Scenario Forecasting of the U.S. Treasury Yield Curve: A Comparative Analysis of the Nelson-Siegel Model, Principal Component Analysis, and the Autoencoder Method

by

Sarah Leeser [884242]

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in

Econometrics and Mathematical Economics

Tilburg School of Economics and Management
Tilburg University

Supervisors

Dr. Valentina Melentyeva
Dr. Christoph Walsh
Dr. Francois Myburg (Van Lanschot Kempen)

Date: July 1, 2025

Acknowledgements

I would like to thank Dr. Francois Myburg at Van Lanschot Kempen for his valuable guidance, thoughtful suggestions, and consistent support throughout this project. His insights were instrumental in shaping the direction of this thesis. I am also grateful to Dr. Valentina Melentyeva for her helpful feedback and encouragement during the research process.

Additionally, I would like to thank my colleagues at Van Lanschot Kempen for the enjoyable interactions and supportive atmosphere during the writing process, which provided me with energy and motivation.

Abstract

Accurate scenario forecasting for the U.S. Treasury yield curve is essential for applications in risk management, monetary policy, and financial forecasting. This study compares three models, Nelson-Siegel, Principal Component Analysis (PCA), and Autoencoders (AE) for generating plausible U.S. Treasury yield curve scenarios. Using Monte Carlo simulations and a rolling forecast framework (1990–2023), each model is evaluated on two criteria: predictive accuracy (via Continuous Ranked Probability Score, CRPS) and the reliability of simulated intervals (via Prediction Interval Coverage Probability, PICP). Additionally, we assess whether the simulated residuals preserve the historical dependence structure.

Each model is retrained across multiple historical windows and used to simulate interest rate scenarios for each quarter of 2024, enabling a robustness analysis over time.

The Autoencoder consistently outperforms the other models, achieving the lowest CRPS and highest PICP in most periods. PCA performs better than Nelson-Siegel in terms of CRPS, while Nelson-Siegel leads in interval coverage. All models perform worst in Q3 and when trained on more recent or shorter samples.

These findings underscore the benefits of non-linear models for capturing complex yield curve dynamics. While AE is most suited for predictive tasks, PCA and Nelson-Siegel remain valuable for their interpretability and transparency.

Contents

1 Introduction

The yield curve is widely regarded as a key indicator in financial markets, as it offers insight into how changes in policy rates and broader monetary instruments are transmitted to a broad spectrum of interest rates in the economy (Reserve Bank of Australia, n.d.). Understanding and forecasting this curve is particularly important for institutions active in fixed income portfolios, where decision-making is highly dependent on interest rate expectations. Interest rate forecasts provide valuable insight into the economic outlook, including the policy stance of the U.S. Federal Reserve, the likelihood of a recession, and expected changes in inflation or economic growth (Estrella and Mishkin, 1996). Forecasts across different maturities are important for valuing fixed income securities, managing interest rate risk, and supporting macroeconomic decision-making. For financial institutions such as Van Lanschot Kempen, robust scenario generation also offers practical value for risk assessment and portfolio stress testing, making it a highly relevant area of research. Traditional methods for modelling the term structure of interest rates, such as Nelson and Siegel (1987) framework and Principal Component Analysis (PCA) (Redfern and McLean, 2014), have become standard tools due to their intuitive structure and strong empirical performance in explaining interest rate variation across maturities.

However, most existing studies focus on predicting the expected future interest rate at each maturity, rather than generating full interest rate scenarios. This is a key limitation for risk management applications, where realistic scenario generation is more informative than single-point forecasts. Although Nelson-Siegel and Principal Component Analysis are well-established techniques for dimensionality reduction, their application to full scenario simulation and calibrated probabilistic forecasts remains relatively underexplored.

Recent developments in machine learning offer more flexible alternatives. One such method is the Autoencoder (AE): a neural network architecture that learns a compact, non-linear representation of input data through unsupervised learning (Bank et al., 2021). Though promising in theory, its application in the context of yield curve simulation remains limited in the academic literature.

This leads to the following central research question:
How do Nelson-Siegel, Principal Component Analysis, and Autoencoders perform in generating plausible scenarios for the Treasury Yield Curve?

The objective of this research is to evaluate three modelling approaches using a parametric Nelson-Siegel model, a linear PCA model, and a non-linear autoencoder for their ability to generate interest rate scenarios. Rather than assessing realism in a qualitative sense, we focus on two quantitative dimensions: predictive accuracy and the statistical coverage of observed values. All models are implemented within a consistent simulation framework

and evaluated across multiple economic periods to test their robustness. While we expect the autoencoder to better capture non-linear structure, particularly in volatile conditions, the overall setup allows for a transparent comparison across methods.

The first chapters of this thesis highlight the relevance of the topic through a literature review on existing methods and research in the field of yield curve modelling (see Section 2). In Section 3, the dataset is described along with the steps of data preprocessing. The methodology, outlined in Section 4, covers the dimensionality reduction techniques (Nelson-Siegel, PCA, and Autoencoders), the forecasting procedure, and the evaluation metrics used.

Finally, Section 6 discusses the empirical results in relation to the research question and provides practical recommendations, while Section 7 summarizes the main findings and reflects on their broader implications.

2 Literature Review

While the importance of interest rate forecasting is widely recognized, much of the academic literature has focused on point predictions at individual maturities. Yet it is not only important to predict a single expected value of interest rate, but rather to model the entire distribution of future rates (Gneiting and Katzfuss, 2014). This is crucial for pricing interest rate derivatives and conducting comprehensive risk management.

