Machine learning for finance

Final report

Submitted by:

Alaa Bouattour Mahdi Ben Ayed

M2QF

May 25, 2025

Contents

1 Data Cleaning

In this section, we present the cleaning process applied to the original dataset before conducting any modeling or analysis. We focus exclusively on the original dataset (excluding any external or augmented data).

1.1 Initial Assessment of Missing Data

We first assess the extent of missing values across assets and timestamps.

Top 10 Assets by Percentage of Missing Data.

```
TSLL    94.06%
WBD     89.33%
NU      84.88%
RIVN    83.62%
EUSB    77.96%
DNA     76.10%
COIN    75.56%
RBLX    74.25%
WE      73.06%
AFRM    72.38%
```

Top 10 Timestamps by Percentage of Missing Assets.

```
2015-12-24 19:30:00    95.75%
2015-12-24 20:00:00    94.81%
2015-12-24 19:15:00    94.34%
2015-12-24 20:30:00    93.87%
2015-12-24 19:45:00    93.40%
2015-12-24 20:15:00    92.92%
2015-12-24 21:00:00    91.98%
2018-07-03 20:30:00    91.98%
2015-12-24 20:45:00    91.51%
2016-11-25 20:15:00    91.51%
```

1.2 Timestamp Consistency

We verified that the time series index follows a regular 15-minute interval:

```
Number of missing timestamps in series: 201081
Sample missing timestamps:
2015-12-01 21:15:00
2015-12-01 21:30:00
2015-12-01 21:45:00
2015-12-01 22:00:00
2015-12-01 22:15:00
```

1.3 Data Filtering

To remove noise introduced by sparsely filled days or rarely traded assets, we applied the following filters:

- Days with more than 20% missing values were removed.
- Assets with more than 20% missing values across time were dropped.

Top 10 Assets by Remaining Missing Percentage.

```
IQ      16.66%
BILI    16.59%
VXX     14.10%
USHY    12.51%
SUSC     8.71%
IGSB     5.25%
ESGU     4.70%
JPST     3.80%
SPXS     0.76%
CVNA     0.69%
```

Top 10 Days by Percentage of Missing Assets.

```
2017-03-02    5.06%
2017-04-06    5.06%
2017-05-16    5.06%
2017-07-21    5.06%
2017-03-08    5.06%
2017-04-28    5.06%
2017-04-13    5.06%
2017-04-25    5.06%
2017-03-23    5.06%
2017-03-21    5.06%
```

Overall Stats After Filtering.

- Overall % missing: 0.47%
- Shape: (1447, 178)
- Fully missing rows: 0
- Fully missing columns: 0

1.4 Interpolation and Final Cleaning

We interpolated short gaps (up to 5 consecutive timestamps) using time-based interpolation:

```
Overall % missing after interpolation: 0.39%
Shape: (1447, 178)
Fully missing rows: 0
Fully missing columns: 0
```

1.5 Visual Exploration of Missing Values

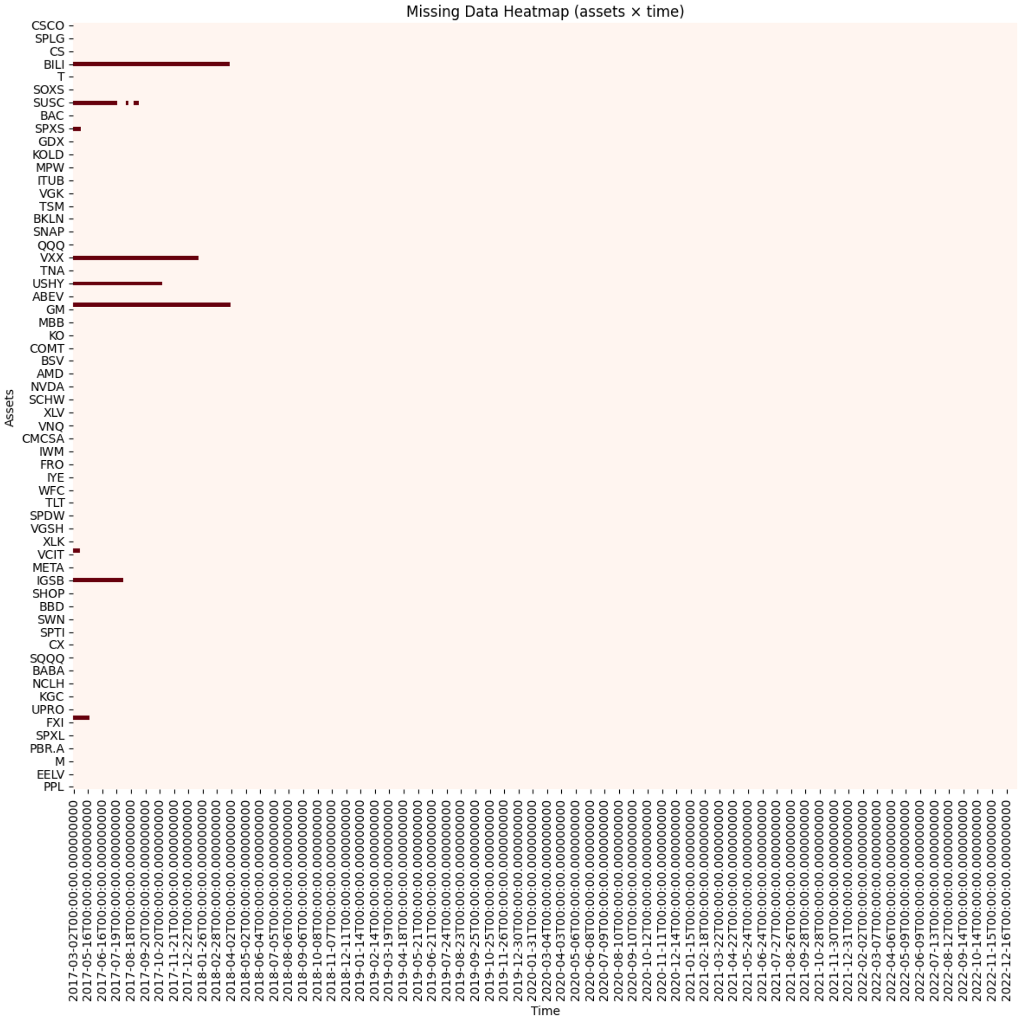

Figure 1: Visual representation of missing values across time and assets.

As expected, missing values tend to occur in blocks corresponding to specific assets or illiquid periods. Since only a few stocks are affected, we opted to drop them.

