ME317 Project Report

What Happened During the COVID-19 Pandemic?

Fabio Antonicoli

July 27, 2025

Introduction

Early 2020 saw one of the most volatile times in recent financial history because of the COVID-19 outbreak. Global stock markets reacted with extreme volatility, and investors were faced with an unusual level of uncertainty. This report undertakes an analysis of ten diversified stocks during and after the pandemic shock, using techniques from the ME317 course. Return dynamics are explored, a risk model is developed and tested using Value-at-Risk (VaR), and the dependence structure between financial institutions is scrutinized using copulas.

1. Data Collection and Preparation

We selected 10 major US stocks from various execution centers, thus providing a comprehensive market picture. Daily adjusted stock prices from January 1, 2019, to December 31, 2022.

2. Market Dynamics During the Pandemic

Visual analysis of the trend in the stock price detected a steep decline between February and March 2020, coinciding with the outbreak of COVID-19 globally and the lockdowns enforced. Shares in nearly all industries lost value but to varying degrees.

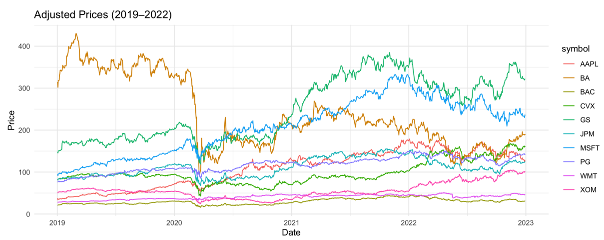

Figure 1: Adjusted stock prices (2019–2022) for the 10 selected companies.

Log-returns plotted over time show a significant spike in volatility during early 2020. For example, Boeing (BA) and JPMorgan (JPM) exhibited single-day losses exceeding 10%.

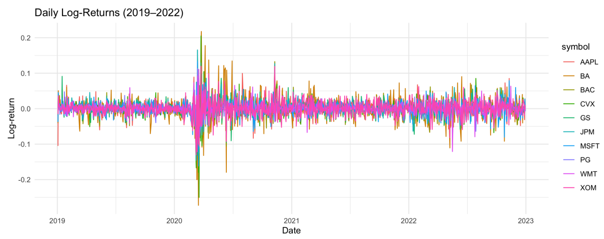

Figure 2: Daily log-returns of all 10 stocks. Volatility increases markedly in March 2020.

3. Sector Relationships and Diversification Effects

To understand co-movement, we constructed scatter plots of daily returns between selected pairs of stocks.

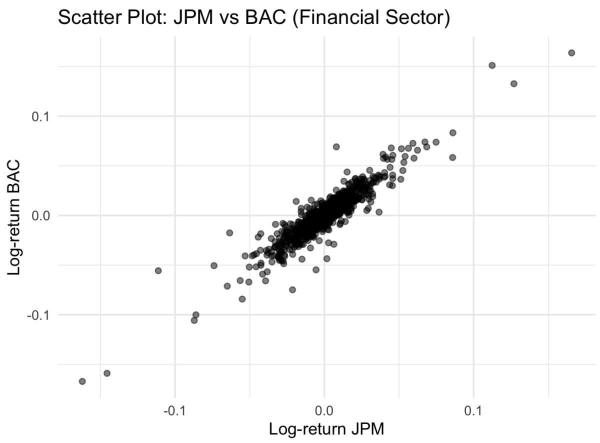

Figure 3: Scatter plot: JPM vs BAC — strong linear correlation within the financial sector.

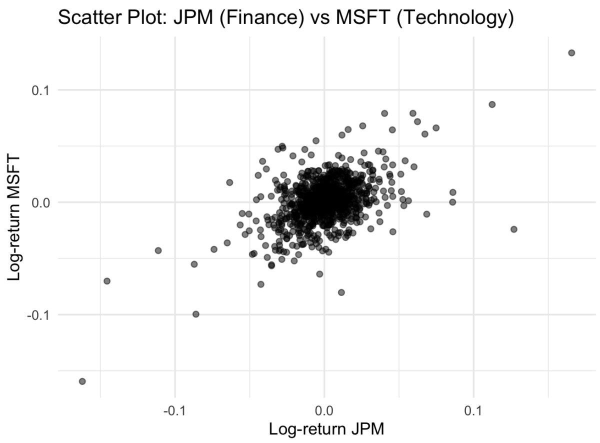

Figure 4: Scatter plot: JPM vs MSFT — weaker correlation between different sectors.

Plots confirm that during normal periods, cross-sector diversification offers some risk reduction, but during crises, correlations tend to increase.