Despite decades of research on the term structure of interest rates, relatively few studies systematically evaluate the actual forecasting performance of yield curve models (Diebold and Li, 2006). Most research emphasizes in-sample fit, that is, how well a model describes historical data rather than its out-of-sample predictive ability. When forecasting performance is evaluated, as in Duffee (2002) and Diebold and Li (2006), traditional models are often found to fall short.

Two widely used approaches in the literature are Nelson-Siegel (NS) and Principal Component Analysis (PCA). Both aim to reduce the complexity of the yield curve using latent factors. The NS model is valued for its simplicity and its ability to capture commonly observed yield curve shapes, such as monotonic, humped, and S-shaped curves. Originally introduced to explore whether a simple model could explain the relationship between maturity and yield for U.S. Treasury securities (Nelson and Siegel, 1987), subsequent work by Diebold and Li (2006) showed that the three factors, typically interpreted as long-, medium-, and short-term components, can be understood as level, slope, and curvature. Their evaluation of the model's forecasting ability revealed that while the NS model does not outperform benchmarks at short horizons, its one-year-ahead forecasts are significantly better. Moreover, more complex extensions of the model did not improve forecasting performance, underscoring the importance of parsimony for out-of-sample prediction.

PCA, by contrast, is a statistical technique used to reduce the dimensionality of the yield curve by identifying the primary directions of variance. In Redfern and McLean (2014), PCA is applied to forward rates, treating the yield curve as a vector of correlated time series. Since adjacent maturities tend to move together, PCA can efficiently capture their joint behavior. The number of principal components (PCs) retained significantly affects reconstruction accuracy: fewer components enhance interpretability, while more components improve precision (Oprea, 2022). Additionally, it is possible to analytically reverse engineer the weights of each PC to fit a specific yield curve, without the need for optimization. Interestingly, the study finds that out-of-sample curve reconstruction often requires more components than in-sample fitting, illustrating the trade-off between simplicity and generalizability (Redfern and McLean, 2014).

Although both NS and PCA perform well in summarizing the yield curve, their application

to full scenario simulation, generating plausible future curves rather than point forecasts, has been explored only to a limited extent. Most empirical studies remain focused on modeling historical data or predicting average future rates, rather than simulating joint paths of the curve over time.

Recently, there has been growing interest in machine learning approaches as flexible alternatives to these traditional methods. One such innovation is the autoencoder, a neural network designed for unsupervised learning and dimensionality reduction. In Suimon et al. (2020), the authors apply an autoencoder to the Japanese bond market, where the yield curve has undergone drastic structural shifts due to unconventional policies, such as the Bank of Japan's Yield Curve Control. These dynamics challenge linear models and create a need for more adaptive techniques.

The autoencoder in their study learns a non-linear, data-driven representation of the yield curve and encodes it into latent factors, which are learned through an unsupervised process (Sokol, 2022). According to Suimon et al. (2020), one of the key advantages they highlight is interpretability: unlike many black-box deep learning models, this autoencoder offers a transparent factor structure suited for financial applications. The authors extend the use of these factors by implementing a simple long-short trading strategy, in which the model identifies mispriced bonds. While the autoencoder proves effective for representation learning, the study concludes that for pure forecasting tasks, a Long Short-Term Memory (LSTM) model performs better. They suggest that future research should focus on hybrid models that combine predictive power with interpretability.

While Nelson-Siegel and PCA remain foundational techniques in yield curve modeling, and autoencoders offer a promising non-linear alternative, there is still a lack of direct comparison between these methods in the specific context of scenario generation. Additionally, the influence of the historical training window length on model performance has been largely overlooked in the literature. This thesis aims to fill these gaps by implementing a unified simulation framework, in which the NS, PCA, and AE models are evaluated under identical conditions. Their ability to generate plausible scenarios is assessed across multiple training periods, using probabilistic metrics such as the Continuous Ranked Probability Score (CRPS) and the Prediction Interval Coverage Probability (PICP).

CRPS is a suitable metric because it is sensitive to the full permissible range of the parameter of interest and does not require predefined classes. Moreover, in the case of deterministic forecasts, CRPS is equivalent to the Mean Absolute Error (MAE), which allows for straightforward interpretation Hersbach, 2000. In many studies involving time series forecasting, PICP is also commonly used as it provides a clear indication of whether predictions fall within the defined prediction intervals (Gusev et al., 2023).

While existing research has explored NS, PCA, and autoencoders individually, direct com-

parison between these models in the context of scenario generation remains scarce. Moreover, little attention has been paid to the influence of different training windows on model performance. This thesis addresses these gaps through a unified evaluation framework.

3 Data

This thesis uses U.S. Treasury yield curve data published by the U.S. Department of the Treasury as its primary source [1]. The dataset contains daily interest rates from January 1990 to December 2024. These yields are widely considered to be risk-free benchmark rates, reflecting the borrowing cost of the U.S. government.

Each observation of the yield curve includes up to 13 standard maturities, ranging from very short-term (1 month) to long-term (30 years). However, this study focuses on a consistent subset of 10 key maturities: 3M, 6M, 1Y, 2Y, 3Y, 5Y, 7Y, 10Y, 20Y, and 30Y. The 1M, 2M, and 4M maturities were excluded from the analysis due to incomplete historical coverage, especially in the early part of the dataset, where these tenors are often missing.

The data is reported on business days, excluding weekends and U.S. holidays. This results in a consistent time series that is well-suited for time series modeling and scenario generation. Table 1 presents five observations from the raw dataset, illustrating daily yield values across ten selected maturities. The excerpt highlights the structure of the data used throughout this study, with consistent entries across business days and missing values excluded through preprocessing.