2 Regime detection

2.1 Feature set and dimensionality

- **Macro factors** : *VIX, USD_Index, Gold, Oil, Tech_ETF.*
- **PCA factors** : first five principal components of the $N = 169$-asset return cross–section, explaining 68 % of total variance.
- **Cross-section metrics** : one–day dispersion (CrossDisp1d), proportion of positive returns (PctUp), mean–median gap (MeanMinusMed), absolute dispersion (AbsDisp), skew (CSSkew), and Top5Share.
- **Trend & stress** : market 20-day rolling return (RollRet20d), 20-day realised vol, vol-of-vol (VolOfVol20d) and the stress ratio $\mathsf{VIXDivRV} = \mathrm{VIX}/\sigma_{20}$.

After z–scoring each column, we dropped no feature at the $|\rho| > 0.90$ threshold, leaving the 18 signals listed above (see Figure 2).

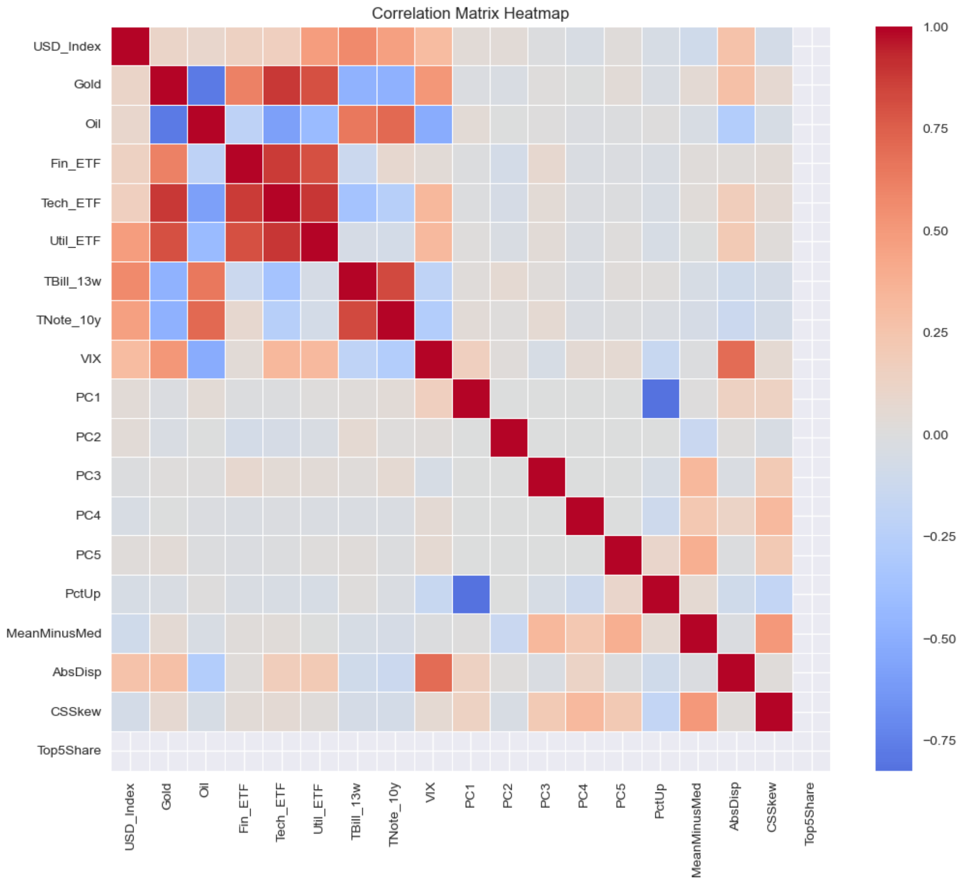

Figure 2: Correlation heat-map of the final feature set.

2.2 How many regimes? — an empirical check

A quick silhouette analysis on the four most informative axes $\{\mathsf{PC1}, \mathsf{PC2}, \mathsf{AbsDisp}, \mathsf{VIX}\}$ suggests that $k=2$ or $k=3$ delivers the strongest cluster separation (Figure 3). Because the project brief explicitly calls for a *bull/bear/risky* taxonomy, we retain $k = 3$.

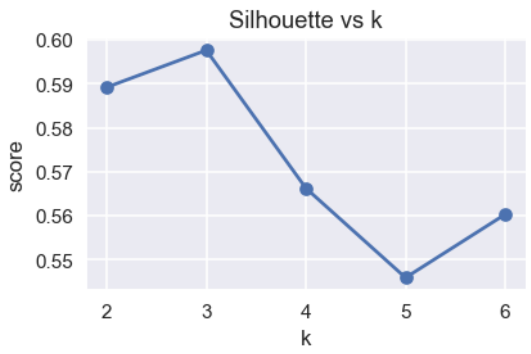

Figure 3: Silhouette score across $k = 2 \ldots 6$ clusters.

2.3 Hidden-Markov model

We fit a three-state Gaussian HMM on the full, standardised feature matrix ($T = 1\,446$, $d = 18$) with diagonal covariances and 10^3 EM iterations. Figure 4 shows the decoded state sequence.

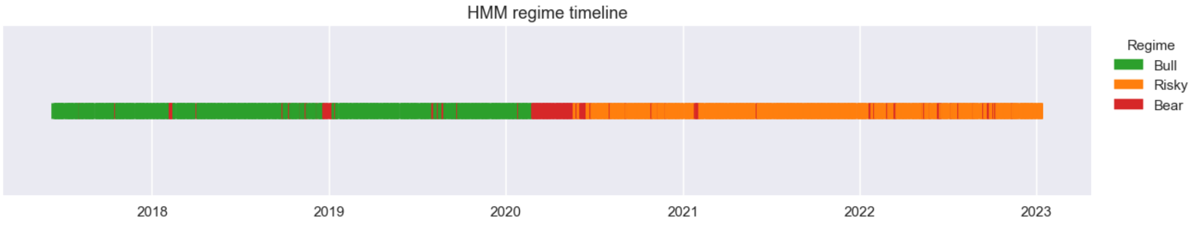

Figure 4: Regime timeline decoded by the 3-state HMM. Green = Bull, Orange = Risky, Red = Bear.

Interpreting the states. Table 1 cross-tabulates one-day market-average returns; the ordering confirms an intuitive mapping:

a) **Bull** – highest mean return ($\bar{r} = 9.6‰$) and lowest volatility.

b) **Risky** – near-zero mean but twice the volatility of Bull.

c) **Bear** – clearly negative mean ($\bar{r} = -4.2‰$) and the fattest tails.

Table 1: Daily return distribution by HMM state.