4. Portfolio Construction and VaR Estimation

We create an equally weighted portfolio by investing \$1000 in each of the 10 stocks. The daily portfolio return is:

$$\text{VaR}_{\alpha}^{\text{emp}} = \text{Quantile}_{1-\alpha}(R_t^{\text{port}}), \quad \text{VaR}_{\alpha}^{\text{norm}} = \mu - z_{\alpha} \cdot \sigma$$

estimated the 1-day VaR at the 95% confidence level using two methods:

$$\text{VaR}_{\text{emp}} = \text{Quantile}_{0.05}(R_t^{\text{port}}), \quad \text{VaR}_{\text{norm}} = \mu - 1.645 \cdot \sigma$$

- Empirical VaR: -2.24%
- Normal VaR: -2.66%

The plot below illustrates the log-returns of the portfolio over the full analysis period.

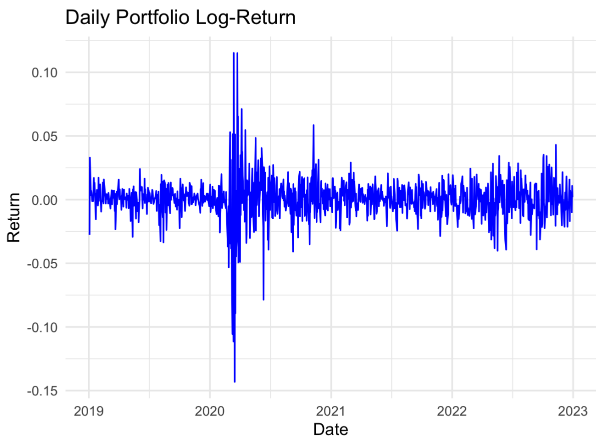

Figure 5: Daily log-return of the equally weighted portfolio. Major spikes in volatility occurred in early 2020.

5. VaR Backtesting on 2023

To assess the accuracy of the risk estimate, we tested how often the 2023 portfolio returns fell below the empirical VaR threshold.

$$I_t = \mathbb{1}\left\{R_t^{\text{port}} < \text{VaR}_{\alpha}\right\}$$

This is significantly below the 5% level, suggesting that the VaR model was conservative in the post-COVID environment.

6. Dependence Analysis with Copulas

We transformed the return data of JPM, BAC, and GS into pseudo-observations, then fitted several copula models and compared them using the AIC criterion

Copula Model	**AIC**
Gaussian	-3219
t-Copula	$\mathbf{-3423}$
Clayton	-2499
Gumbel	-2993

Table 1: Model comparison: the t-Copula performs best according to AIC.

6.1 Additional Copula Analysis Using the `QRM` Package

To validate our previous multivariate results, we performed a focused bivariate copula analysis between **JPMorgan (JPM)** and **Bank of America (BAC)** using the `QRM` and `tseries` packages in R.

We transformed daily log-returns into pseudo-observations using the empirical distribution function (EDF), and fitted the following copula families:

- Gaussian Copula
- Student-t Copula
- Gumbel Copula (Archimedean)
- Clayton Copula (Archimedean)

The table below reports the log-likelihood values from maximum likelihood estimation:

Copula Model	**Log-Likelihood**
Gaussian	928
Student-t	**990**
Gumbel	952
Clayton	752

Table 2: Log-likelihood comparison of fitted bivariate copulas for JPM-BAC.

The Student-t copula achieved the highest log-likelihood, confirming its ability to model joint extreme events (tail dependence) between the two financial institutions.

We also calculated rank-based dependence measures:

- Spearman's $\rho = 0.91$
- Kendall's $\tau \approx 0.925$

These high values confirm a strong dependency structure, particularly during periods of market stress such as the COVID-19 crisis.

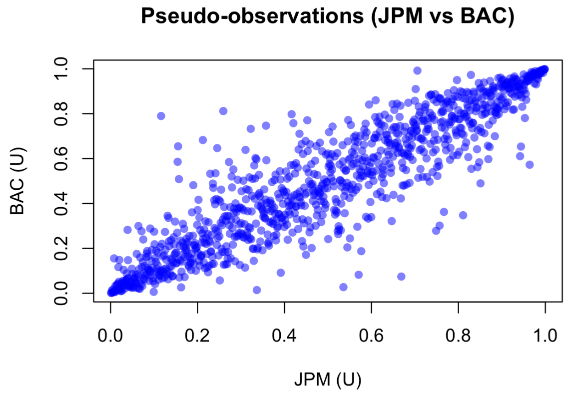

Figure 6: Pseudo-observations (EDF) for JPM vs BAC.