Date	3M	6M	1Y	2Y	3Y	5Y	7Y	10Y	20Y	30Y
1995-01-03	5.95	6.66	7.23	7.73	7.84	7.88	7.91	7.88	8.07	7.93
1995-01-04	5.85	6.57	7.15	7.62	7.75	7.81	7.82	7.82	7.98	7.85
1995-01-05	5.88	6.61	7.32	7.66	7.83	7.87	7.89	7.88	8.03	7.91
1995-01-06	5.90	6.61	7.26	7.64	7.81	7.87	7.89	7.87	8.00	7.87
1990-01-09	6.00	6.67	7.27	7.68	7.84	7.90	7.92	7.89	8.03	7.90

Table 1: Excerpt from the U.S. Treasury yield dataset, showing selected maturities on five business days in early January 1995.

3.1 Data Preprocessing

The dataset used in this study contains daily U.S. Treasury yields across multiple maturities from 1990 to 2024. Data from 1990 to 2023 was used for model training, while 2024 observations were reserved for out-of-sample evaluation and scenario forecasting. Several preprocessing steps were applied to ensure consistency and reliability.

First, the dataset was restricted to business days only, thereby excluding weekends and public holidays. Short-term maturities such as 1-month, 2-month, and 4-month yields were removed due to substantial data gaps and limited historical availability. To address occasional missing values in the remaining maturities, a simple imputation method was

[1] Available at: `https://home.treasury.gov/interest-rates-data-csv-archive`

used, replacing missing entries with the average yield for the corresponding maturity. This choice was justified by the low overall proportion of missing values and their seemingly random distribution across the dataset.

The final sample consists of ten maturities ranging from 3 months to 30 years. Table 2 summarizes the descriptive statistics of these yields over the full sample period. As expected, average yields increase with maturity, while shorter maturities exhibit higher volatility. The range of observations reflects the diverse economic environments encountered during the period, including low-rate regimes and inflationary cycles.

Maturity	**Mean**	**Std Dev**	**Min**	**Max**
3 Mo	2.68	2.29	0.00	8.26
6 Mo	2.81	2.33	0.02	8.49
1 Yr	2.93	2.32	0.04	8.64
2 Yr	3.19	2.32	0.09	9.05
3 Yr	3.39	2.27	0.10	9.11
5 Yr	3.75	2.17	0.19	9.11
7 Yr	4.03	2.09	0.36	9.12
10 Yr	4.25	2.00	0.52	9.09
20 Yr	4.36	1.68	0.87	8.30
30 Yr	4.74	1.94	0.99	9.18

Table 2: Descriptive statistics of yields (1990–2023)

This cleaned and filtered dataset forms the basis for all further analysis, including model training, forecasting, and scenario simulation.

4 Methodology

This chapter outlines the methodology used to generate and evaluate future scenarios for the U.S. Treasury yield curve. The full forecasting pipeline is summarized in Figure 1, which provides a schematic overview of the key steps. Each stage corresponds to a section in this chapter: the yield curves are first reduced to a latent representation using three different models (Section 4.1). Next, the resulting latent factor series are tested for stationarity and transformed via differencing (Section 4.2), after which they are modeled using autoregressive processes and simulated forward in time (Sections 4.2–4.3). The simulated paths are then reconstructed into full yield curves (Section 4.4) and evaluated using probabilistic performance metrics (Section 4.5).

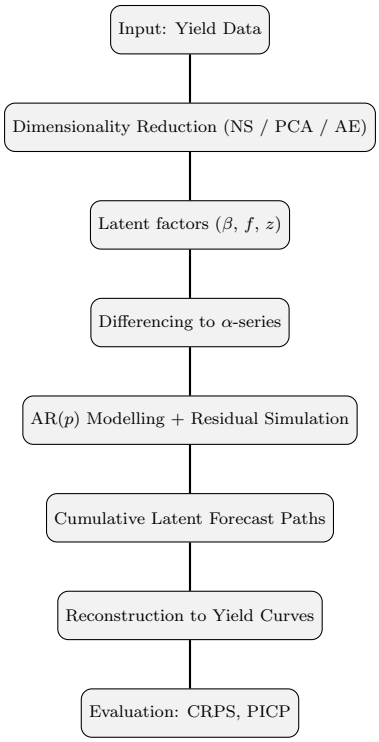


Figure 1: Schematic overview of the unified scenario pipeline. Each step corresponds to a section in this chapter: Dimensionality Reduction (§4.1), Stationarity Testing and Differencing (§4.2), AR Modeling and Residual Simulation (§4.2–§4.3), Reconstruction into yield curves (§4.4),and Scenario Evaluation (§4.5).

The following sections explain each step in detail, starting with dimensionality reduction.

4.1 Dimensionality Reduction

To generate accurate scenarios of the yield curve, it is essential to reduce the information contained in historical yield curves to a lower-dimensional representation. This step helps isolate the key dynamic components that drive changes in the term structure over

time, making it possible to simulate future paths in a tractable and interpretable way. In this study, we use three different models for this purpose: the Nelson-Siegel (NS) model, Principal Component Analysis (PCA), and an autoencoder (AE).