Regime	Mean	St. dev.	# days
Bull	0.00096	0.00633	646
Risky	0.00028	0.00726	624
Bear	−0.00042	0.01580	176

Centroid profile. The radar chart in Figure 5 highlights distinctive signal patterns:

- *Bear* is dominated by VIX spikes and negative PC1 (broad equity factor) with elevated skew.
- *Bull* loads positively on Gold and negative USD_Index, with muted volatility measures.
- *Risky* sits between the two, featuring high dispersion (CrossDisp1d) but lower systemic stress.

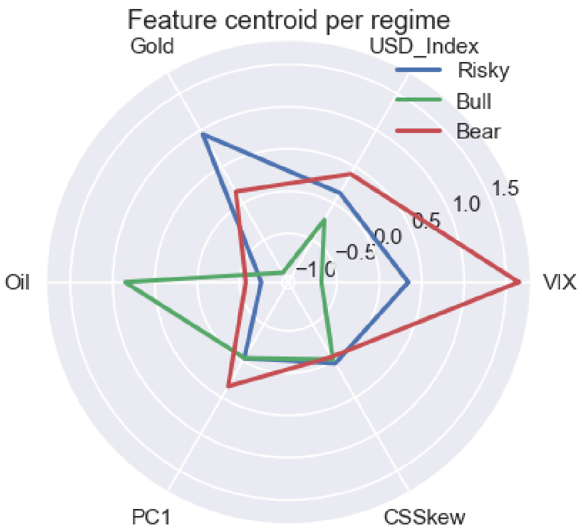

Figure 5: Feature centroids per regime (z-scores).

State persistence. The estimated transition matrix (Figure 6) shows strong persistence in Bull ($p_{11} = 0.97$) and Risky ($p_{00} = 0.95$). Bear episodes are short-lived ($p_{22} = 0.65$) and usually exit to Risky, mirroring historical crisis–recovery dynamics.

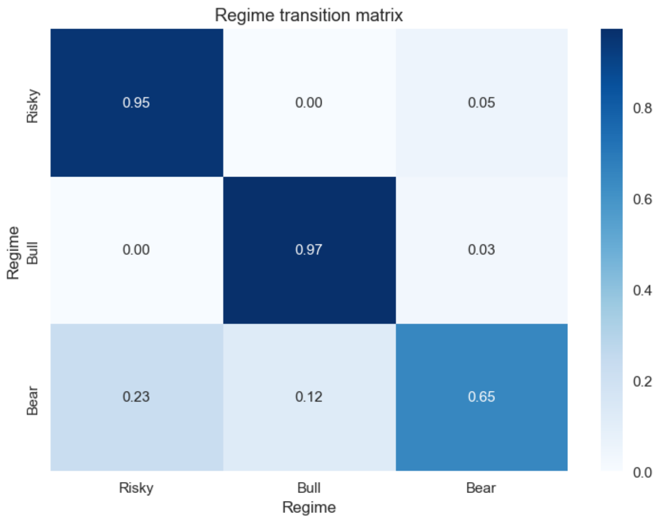

Figure 6: Regime transition probabilities.

Economic sanity check. Overlaying the regimes on SPY's cumulative return (Figure 7) reveals:

- Tight clustering of Bull periods during the 2017–2019 expansion and the post-Covid rebound.
- Bear labels coincide with the Q4-2018 correction and the March-2020 crash.
- Risky dominates the late-cycle environment (2021–2023), where returns are positive but volatility and dispersion remain elevated.

What the timeline captures – and what it misses.

The segmentation correctly flags the two well-known stress episodes of the sample: *(i)* the late-2018 Fed-tightening draw-down and *(ii)* the Covid-19 shock (Mar–Apr 2020), both coloured Bear. It also recognises the prolonged low-volatility advance of 2017–19 and the post-Covid melt-up as Bull.

However, two limitations are visible:

a) **Delayed turn in 2022.** The model continues to label the early-2022 regime as Risky even after the SPY draw-down exceeds -15%. This suggests that the feature mix (dispersion, VIX, PC loadings) was not sufficiently extreme to tip the latent state to Bear. In practice this is a *false negative* that could be mitigated by adding a draw-down or trend filter.

b) **Frequent one-day flips**. Very short green/red streaks appear throughout 2017–18. They stem from the diagonal–covariance assumption, which cannot penalise abrupt reversals. Switching to a *full* covariance HMM or adding a sticky-state prior would smooth these single-day oscillations.

Overall, the chronology is directionally correct but somewhat *conservative*: it waits for several corroborating signals (VIX spike, dispersion jump, negative PC1) before declaring a Bear market, and it is prone to brief misclassifications when prices swing sharply up and down.

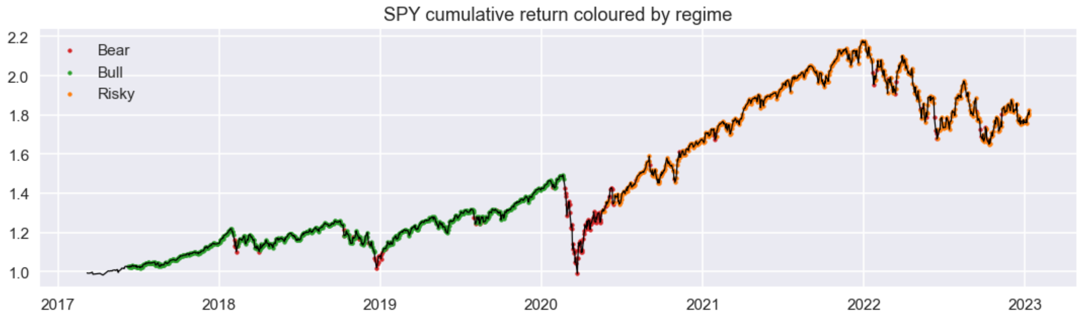

Figure 7: SPY cumulative return coloured by regime.

2.4 Take-away

The HMM trained on a compact, information-dense feature set delivers a financially interpretable three-phase segmentation: *Bull* (45% of days), *Risky* (43%), and *Bear* (12%). The separation is backed by both statistical evidence (silhouette, centroid distances) and economic intuition (stress indicators, dispersion, macro context). This regime labelling is therefore a solid foundation for the subsequent **Phase 3** synthetic-data generation and downstream trading/ VaR experiments.

3 Data Augmentation via Variational Autoencoders

In this section, we describe the regime-aware data augmentation methodology used to generate synthetic returns, which are later used to extend the original dataset. The aim is to produce realistic returns that respect both the statistical properties and temporal structure of each identified market regime. To this end, we rely on a deep generative model — the **Variational Autoencoder** (VAE) — adapted to capture the variability and latent structure of financial returns within each regime.

