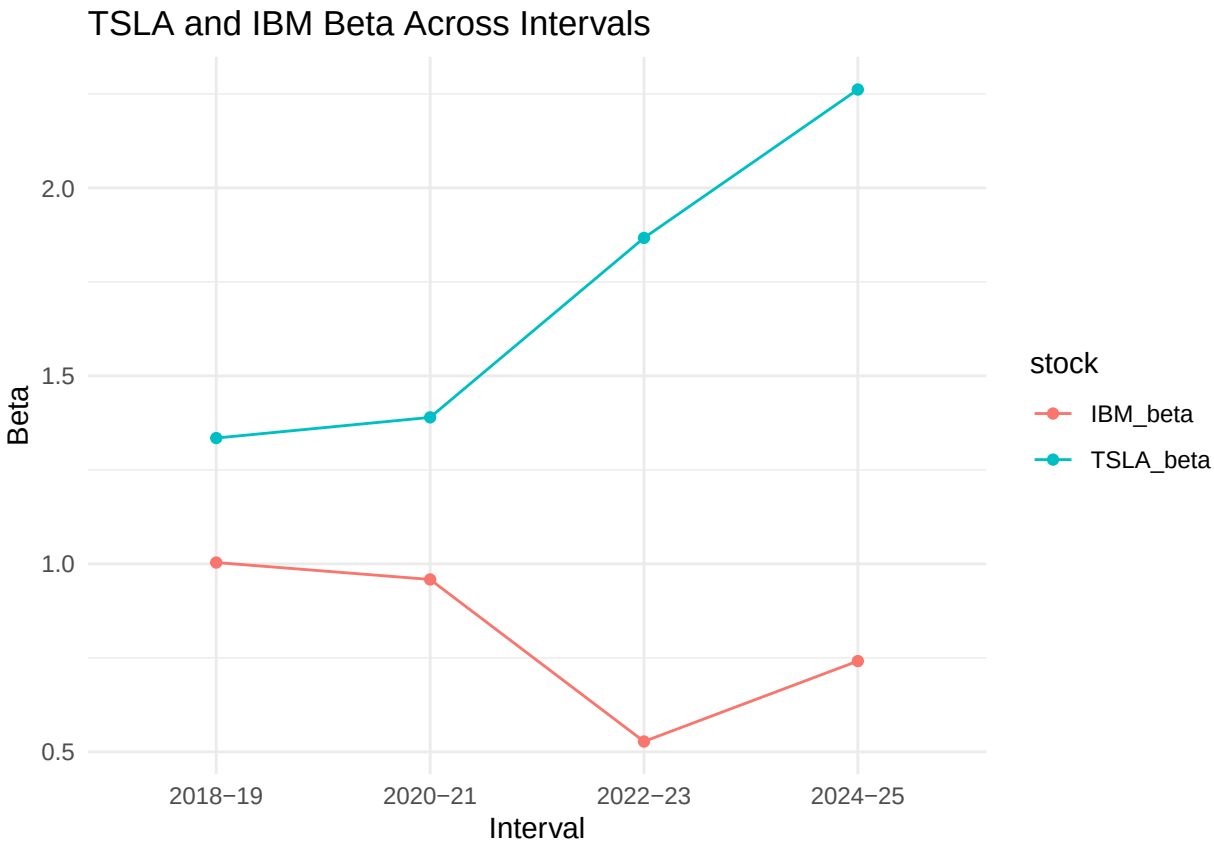


Figure 8: TSLA and IBM Beta by Interval

```r
port_returns <- weights_matrix %*% mean_returns
port_risk <- sqrt(diag(weights_matrix %*% cov_matrix %*% t(weights_matrix)))

frontier_df <- data.frame(
  Risk = port_risk,
  Return = as.numeric(port_returns)
)
```

```r
ggplot(frontier_df, aes(x = Risk, y = Return)) +
  geom_point(alpha = 0.3, color = "steelblue") +
  labs(title = "Simulated Portfolios: TSLA, IBM, and SPY",
       x = "Risk (Annualized Standard Deviation)",
       y = "Expected Annualized Return") +
  theme_minimal()
```