Nelson-Siegel Model The Nelson-Siegel (NS) model is a widely used parametric specification for modeling the yield curve. It describes the term structure of interest rates based on three latent factors: level, slope, and curvature. The functional form of the model is as follows:

$$y_t(\tau) = \beta_{1,t} + \beta_{2,t}\left(\frac{1-e^{-\lambda\tau}}{\lambda\tau}\right) + \beta_{3,t}\left(\frac{1-e^{-\lambda\tau}}{\lambda\tau} - e^{-\lambda\tau}\right) \tag{1}$$

Here, $y_t(\tau)$ denotes the interest rate at time t with maturity τ, and λ is a hyperparameter that controls the exponential decay of the basis functions. The value of λ also determines the peak of the curvature component.

The three β parameters are interpreted as follows (Diebold and Li, 2006):

- $\beta_{1,t}$: the long-term component, which reflects the general interest rate level. This value does not change as τ increases.
- $\beta_{2,t}$: the short-term component, also referred to as the slope. This factor primarily affects the difference between short and long maturities.
- $\beta_{3,t}$: the medium-term component or curvature factor. This factor adds curvature to the curve and particularly affects mid-range maturities.

To apply the model properly, an appropriate value for λ needs to be determined. Instead of selecting this value manually, a grid search is performed over the interval $[0.2, 1.0]$.

The optimal value of λ^* is the one that minimizes the average root mean squared error (RMSE) between observed and reconstructed yield curves. This procedure, inspired by Ibáñez (2015), provides a robust and consistent way of tailoring the basis functions to the data.

$$\lambda^* = \arg\min_{\lambda\in[0.2,1.0]} \left\{ \frac{1}{T}\sum_{t=1}^{T}\left(\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y_t(\tau_i) - \hat{y}_t^{(\lambda)}(\tau_i)\right)^2}\right)\right\} \tag{2}$$

where:

- T is the number of days in the training set,
- N is the number of maturities,

- $y_t(\tau_i)$ is the observed interest rate on day t for maturity τ_i,
- $\hat{y}_t^{(\lambda)}(\tau_i)$ is the reconstructed rate using the Nelson-Siegel model with parameter λ,
- and λ^* is the value that minimizes the average RMSE.

Once the optimal λ^* is identified, the NS basis functions are fixed. For each day t, the parameters $\beta_{1,t}$, $\beta_{2,t}$, and $\beta_{3,t}$ are then estimated via ordinary least squares (OLS). Each daily yield curve is represented as a linear combination of the three basis functions, and OLS is applied to find the best-fitting coefficients (Annaert et al., n.d.):

$$\hat{\boldsymbol{\beta}}_t = \left(\mathbf{X}_{\lambda^*}^\top \mathbf{X}_{\lambda^*}\right)^{-1} \mathbf{X}_{\lambda^*}^\top \mathbf{y}_t \tag{3}$$

where:

- $\mathbf{y}_t$ is the vector of observed rates on day t (dimension $N \times 1$),
- $\mathbf{X}_{\lambda^*}$ is the $N \times 3$ design matrix containing the three basis functions evaluated at maturities $\tau_1, \ldots, \tau_N$, also known as the factor loadings,
- $\boldsymbol{\beta}_t = (\beta_{1,t}, \beta_{2,t}, \beta_{3,t})^\top$,

This results in a vector of estimated factors for each day t, which summarizes the yield curve in a low-dimensional representation. These $\boldsymbol{\beta}_t$ sequences form the basis for the time series analysis and scenario simulation in the subsequent parts of this study.

Principal Component Analysis (PCA) As a statistical alternative to the Nelson-Siegel model, we apply Principal Component Analysis (PCA). PCA is a widely used dimensionality reduction technique that identifies new variables which are linear combinations of the original data. These new variables, known as principal components (PCs), are constructed to successively maximize variance while being mutually uncorrelated. Finding such components corresponds to solving an eigenvalue–eigenvector problem (Jolliffe and Cadima, 2016).

Each yield curve at time t, denoted $y_t \in \mathbb{R}^N$, is approximated as follows (Redfern and McLean, 2014):

$$y_t \approx \bar{y} + \sum_{i=1}^{n} f_{i,t} \cdot v_i \tag{4}$$

where:

- $\bar{y}$ is the historical average yield curve,
- v_i is the i-th principal component (also referred to as a *loading vector*),

- $f_{i,t}$ represents the score or projection onto the i-th component at time t.

The component vectors v_i are obtained through the following procedure. First, the covariance matrix of the centered yield curves is computed (Redfern and McLean, 2014):

$$\Sigma = \frac{1}{n-1} \sum_{t=1}^{n} (y_t - \bar{y})(y_t - \bar{y})^\top \tag{5}$$

An eigendecomposition is then performed on this covariance matrix:

$$\Sigma v_i = \lambda_i v_i \tag{6}$$

Here, v_i denotes the i-th eigenvector (principal component), and λ_i is the corresponding eigenvalue, indicating how much variance is explained by that component.

The principal components (i.e., the component scores) at time t are then computed as the projection of the centered yield curve onto the respective component:

$$f_{i,t} = (y_t - \bar{y})^\top v_i \tag{7}$$

The components are ordered such that the first principal component explains the largest proportion of variance, followed by the second, and so forth. In practice, the first three components f_1, f_2, f_3 together account for the vast majority of the variation in the data. These are often interpreted as level, slope, and curvature, analogous to the β factors in the Nelson-Siegel model (Oprea, 2022).

Autoencoder The third approach uses a non-linear dimensionality reduction technique based on a neural network: the autoencoder. An autoencoder is a type of neural network designed to compress data into a lower-dimensional representation and then reconstruct it. Autoencoders are able to uncover non-linear relationships in the data, which can result in a more effective representation compared to linear methods like PCA.