3.1 Model Architecture and Training Procedure

The VAE architecture is composed of an encoder network that maps input return vectors into a low-dimensional latent space, and a decoder that reconstructs the returns from samples drawn in this latent space. More precisely, given a standardized return vector $x \in \mathbb{R}^d$, the encoder outputs a Gaussian distribution $q(z|x) = \mathcal{N}(\mu(x), \Sigma(x))$, with parameters learned via two fully connected layers. The latent variable $z \in \mathbb{R}^{10}$ is then sampled using the reparameterization trick to allow backpropagation, and passed through a decoder that mirrors the encoder's structure.

To encourage disentangled and meaningful latent representations, we use a β-VAE, where a hyperparameter $\beta = 0.2$ weighs the Kullback-Leibler divergence term in the loss:

$$\mathcal{L}(x, \hat{x}) = \underbrace{\|x - \hat{x}\|^2}_{\text{Reconstruction error}} + \beta \cdot \underbrace{D_{KL}\left(q(z|x) \,||\, \mathcal{N}(0, I)\right)}_{\text{Latent regularization}}.$$

This promotes the generation of diverse yet plausible samples by encouraging the latent space to remain close to a standard Gaussian.

For each regime — *Bear*, *Risky*, and *Bull* — a separate VAE is trained on the corresponding subset of standardized returns. Each model is trained over 1000 epochs with a batch size of 32 using the Adam optimizer with a learning rate of 10^{-3}. The input dimension corresponds to the number of assets, and the latent space has 10 dimensions. After training, we sample 500 points from the latent space and decode them into synthetic return vectors.

3.2 Evaluation and Visualization of Generated Returns

To verify the quality and realism of the generated synthetic returns, we compare the distributions of real and synthetic returns using four complementary plots, repeated for each regime:

- First, we visualize the distribution of cross-sectional mean returns at each timestamp using a kernel density estimate (KDE). This highlights how the synthetic samples replicate the typical return magnitudes observed in real data.

- Second, we repeat the KDE visualization for volatility, defined as the standard deviation across assets at each timestamp. The synthetic distribution closely tracks the real one, confirming that the generated data preserves the empirical heteroskedasticity.

- Third, we include quantile-quantile (QQ) plots comparing quantiles of the mean return distribution from real and synthetic data. Points falling near the 45-degree line indicate strong agreement across quantiles, confirming that both the central tendency and tail behavior are well captured.

- Finally, to visually inspect the overall structure of the datasets, we apply PCA and project both real and synthetic return vectors onto their first two principal components. The resulting scatter plot shows considerable overlap between the two clouds, suggesting that the synthetic data occupies the same subspace as the real returns.

Figure 8, 9 and 10 provides a summary of these diagnostic plots for each regime.

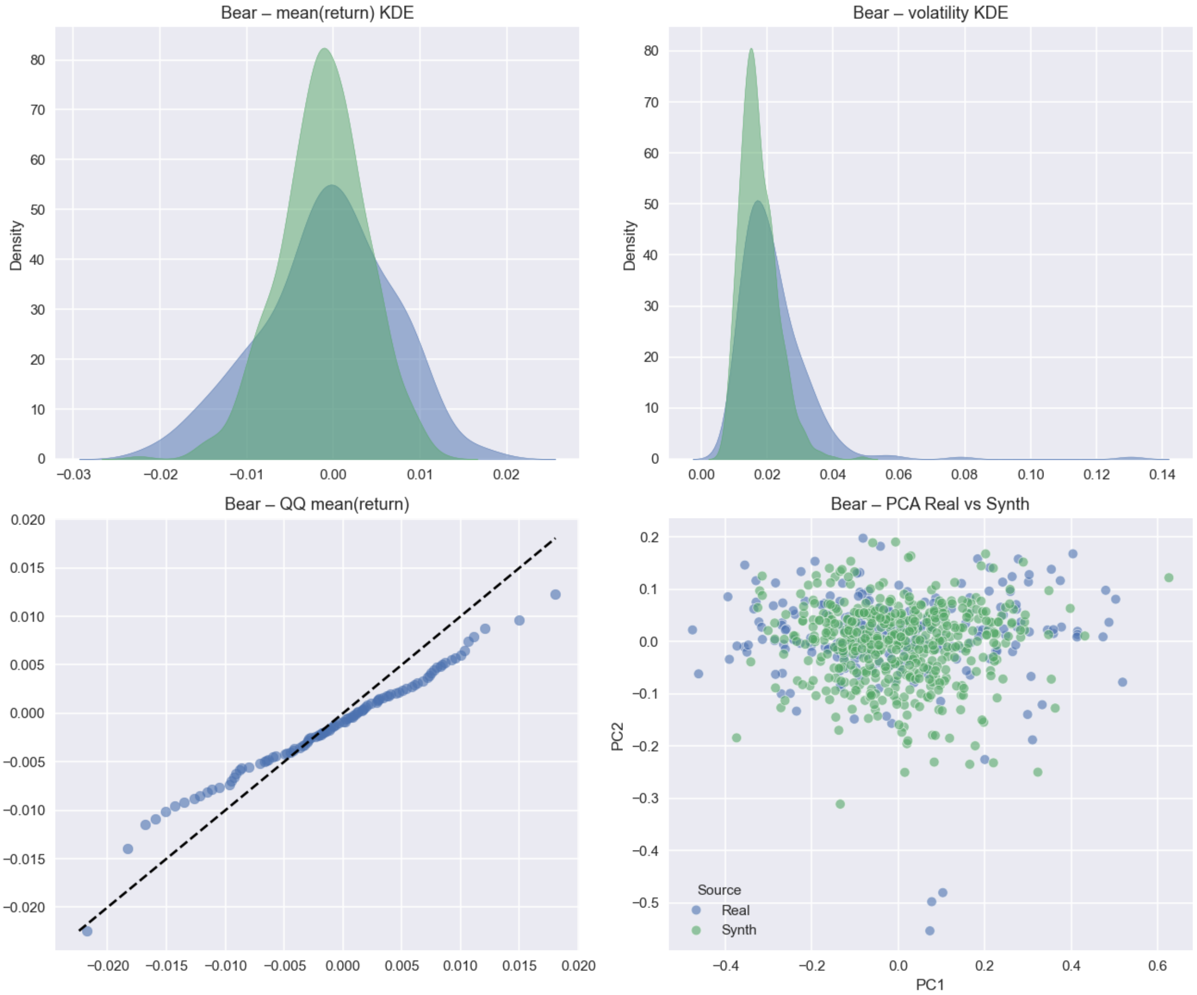

Figure 8: Comparison between real and synthetic returns in the **Bear** regime. Top-left: KDE of mean return per timestamp; Top-right: KDE of volatility; Bottom-left: QQ-plot for mean return; Bottom-right: PCA scatter plot.