0.1 7. Backtesting the Empirical VaR: Kupiec Test Results

Backtesting the Empirical VaR: Kupiec Test Results

The Kupiec Proportion of Failures (POF) test assessed the accuracy of the VaR model set at 95% at 1 day during 2023. With only one breach in 251 trading days (0.4% compared to the expected 5%), the likelihood ratio was $LR_{\mathrm{POF}} = 11.83$ (p-value = 0.0006). The null hypothesis of correct coverage was rejected, indicating that the model significantly overestimates risk.

Conclusion

Our research confirms the COVID-19 crisis caused excessive volatility, high correlation spikes, and systemic risk—primarily in financials. Diversification worked well to minimize risk during non-stress times, but not so when there were stress times. Empirical VaR gave realistic estimates, and t-Copula was best positioned to model joint tail risk.

Script R

```
# --- Load libraries ---
library(tidyquant); library(dplyr); library(ggplot2); library(tidyr)

# --- 1. Select stocks (3 financial + 7 non-financial) ---
tickers <- c("JPM", "BAC", "GS", "AAPL", "MSFT", "XOM", "CVX", "WMT", "PG", "BA")

# --- 2. Download historical prices (2019{2022) ---
stock_data <- tq_get(tickers,
                     from = "2019-01-01",
                     to   = "2022-12-31",
                     get  = "stock.prices")
```

```
# --- 3. Compute daily log-returns ---
stock_returns <- stock_data %>%
  group_by(symbol) %>%
  arrange(date) %>%
  mutate(log_return = log(adjusted / lag(adjusted))) %>%
  filter(!is.na(log_return))

# --- 4. Plot adjusted prices ---
ggplot(stock_data, aes(x = date, y = adjusted, color = symbol)) +
  geom_line() +
  labs(title = "Adjusted Prices (2019{2022)", x = "Date", y = "Price") +
  theme_minimal()

# --- 5. Plot daily log-returns ---
ggplot(stock_returns, aes(x = date, y = log_return, color = symbol)) +
  geom_line() +
  labs(title = "Daily Log-Returns (2019{2022)", x = "Date", y = "Log-return") +
  theme_minimal()

# --- 6. Scatter plot: JPM vs BAC (same sector) ---
jpm_bac <- stock_returns %>%
  filter(symbol %in% c("JPM", "BAC")) %>%
  select(date, symbol, log_return) %>%
  pivot_wider(names_from = symbol, values_from = log_return) %>%
  filter(!is.na(JPM) & !is.na(BAC))

ggplot(jpm_bac, aes(x = JPM, y = BAC)) +
  geom_point(alpha = 0.5) +
  labs(title = "Scatter Plot: JPM vs BAC", x = "JPM", y = "BAC") +
  theme_minimal()

# --- 7. Scatter plot: JPM vs MSFT (different sectors) ---
jpm_msft <- stock_returns %>%
  filter(symbol %in% c("JPM", "MSFT")) %>%
  select(date, symbol, log_return) %>%
  pivot_wider(names_from = symbol, values_from = log_return) %>%
  filter(!is.na(JPM) & !is.na(MSFT))

ggplot(jpm_msft, aes(x = JPM, y = MSFT)) +
  geom_point(alpha = 0.5) +
  labs(title = "Scatter Plot: JPM vs MSFT", x = "JPM", y = "MSFT") +
  theme_minimal()

# --- 8. Portfolio return ---
portfolio_return <- stock_returns %>%
```

```
  group_by(date) %>%
  summarise(portfolio_return = mean(log_return, na.rm = TRUE)) %>%
  ungroup()

# --- 9. Value-at-Risk (VaR) ---
empirical_VaR <- quantile(portfolio_return$portfolio_return, probs = 0.05)
mu <- mean(portfolio_return$portfolio_return)
sigma <- sd(portfolio_return$portfolio_return)
normal_VaR <- mu - 1.645 * sigma

# --- 10. Plot portfolio returns ---
ggplot(portfolio_return, aes(x = date, y = portfolio_return)) +
  geom_line(color = "blue") +
  labs(title = "Daily Portfolio Log-Return", x = "Date", y = "Return") +
  theme_minimal()

# --- 11. Backtest with 2023 data ---
future_data <- tq_get(tickers,
                      from = "2023-01-01",
                      to   = "2023-12-31",
                      get  = "stock.prices")

future_returns <- future_data %>%
  group_by(symbol) %>%
  arrange(date) %>%