An autoencoder consists of two parts: an *encoder* and a *decoder*. The encoder reduces the original yield curve $\mathbf{y}_t \in \mathbb{R}^n$ to a compact latent representation $\mathbf{z}_t \in \mathbb{R}^k$, where $k < n$. The decoder then reconstructs an approximation $\hat{\mathbf{y}}_t$ of the original curve based on $\mathbf{z}_t$. This process is visualized in Figure 2. The figure shows how the input layer (representing the full yield curve) is compressed into a lower-dimensional "bottleneck" representation (the latent vector $\mathbf{z}$), before being expanded again into the output. This structure illustrates the key principle of an autoencoder: it forces the model to learn a compact, efficient representation of the input. The network is trained by minimizing the reconstruction error across all time

steps t.

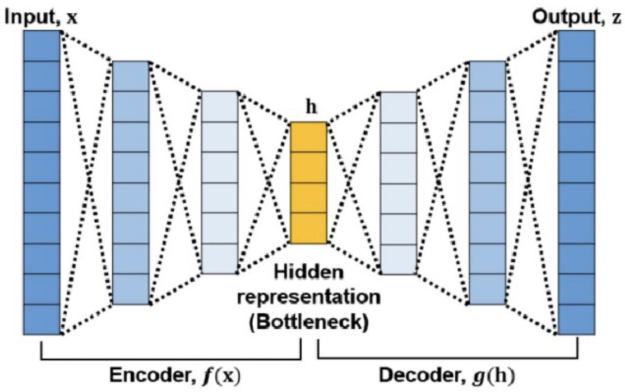


Figure 2: Structure of an autoencoder: the encoder $f(\mathbf{x})$ maps the input yield curve to a compact latent representation $\mathbf{z}$, which is then decoded by $g(\mathbf{h})$ to reconstruct the original curve. The bottleneck forces the network to learn meaningful latent features. Adapted from Kim and Song (2020).

To formally describe the architecture, we follow the notation introduced by Pathirage et al. (2018). The encoder and decoder are denoted by the functions $f(\mathbf{x})$ and $g(\mathbf{h})$, where $\mathbf{x} \in \mathbb{R}^m$ represents the input vector (e.g., a flattened yield curve), and $\mathbf{z} \in \mathbb{R}^n$ denotes the corresponding latent vector. The encoder $f(\mathbf{x})$ projects the input to a latent vector $\mathbf{z}$, while the decoder $g(\mathbf{h})$ maps it back to a reconstruction $\hat{\mathbf{x}} \in \mathbb{R}^m$.

These mappings are modeled as non-linear transformations of the form:

$$\mathbf{z} = f(\mathbf{x}) = \Phi(W\mathbf{x} + \mathbf{b}), \quad \hat{\mathbf{x}} = g(\mathbf{z}) = \Phi(\widetilde{W}\mathbf{z} + \tilde{\mathbf{b}})$$

where:

- $W, \widetilde{W} \in \mathbb{R}^{m \times n}$ are the weight matrices of the encoder and decoder respectively,
- $\mathbf{b}, \tilde{\mathbf{b}} \in \mathbb{R}^m$ are the corresponding bias vectors.
- Φ is a non-linear activation function (e.g., ReLU or sigmoid).

The network is trained to minimize the average reconstruction error across all training samples, using the following objective function:

$$[W^*, \bar{b}^*, \hat{W}^*, \hat{b}^*] = \arg \min_{W,\bar{b},\hat{W},\hat{b}} \frac{1}{m} \sum_{i=1}^{m} \left(\frac{1}{2} \| g(f(x^{(i)})) - x^{(i)} \|^2 \right) \tag{8}$$

where:

- m is the number of training samples,

- $x^{(i)}$ is the i-th input vector,
- $f(\cdot)$ and $g(\cdot)$ denote the encoder and decoder mappings.

After training, the latent vector $\mathbf{z}_t$ contains a compressed representation of the yield curve at time t. Conceptually similar to the vector $\boldsymbol{\beta}_t$ in the Nelson-Siegel model and the vector $\mathbf{f}_t$ in PCA, this representation is used for time series modeling and scenario analysis.

4.2 Forecasting Procedure

Once the latent vectors ($\boldsymbol{\beta}_t$, $\mathbf{f}_t$, or $\mathbf{z}_t$) have been obtained via one of the dimensionality reduction techniques described in Section 4.1. The next step is to simulate realistic future scenarios for the yield curve. This section outlines the general procedure applied to all three models.

Stationarity and Transformation To apply time series models, it is essential that the latent sequences are stationary. A stationary series maintains constant statistical properties over time, such as mean, variance, and autocorrelation (Tsay, 2010). To test this, both the Augmented Dickey-Fuller (ADF) and Kwiatkowski–Phillips–Schmidt–Shin (KPSS) tests are performed on each latent factor sequence. These complementary tests provide a comprehensive view of the presence of unit roots and potential trend components (Lopez, 1997).

Non-stationary series such as random walks pose three major problems for forecasting:

- **No mean-reverting behavior:** The expected value at any future point equals the last observation: $\hat{p}_h(\ell) = p_h$. This leads to limited predictive power.
- **Exploding variance:** Forecast error variance grows with horizon: $\mathrm{Var}(e_h(\ell)) = \ell\sigma_a^2$, causing increasing uncertainty as $\ell \to \infty$.
- **Permanent shocks:** Every shock has a lasting effect, leading to long memory that is difficult to model with classical AR models.