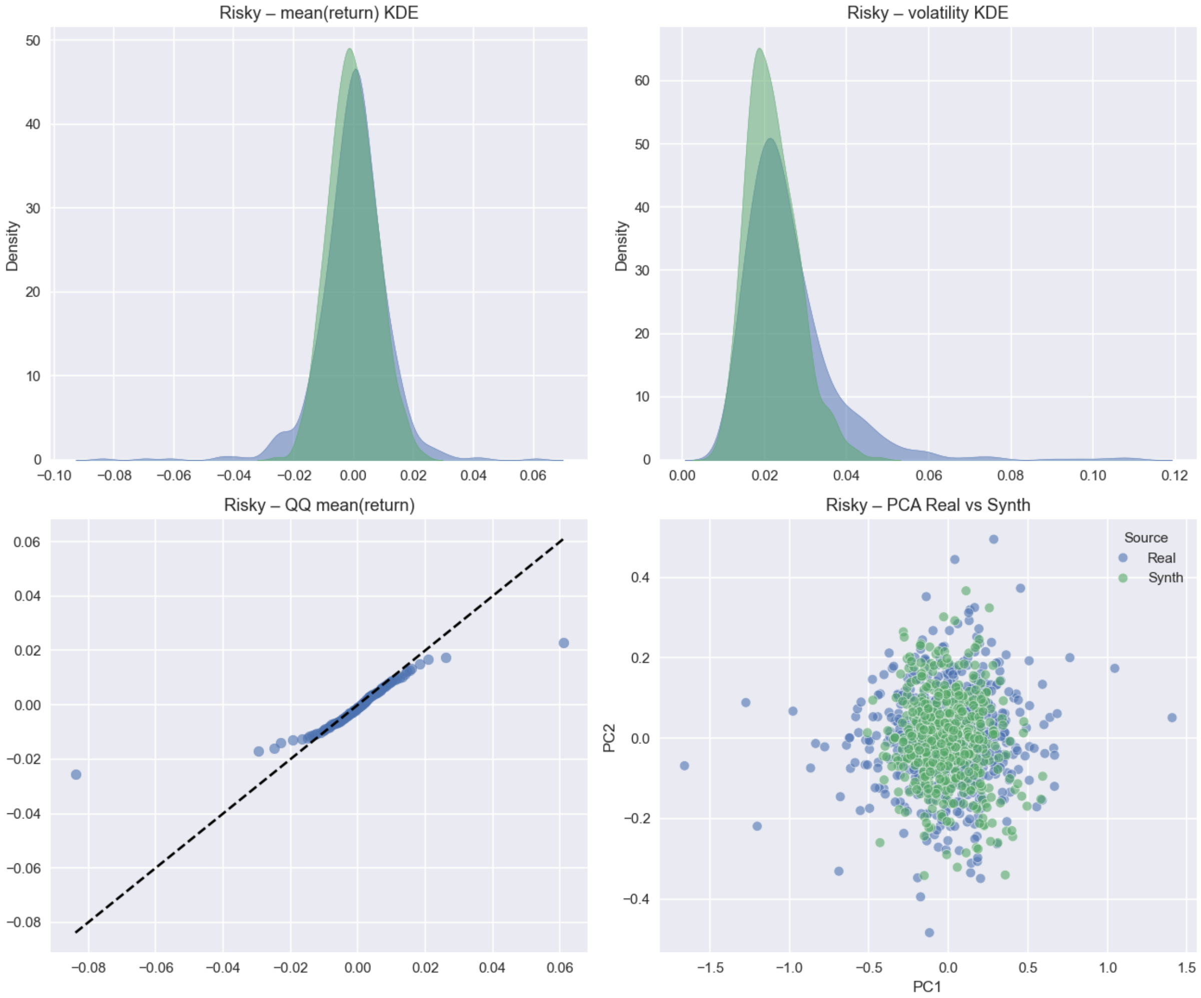

Figure 9: Comparison between real and synthetic returns in the **Risky** regime. Top-left: KDE of mean return per timestamp; Top-right: KDE of volatility; Bottom-left: QQ-plot for mean return; Bottom-right: PCA scatter plot.