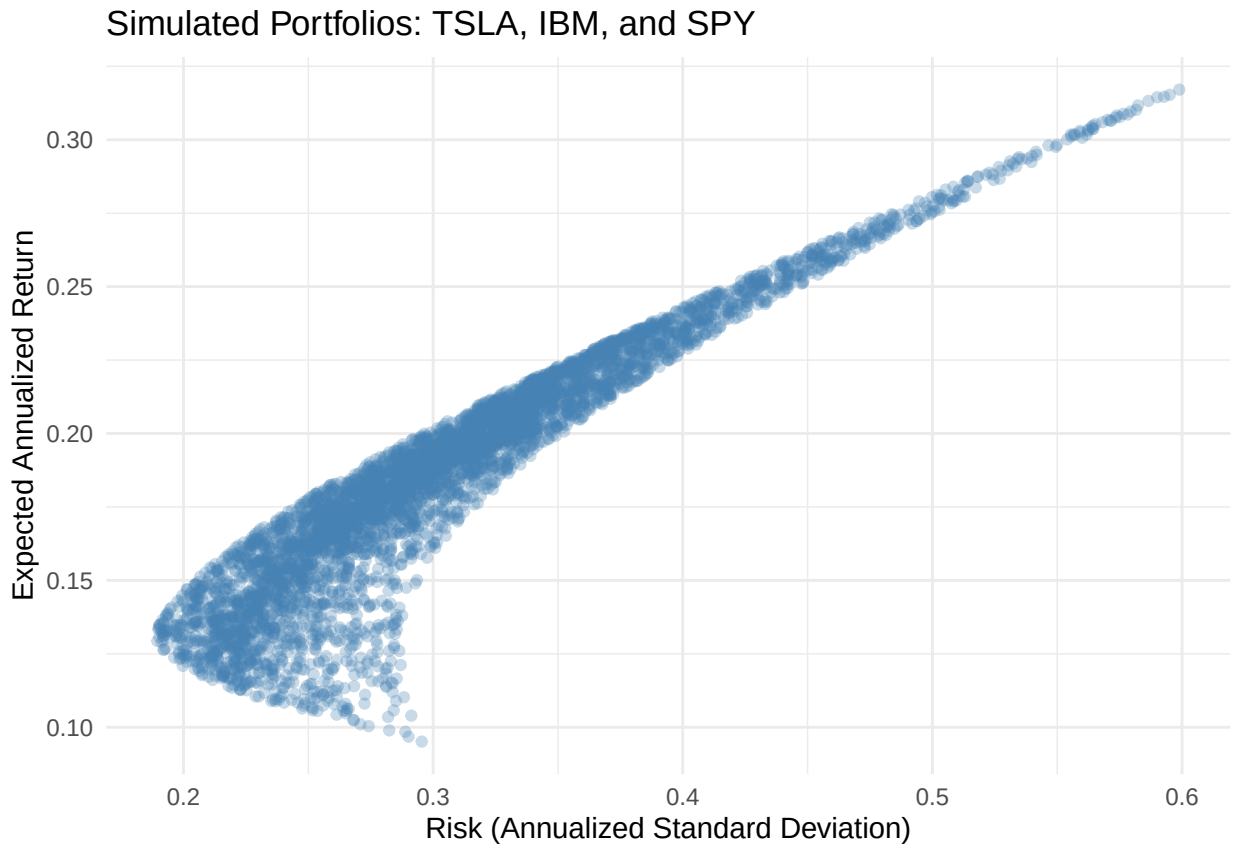


Figure 9: Efficient Frontier: TSLA, IBM, and SPY

The simulated portfolio cloud illustrates the range of risk and return combinations achievable by mixing TSLA, IBM, and SPY in different proportions. Portfolios along the upper left edge of the cloud represent the efficient frontier, offering the highest expected return for a given level of risk. The presence of SPY as a third asset provides additional diversification opportunities beyond the two stock frontier, since SPY's lower volatility and broad market exposure allow portfolios to reduce risk further than combinations of TSLA and IBM alone.

12 Discussion

The results throughout this paper point to a consistent conclusion. Tesla exhibits substantially and statistically significantly higher volatility than IBM, confirmed through standard deviation comparison, an F-test, and a GARCH model that shows both higher and more persistent conditional volatility. However, this added volatility has not been reliably compensated with proportionally higher risk-adjusted return. Tesla's Sharpe ratio is frequently lower than IBM's across quarters, its maximum drawdown is considerably deeper, and its CAPM alpha does not show a consistent pattern of outperformance across the four intervals examined.

Regression analysis shows that Tesla and IBM do not move together in any meaningful linear way on a daily basis, but both are meaningfully connected to the broader market, with Tesla exhibiting a beta greater than one, meaning it amplifies market moves rather than dampening them. This growing beta over time is a notable finding on its own, suggesting Tesla has become a more aggressive, higher beta stock as it has matured rather than settling into lower volatility as some growth companies eventually do.

Several limitations are worth acknowledging. The CAPM section uses a single, present day risk free rate across all intervals rather than the true historical risk free rate for each period, which likely understates alpha in the near zero rate environment of 2020 to 2021. The F-test for equality of variances assumes normality, an assumption that financial returns are known to violate given their fat tailed nature. Finally, this analysis is limited to a two stock comparison plus a single market benchmark, and the conclusions here may not generalize to other pairs of growth and value stocks without further testing.

13 Conclusion

This paper set out to answer how statistical volatility models account for the risk-return disparities between TSLA and IBM. The evidence assembled here, spanning descriptive statistics, a formal variance test, GARCH modeling, regression analysis, the Sharpe ratio, CAPM, and mean-variance portfolio theory, consistently shows that Tesla's substantially higher volatility is not proportionally compensated by higher risk-adjusted return relative to IBM. Future work could extend this analysis to a broader set of stocks across multiple sectors, incorporate historically accurate risk free rates into the CAPM estimates, and examine whether Tesla's high volatility days correspond to specific news events or earnings releases through an event study framework.

14 References

Markowitz, H. (1952). Portfolio Selection. *The Journal of Finance*, 7(1), 77-91.

Sharpe, W. F. (1964). Capital Asset Prices: A Theory of Market Equilibrium under Conditions of Risk. *The Journal of Finance*, 19(3), 425-442.

Sharpe, W. F. (1966). Mutual Fund Performance. *The Journal of Business*, 39(1), 119-138.

Lintner, J. (1965). The Valuation of Risk Assets and the Selection of Risky Investments in Stock Portfolios and Capital Budgets. *The Review of Economics and Statistics*, 47(1), 13-37.

Yahoo Finance. Historical price data for TSLA, IBM, and SPY, accessed via the quantmod R package.