To address potential non-stationarity in the latent factor series, differencing is applied. Let $\beta_{i,t}$ denote the value of the i-th latent factor at time t. Then the differenced series $\Delta^n \beta_{i,t}$ is defined recursively as follows:

$$
\begin{aligned}
\Delta\beta_{i,t} &= \beta_{i,t} - \beta_{i,t-1} \\
\Delta^2\beta_{i,t} &= \Delta\beta_{i,t} - \Delta\beta_{i,t-1} = (\beta_{i,t} - \beta_{i,t-1}) - (\beta_{i,t-1} - \beta_{i,t-2}) \\
&= \beta_{i,t} - 2\beta_{i,t-1} + \beta_{i,t-2} \\
&\vdots \\
\Delta^n\beta_{i,t} &= \sum_{j=0}^{n} (-1)^j \binom{n}{j} \beta_{i,t-j}
\end{aligned}
$$

This general formulation allows for differencing of any order n, depending on the level of non-stationarity detected in the data. Applying differencing transforms a non-stationary time series into a stationary one, which is a necessary condition for many time series models to yield reliable forecasts.

A similar transformation is applied to the latent vectors from PCA ($\mathbf{f}_t$) and the Autoencoder ($\mathbf{z}_t$). For each latent factor sequence $\beta_{i,t}$, the differenced series is defined as

$$
\alpha_{i,t} = \Delta^n \beta_{i,t}, \tag{9}
$$

where Δ^n denotes the n-th order difference operator. In the case of PCA and Autoencoder models, the same transformation is applied to the corresponding latent factors, replacing $\beta_{i,t}$ with $f_{i,t}$ and $z_{i,t}$, respectively.

After transformation, most α-series satisfy the criteria for stationarity. Although some exceptions exist, the series are generally suitable as inputs for autoregressive models.

Time Series Modeling Each differenced factor series is modeled as in Eq. 9. An autoregressive model of order p (AR(p)) is estimated for each series. The optimal lag p is selected through a grid search using both the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC):

$$
\alpha_{i,t} = c + \sum_{j=1}^{p} \phi_j \alpha_{i,t-j} + \varepsilon_t \tag{10}
$$

where c is the constant term, ϕ_j are the lag coefficients, and ε_t denotes the residuals.

Parameters are estimated via ordinary least squares (OLS). To ensure model adequacy, residuals are tested for autocorrelation using the Ljung–Box test.

4.3 Residual Simulation

After fitting the AR(p) models, the residuals ε_t are retained to form the basis for scenario simulation. Adding realistic noise to future predictions is essential to generate plausible yield curve trajectories.

For each α-component, residuals are collected and their empirical cumulative distribution function (ECDF) is computed. Inverting this ECDF allows uniform samples to be mapped back to residuals consistent with the empirical distribution.

- For each future time step $t \in \{1, \ldots, h\}$, a fixed number of residual samples is generated per α-component.
- This results in a tensor of dimension $h \times S \times k$, where S is the number of simulation paths and k is the number of latent factors. Each slice captures a unique residual sequence for one scenario across all components.

Preserving Correlation Structure via Copula Although the α-series are modeled separately, it is crucial to retain the interdependence between residuals, especially in extreme market conditions (tail dependence).

To this end, a copula decomposition is employed. A copula is a function that joins univariate distributions into a multivariate distribution with a specified dependence structure (Trivedi and Zimmer, 2011).

Historical residuals are first transformed to the uniform scale using empirical cumulative distribution functions (ECDFs). A *Gaussian copula* is then fitted to these transformed values. By applying the copula to new samples drawn from the uniform distribution, simulated residuals $\hat{\varepsilon}_{t+1}$ are obtained with a dependence structure similar to the training data.

4.4 Reconstructing Latent Representations

With a set of S simulated residuals and differenced latent factors, we now turn to reconstructing the original latent factor trajectories via forward simulation.

Forward Simulation via AR(p) With the simulated residuals and historical α values in place, future α-series can be generated via Monte Carlo simulation over a horizon of h business days.

For each scenario s, the forecast at time $t+1$ is computed as:

$$\alpha_{i,t+1}^{(s)} = c + \sum_{j=1}^{p} \phi_j \alpha_{i,t+1-j}^{(s)} + \hat{\varepsilon}_{t+1}^{(s)} \tag{11}$$

This process is iterated through $t+2, \ldots, t+h$ for each of the three components. The result is a set of simulated α-values serving as inputs for reconstructing the original latent representations.

Reconstruction of Latent Factors The simulated α-series are then cumulatively summed to recover future latent factor values. In the case of first-order differencing ($n = 1$), the reconstruction proceeds as follows:

$$\begin{aligned}
\beta^{(s)}_{i,t+1} &= \beta^{\text{last}}_{i,t} + \alpha^{(s)}_{i,t+1} &&= \mathcal{R}_1\left(\beta^{\text{last}}_{i,t}, \alpha^{(s)}_{i,t+1}\right) \\
\beta^{(s)}_{i,t+2} &= \beta^{(s)}_{i,t+1} + \alpha^{(s)}_{i,t+2} &&= \mathcal{R}_1\left(\beta^{(s)}_{i,t+1}, \alpha^{(s)}_{i,t+2}\right) \\
&\vdots && \vdots && \vdots \\
\beta^{(s)}_{i,t+h} &= \beta^{(s)}_{i,t+h-1} + \alpha^{(s)}_{i,t+h} &&= \mathcal{R}_1\left(\beta^{(s)}_{i,t+h-1}, \alpha^{(s)}_{i,t+h}\right)
\end{aligned}$$

Here $\mathcal{R}_1(a, b) = a + b$ denotes the first-order reconstruction operator, which recursively sums the differenced values to recover the level process.