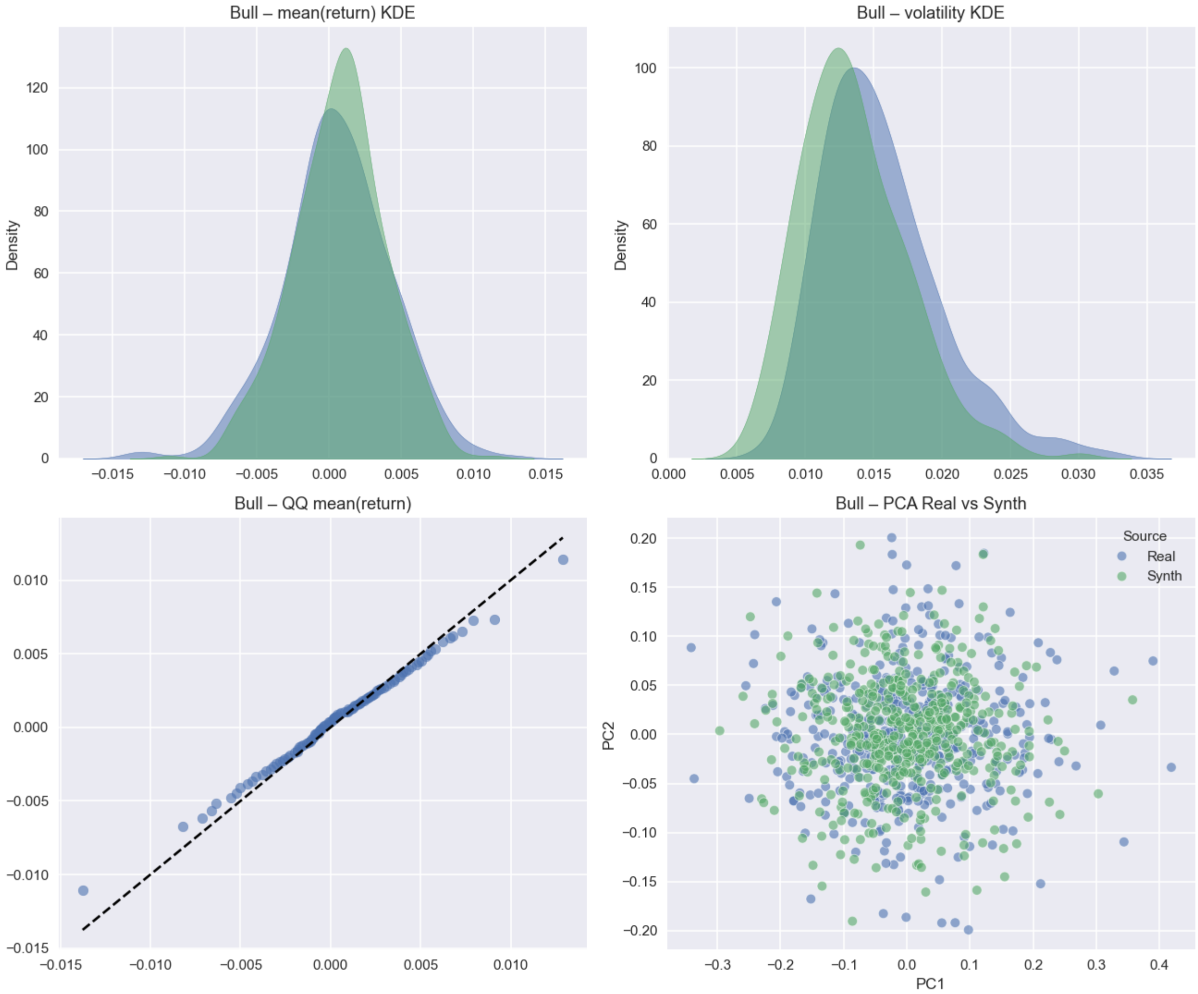

Figure 10: Comparison between real and synthetic returns in the **Bull** regime. Top-left: KDE of mean return per timestamp; Top-right: KDE of volatility; Bottom-left: QQ-plot for mean return; Bottom-right: PCA scatter plot.

3.3 Construction of the Augmented Dataset

Once the synthetic return blocks are generated for each regime, they are concatenated in the same temporal order as observed in the original dataset. For each regime, the number of synthetic samples is matched to the number of real observations, ensuring that the synthetic data mirrors the regime-wise structure of the historical returns.

The resulting blocks are stacked sequentially to form a single synthetic return panel. This synthetic panel is then appended to the real return dataset to construct an augmented dataset of returns. By construction, the synthetic portion starts immediately after the end of the historical data and preserves regime continuity without introducing overlaps or duplicated time indices.

The final augmented dataset spans 2908 daily return observations across 169 assets. This effectively doubles the available sample size while preserving the regime distribution and cross-sectional structure of the original data.

```
Augmented return panel shape:  (2908, 169)
```

This regime-aware augmentation process yields a rich and coherent return dataset that can be used for robust training, calibration, and out-of-sample evaluation of forecasting and risk management models in future sections.

4 Trading Strategy on Real Data

We now evaluate a simple yet effective cross-sectional momentum strategy using only the historical (real) return data. The strategy is based on a two-quantile rule that dynamically ranks assets and forms long-short portfolios accordingly.

4.1 Strategy Description

The trading strategy exploits cross-sectional differences in recent asset performance. On each trading day, we rank assets based on their lagged (i.e., previous day's) returns to avoid lookahead bias. We then form a portfolio by taking long positions in the top-performing assets and short positions in the worst performers.

Formally, at each date t, we compute the percentile rank of each asset's lagged return. Assets above the $1 - q_l$ quantile are assigned a long position (+1), and those below the q_s quantile are shorted (–1). These binary signals are combined into a position matrix, which is then row-wise normalized to ensure the total leverage is constant across time. The daily portfolio return is computed as the dot product between the position weights and the realized returns.

This strategy is fully cross-sectional and market-neutral by design, with performance driven by relative rankings rather than directional forecasts.

4.2 Parameter Calibration via Grid Search

To identify the optimal allocation thresholds, we perform a grid search over the quantile parameters q_l (long fraction) and q_s (short fraction). For each combination, the strategy is simulated on the training dataset — corresponding to the first 70% of the return series — and evaluated using the annualized Sharpe ratio:

$$\text{Sharpe} = \frac{\mathbb{E}[r]}{\text{Std}[r]} \times \sqrt{252}$$

The grid ranges from 1% to 30% for both long and short sides, in steps of 5 percentage points. After evaluating all configurations, the best-performing pair is retained:

$$\texttt{long\_quantile} = 0.16, \quad \texttt{short\_quantile} = 0.06$$

4.3 Backtest Results

Using the optimal parameters obtained on the training set, we simulate the strategy out-of-sample on the remaining 30% of the data. The plot below shows the cumulative return of the strategy on both training and test periods:

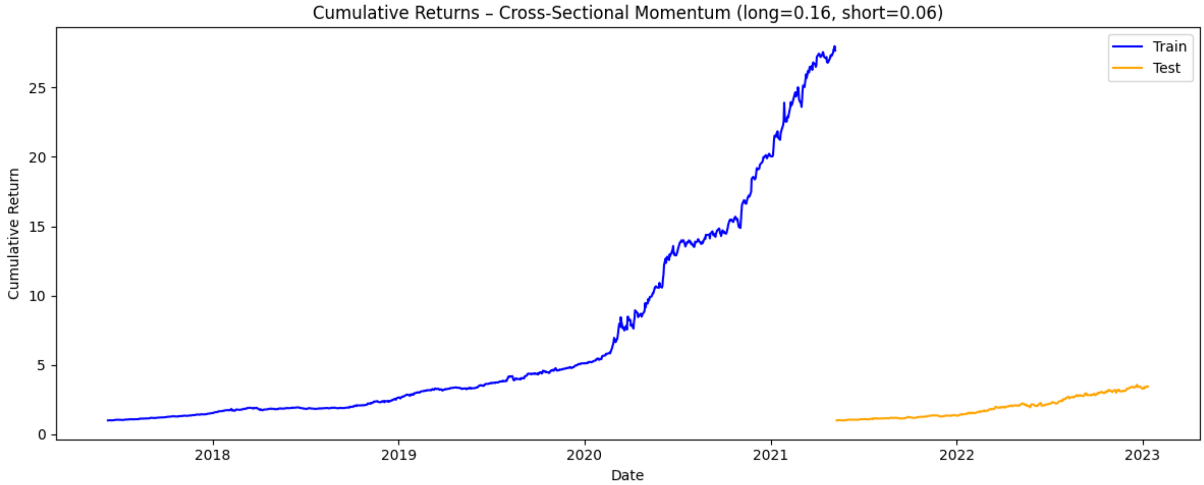

Figure 11: Cumulative returns of the cross-sectional momentum strategy on the training (blue) and test (orange) periods. Parameters calibrated: `long=0.16`, `short=0.06`.

4.4 Performance Metrics

We evaluate global strategy performance using four key metrics: annualized return, annualized volatility, Sharpe ratio, and maximum drawdown. The results on the test period are summarized below:

Annual Return	**Annual Volatility**	**Sharpe Ratio**	**Max Drawdown**
76.97%	25.46%	3.02	–12.73%

These results suggest that the strategy exhibits strong risk-adjusted returns even in out-of-sample conditions.