  mutate(log_return = log(adjusted / lag(adjusted))) %>%
  filter(!is.na(log_return))

future_portfolio <- future_returns %>%
  group_by(date) %>%
  summarise(portfolio_return = mean(log_return, na.rm = TRUE)) %>%
  ungroup()

# --- 12. VaR backtesting (violations) ---
violations <- mean(future_portfolio$portfolio_return < empirical_VaR)
violations

# --- 13. Copula analysis ---
copula_data <- stock_returns %>%
  filter(symbol %in% c("JPM", "BAC", "GS")) %>%
  select(date, symbol, log_return) %>%
  pivot_wider(names_from = symbol, values_from = log_return) %>%
  drop_na()

pseudo_obs <- apply(copula_data[, -1], 2, rank) / (nrow(copula_data) + 1)
```

```
gaussian_cop <- normalCopula(dim = 3, dispstr = "un")
fit_gauss <- fitCopula(gaussian_cop, pseudo_obs, method = "ml")

t_cop <- tCopula(dim = 3, dispstr = "un")
fit_t <- fitCopula(t_cop, pseudo_obs, method = "ml")

clayton_cop <- claytonCopula(dim = 3)
fit_clayton <- fitCopula(clayton_cop, pseudo_obs, method = "ml")

gumbel_cop <- gumbelCopula(dim = 3)
fit_gumbel <- fitCopula(gumbel_cop, pseudo_obs, method = "ml")

aics <- c(
  AIC(fit_gauss),
  AIC(fit_t),
  AIC(fit_clayton),
  AIC(fit_gumbel)
)
names(aics) <- c("Gaussian", "t-Copula", "Clayton", "Gumbel")
aics

# Load required packages
library(QRM)
library(tseries)

# 1. Select return data for JPM and BAC (two financial stocks)
copula_data <- stock_returns %>%
  filter(symbol %in% c("JPM", "BAC")) %>%
  select(date, symbol, log_return) %>%
  pivot_wider(names_from = symbol, values_from = log_return) %>%
  drop_na()

# 2. Convert to matrix and remove rows with zero returns in both assets
X <- as.matrix(copula_data[, -1])
X <- X[X[,1] != 0 & X[,2] != 0, ]

# 3. Create pseudo-observations using empirical distribution function (edf)
copulaX <- apply(X, 2, edf, adjust = 1)

# 4. Fit copulas
copulaXGauss <- fit.gausscopula(copulaX)          # Gaussian copula
copulaXt     <- fit.tcopula(copulaX)              # Student-t copula
copulaXGumb  <- fit.AC(copulaX, "gumbel")         # Gumbel copula (Archimedean)
copulaXClay  <- fit.AC(copulaX, "clayton")        # Clayton copula (Archimedean)
```

```
# 5. Compare log-likelihoods to select the best-fitting copula
copula <- c("Gaussian", "Student-t", "Gumbel", "Clayton")
logLik <- c(copulaXGauss$ll.max, copulaXt$ll.max, copulaXGumb$ll.max, copulaXClay$ll.max)
loglik_table <- data.frame(copula, logLik)

print(loglik_table)

# 6. (Optional) Compute Spearman's and Kendall's tau
print(Spearman(copulaX))
print(sin(pi * Kendall(copulaX) / 2))

# Pseudo-osservazioni (copulaX) già calcolate
plot(copulaX, main = "Pseudo-observations (JPM vs BAC)",
     xlab = "JPM (U)", ylab = "BAC (U)", pch = 16, col = rgb(0, 0, 1, 0.5))

# --- VaR Backtesting (Kupiec POF Test) ---

# Parameters
alpha <- 0.95
VaR_level <- 1 - alpha
T <- nrow(future_portfolio)
violations_vector <- future_portfolio$portfolio_return < empirical_VaR
x <- sum(violations_vector)         # Number of observed violations
p_hat <- x / T                      # Empirical failure rate

# Kupiec test statistic (Likelihood Ratio for Proportion of Failures)
LR_POF <- -2 * log(((1 - p_hat)^(T - x) * (p_hat^x)) / ((1 - VaR_level)^(T - x) * (VaR_level
p_value <- 1 - pchisq(LR_POF, df = 1)

# Output
cat("Kupiec POF Test:\n")
cat("Observed violations:", x, "out of", T, "days\n")
cat("LR_POF =", round(LR_POF, 3), "\n")
cat("p-value =", round(p_value, 4), "\n")

if (p_value < 0.05) {
  cat("The VaR model is statistically rejected at the 5% level\n")
} else {
  cat("The VaR model is accepted: no evidence of misspecification in violation rate\n")
}
\newpage

\section*{Appendix 1: Acknowledgement { The use of generative AI tools}
\addcontentsline{toc}{section}{Appendix 1: Acknowledgement { The use of generative AI tools}
```