To allow for differencing of arbitrary order n, the general reconstruction is expressed as:

$$\begin{aligned}
\beta^{(s)}_{i,t+1} &= \mathcal{R}_n\left(\beta^{(s)}_{i,t}, \ldots, \beta^{(s)}_{i,t+1-n}, \alpha^{(n)}_{i,t+1}\right) \\
\beta^{(s)}_{i,t+2} &= \mathcal{R}_n\left(\beta^{(s)}_{i,t+1}, \ldots, \beta^{(s)}_{i,t+2-n}, \alpha^{(n)}_{i,t+2}\right) \\
&\vdots \\
\beta^{(s)}_{i,t+h} &= \mathcal{R}_n\left(\beta^{(s)}_{i,t+h-1}, \ldots, \beta^{(s)}_{i,t+h-n}, \alpha^{(n)}_{i,t+h}\right)
\end{aligned}$$

where $\mathcal{R}_n(\cdot)$ denotes the inverse of the n-th order differencing operator. This operator reconstructs the level series based on the simulated differences and the required number of lagged values.

The same reconstruction procedure is applied to the PCA-based scores $\mathbf{f}_t$ and the autoencoder representations $\mathbf{z}_t$.

Reconstruction of Yield Curves Finally, the simulated latent vectors are used to reconstruct a full yield curve for each scenario and time step:

- **Nelson-Siegel:** Simulated $\boldsymbol{\beta}$ values are inserted into Equation (1), with fixed λ.

- **PCA:** Simulated $\mathbf{f}$ components are linearly combined with their corresponding loading vectors.
- **Autoencoder:** $\mathbf{z}$ vectors are passed through the decoder, followed by inverse standardization.

This results in a total of $h \times S$ yield curves, one for each time point and scenario. Since each simulation incorporates random residuals, a full distribution of plausible interest rate curves is generated for every future day.

4.5 Evaluation Metrics

To assess the quality of the generated yield curve scenarios, two probabilistic evaluation metrics are applied: the *Continuous Ranked Probability Score* (CRPS) and the *Prediction Interval Coverage Probability* (PICP). For each model and each training window, these scores are computed based on S Monte Carlo simulations over the forecast horizon $t + h$.

Continuous Ranked Probability Score (CRPS) The *Continuous Ranked Probability Score* (CRPS) is a scoring rule for probabilistic forecasts that compares the full cumulative distribution function (CDF) of a forecast with the observed realization. Unlike point estimation metrics such as the *Mean Squared Error* (MSE), CRPS evaluates the entire distribution and penalizes both bias and dispersion (Gneiting and Raftery, 2007).

The formal definition is given by (Gneiting and Raftery, 2007):

$$\mathrm{CRPS}(F, x) = \int_{-\infty}^{\infty} \left(F(y) - \mathbb{1}_{\{y \geq x\}}\right)^2 \, dy \tag{12}$$

where:

- F is the cumulative distribution function (CDF) of the predictive scenario,
- x is the observed realization,
- $\mathbb{1}_{\{y \geq x\}}$ is the indicator function, which equals 1 if $y \geq x$, and 0 otherwise.

Lower CRPS values indicate better forecasts: the closer the predicted distribution is to the observed value, the lower the score. In the special case of deterministic forecasts, CRPS reduces to the *Mean Absolute Error* (MAE) (Hersbach, 2000).

In this study, for each forecast day $t + h$, a total of S yield curves are simulated. Each curve contains interest rates for 10 different maturities $\tau_1, \ldots, \tau_{10}$.

The observed yield curve on day t is denoted as:

$$\mathbf{y}_t^{\text{true}} = [y_{t,1}, y_{t,2}, \ldots, y_{t,10}]$$

where $y_{t,i}$ denotes the observed interest rate at maturity τ_i.

For each maturity τ_i, we obtain S simulated values $y_{t,i}^{(1)}, \ldots, y_{t,i}^{(S)}$. From these samples, an empirical distribution F_i is constructed. A CRPS score is then computed for each of the 10 maturities:

$$\text{CRPS}_t^{(i)} = \text{CRPS}(F_i, y_{t,i})$$

This yields ten CRPS values per day t, one for each maturity. To summarize the overall curve forecast quality, we compute the unweighted average over all maturities (Gneiting and Raftery, 2007):

$$\text{CRPS}_t^{\text{curve}} = \frac{1}{10} \sum_{i=1}^{10} \text{CRPS}_t^{(i)}$$

Prediction Interval Coverage Probability (PICP) The *Prediction Interval Coverage Probability* (PICP) measures the percentage of observed values that fall within the predicted prediction intervals. It serves as an indicator of the *calibration* or reliability of a probabilistic forecast.

The formal definition is given by Gusev et al. (2023):

$$\text{PICP} = \frac{1}{N} \sum_{i=1}^{N} c_i \tag{13}$$

where:

- N is the number of observations (e.g., maturities per day),
- $c_i = 1$ if the observation y_i lies within the predicted interval $[l_i, u_i]$, and 0 otherwise,
- l_i and u_i are respectively the 0.5% and 99.5% quantiles of the forecast distribution (for a 99% interval).

A well-calibrated forecast results in a PICP close to the nominal confidence level (e.g., 99%). Values below this level suggest underestimated uncertainty, while values above it imply overly wide intervals.

In this study, the PICP is implemented as follows. For each forecast day t:

- S interest rate scenarios are generated for each of the N maturities $\tau_1, \ldots, \tau_N$,

- for each maturity τ_i, the 99% prediction interval is computed as the $[0.5\%, 99.5\%]$ quantile range of the simulations,
- it is then checked whether the observed rate $y_{t,i}$ lies within this interval.