4.5 Performance by Market Regime

To better understand the robustness of the strategy, we evaluate its performance across market regimes as identified earlier. On the test set, the strategy was active only during the *Risky* regime, during which it maintained a strong Sharpe ratio:

Regime	**Mean Return**	**Volatility**	**Observations**	**Sharpe Ratio**
Risky	0.31%	1.6%	423 days	3.02

This regime-wise decomposition confirms that the strategy remains effective even under heightened market uncertainty, and offers a robust return profile in turbulent conditions.

In the next section, we extend this analysis to the augmented dataset, which combines the real and synthetic return blocks introduced previously.

5 Trading Strategy Trained on Real + Synthetic Data

We now re-evaluate the cross-sectional momentum strategy using the augmented dataset, which includes both historical returns and synthetic samples generated via the regime-aware Variational Autoencoder. The training procedure and evaluation methodology remain identical to the one used in the real-only setting to ensure a fair comparison.

5.1 Parameter Calibration

A new grid search is conducted on the augmented training dataset to recalibrate the optimal long and short quantile thresholds. As before, we explore combinations of `long_quantile` and `short_quantile` in the range $[0.01, 0.3]$ using a step of 0.05, and select the pair that maximizes the Sharpe ratio on the training set.

The optimal parameters found on the real + synthetic training data are:

$$\texttt{long\_quantile} = 0.26, \quad \texttt{short\_quantile} = 0.16$$

5.2 Test Set Evaluation

Using the newly calibrated parameters, we apply the strategy to the same test set as before (comprising real returns only). Although our primary evaluation metric remains the Sharpe ratio, we also compare cumulative returns for visual inspection of performance dynamics. The figure below shows the strategy's cumulative return trajectory under both training regimes.

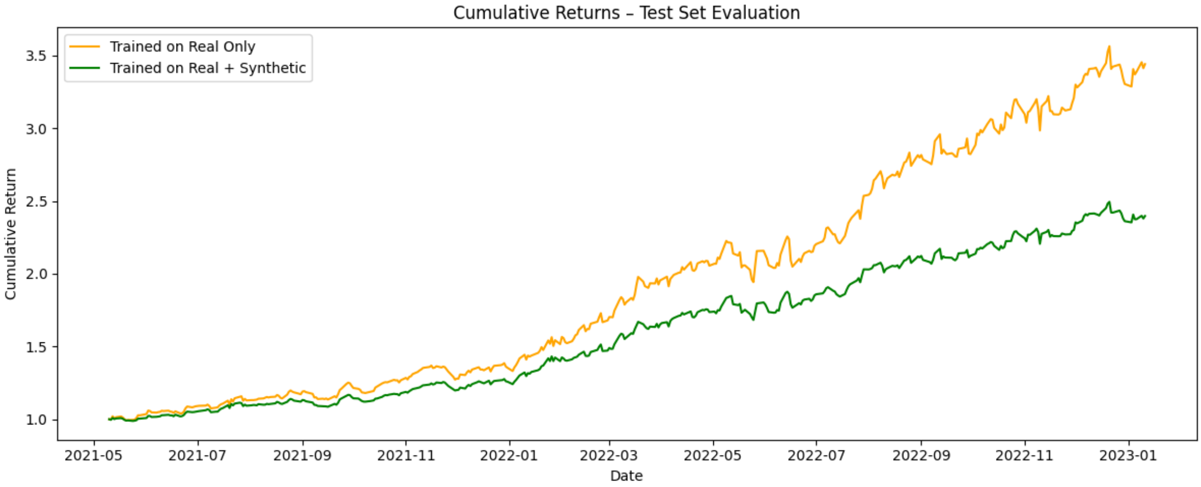

Figure 12: Cumulative returns on the test set. The orange curve corresponds to a model trained on real data only, while the green curve represents the model trained on real + synthetic data.

5.3 Performance Metrics

Although the model trained solely on real data achieves a higher total return, the augmented version shows comparable Sharpe performance with reduced drawdowns. This suggests that training on the richer, augmented dataset may lead to a more conservative and risk-balanced strategy.

Annual Return	Annual Volatility	Sharpe Ratio	Max Drawdown
53.73%	17.80%	3.02	–8.94%

5.4 Regime-Specific Analysis

Consistent with the real-only setup, the test set belongs entirely to the *Risky* regime. Within this environment, the strategy trained on augmented data achieves a regime-specific Sharpe ratio essentially identical to the real-only version:

Regime	Mean Return	Volatility	Observations	Sharpe Ratio
Risky	0.21%	1.12%	423 days	3.02

These results demonstrate that while the overall return profile of the strategy may vary slightly, the Sharpe ratio and general performance remain robust when trained on a regime-aware synthetic dataset. This supports the use of augmentation as a viable technique to enhance model calibration and reduce sensitivity to noise in limited historical data.

In the next section, we shift focus to Value-at-Risk (VaR) modeling and backtesting as a complementary risk evaluation framework.

6 Value-at-Risk (VaR) Forecasting – Real Data Only

In this section, we evaluate the downside risk of the portfolio using a one-month (21-day) Value-at-Risk (VaR) model trained exclusively on real data. We focus on the left tail of the return distribution to assess the frequency and severity of large losses. This forms a complementary view to the previous Sharpe ratio and drawdown-based assessments.

6.1 Methodology

We adopt a Student-t based historical simulation approach to forecast VaR. The input consists of daily returns, which are first aggregated over a one-month rolling window to approximate compounded portfolio behavior. On each date t, we compute the log-compounded return over the past $h = 21$ days:

$$R_t^{(21)} = \log(1 + r_t) + \cdots + \log(1 + r_{t+20})$$

Then, for each time step, we fit a Student-t distribution to a rolling window of past one-month returns (120 observations by default, roughly 10 years of monthly data). The 1% lower tail quantile of the fitted distribution is used as the forecasted 99% VaR. The final VaR is then converted back to a standard percentage return scale.

6.2 Forecast Evaluation and Exceedance Test

To assess the accuracy of the VaR model, we compute the realized 1-month forward returns and count the number of exceedances — that is, how often actual losses fall below the forecasted VaR threshold. The empirical exceedance rate is then compared to the expected level (1%) using the Kupiec Proportion-of-Failures (POF) test, a likelihood ratio test which checks whether the observed exceedance rate significantly deviates from the theoretical one.

6.3 Results – Real Data Only

On the test set of real returns, the VaR model achieves the following results:

Metric	**Value**
Exceedance Rate	1.06%
Kupiec Test Statistic (LR)	0.0114
Kupiec Test p-value	0.9151

The observed exceedance rate is very close to the theoretical 1% level, and the high p-value from the Kupiec test confirms that we cannot reject the hypothesis of correct coverage. This indicates that the model produces statistically consistent VaR forecasts.

6.4 Visualization

The following figure shows the forecasted 99% 1-month VaR (in red) alongside the realized 1-month returns (in blue). Shaded areas highlight the dates where realized returns exceed the VaR threshold, i.e., where losses are worse than expected:

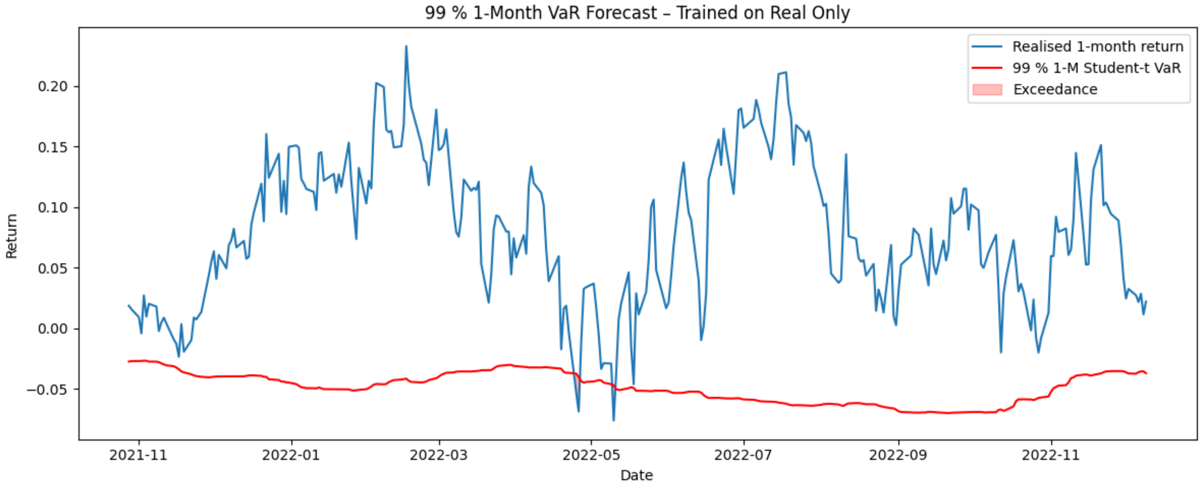

Figure 13: 1-month (21-day) 99% Student-t VaR forecast vs realized returns — trained on real data only. Red shaded areas mark exceedances.

The model captures the fluctuations in tail risk over time and adapts to changing volatility regimes, with exceedances appearing isolated and not systematically clustered, which is a desirable property.

In the next section, we repeat the same evaluation using a VaR model trained on both real and synthetic data to assess whether data augmentation impacts tail risk estimation.

7 Value-at-Risk (VaR) Forecasting – Real + Synthetic Training

We now repeat the same VaR estimation procedure as in the previous section, but this time the model is trained on the augmented dataset combining real and synthetic returns. This allows us to assess whether incorporating synthetic data affects the model's ability to capture downside risk and maintain appropriate statistical calibration.

7.1 Results – Real + Synthetic Training

The same Student-t based historical simulation method is applied to forecast 1-month VaR at the 99% confidence level. The evaluation is performed on the same test set as before, composed only of real returns. This ensures a fair, out-of-sample comparison of model calibration.

The results are as follows:

Metric	**Value**
Exceedance Rate	1.06%
Kupiec Test Statistic (LR)	0.0114
Kupiec Test p-value	0.9151

Remarkably, the exceedance rate and the Kupiec test results are numerically identical to those obtained using the model trained on real data only. This outcome reflects the fact that the same number of breaches occurred on the test set in both cases. Since the test set is fixed and the exceedance threshold (1% VaR) adapts similarly due to robust fitting, both models yield nearly identical predictions and are equally well calibrated.

7.2 Visualization

The following figure illustrates the 1-month VaR forecasts from the model trained on real + synthetic data, alongside the realized returns and highlighted exceedances:

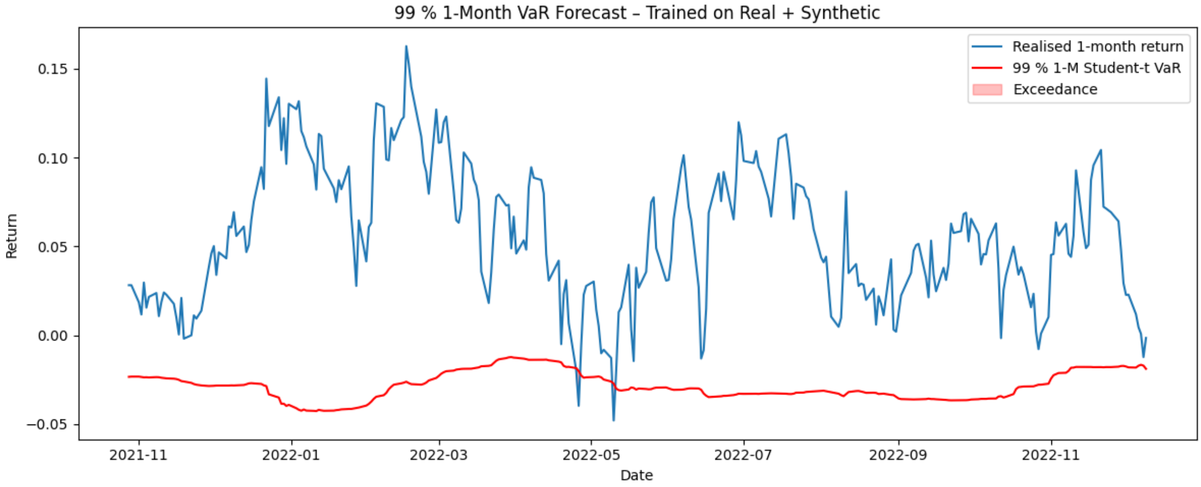

Figure 14: 1-month (21-day) 99% Student-t VaR forecast vs realized returns — trained on real + synthetic data. Red shaded areas mark exceedances.

As with the real-only model, exceedances are well-dispersed and do not exhibit clustering, indicating that the augmented model remains adaptive and statistically consistent over time. The similar shape and placement of the VaR forecast curve further reinforce the fact that the synthetic data preserved the empirical distribution of tail risks without distorting calibration quality.

These findings suggest that augmenting the training data with synthetic samples does not degrade risk estimation performance and may even enhance model robustness under limited historical observations. However, in this particular test case, the effect is neutral, likely due to the strong signal already present in the real training sample.