This produces N binary indicator values $c_i \in \{0, 1\}$ per day. Averaging over these yields the daily PICP score:

$$\text{PICP}_t = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}\{l_i \leq y_{t,i} \leq u_i\} \tag{14}$$

A value of $\text{PICP}_t = 0.99$ implies that 99% of observed rates on day t fall within their respective prediction intervals.

Interpretation The CRPS and PICP are complementary: CRPS evaluates the full shape and sharpness of the forecast distribution, while PICP focuses specifically on interval reliability. Together, they provide a robust assessment of model performance for scenario-based yield curve forecasting, balancing both accuracy and uncertainty quantification.

4.6 Methodological Choices

Several methodological decisions were made in the implementation of the modelling framework, with the aim of ensuring consistency, interpretability, and robustness, regardless of which of the three models is applied.

Uniform Modelling Pipeline Despite methodological differences between the three models (Nelson-Siegel, PCA, and Autoencoder), a uniform forecasting pipeline is applied across all methods. Each model generates a latent time series consisting of three components: the $\boldsymbol{\beta}$-series for Nelson-Siegel, principal component scores $\mathbf{f}_t$ for PCA, and bottleneck representations $\mathbf{z}_t$ for the autoencoder (see Section 4.1).

These latent sequences are first tested for stationarity and transformed using 1st order differencing (see Section 4.2), resulting in the differenced $\boldsymbol{\alpha}$-series. Each α-component is then modeled using an autoregressive AR(p) process.

Once the AR coefficients are estimated, S scenarios are generated via Monte Carlo simulation of the residuals, using either an ECDF transform or a Gaussian copula, depending on the model (See Section 4.3). The α-series are then cumulatively summed into latent forecast paths and reconstructed into full yield curves using the appropriate inverse mapping: the NS formula, the PCA projection, or the AE decoder (Section 4.4).

This generic structure allows for a fair comparison between models and enables probabilistic evaluation based on identical output formats, using CRPS and PICP scores (Section 4.5).

Choice of Three Latent Factors Both the PCA and Autoencoder models were implemented using a three-dimensional latent representation, striking a balance between information reduction and explanatory power. For PCA, this choice was based on the scree plot and the cumulative explained variance (see Figures 3 and 4). The first three components jointly explain over 99% of the total variance. As shown in Figure 3, the eigenvalue drops sharply after the third component, suggesting that most information is concentrated in the first three components. Figure 4 illustrates how the cumulative variance flattens beyond this point, indicating that additional components beyond the third add little explanatory value. Based on these patterns, retaining three latent components preserves almost all variation in the dataset while avoiding overfitting.

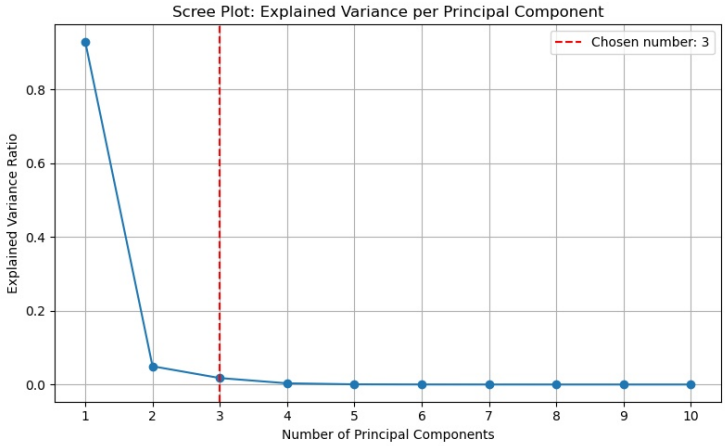


Figure 3: Scree plot showing the proportion of explained variance per principal component. The first three components jointly account for over 99% of the total variance, which supports the choice of a three-factor representation for PCA.

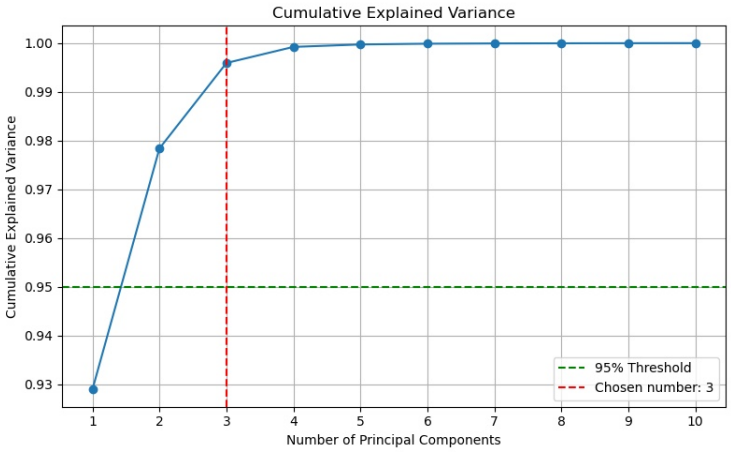


Figure 4: Cumulative explained variance of PCA components. The curve plateaus after the third component, confirming that the majority of variance in the yield curve data is captured by the first three components.

For the Autoencoder, a reconstruction error curve was used to assess the effect of the bottleneck dimension on model fit. As shown in Figure 5, reconstruction error (measured via mean squared error) decreases as the number of latent nodes increases, with the lowest error observed at dimension 9. However, the drop in error beyond three dimensions is small and inconsistent, suggesting that most of the signal is already captured by the first three factors. Therefore, three latent nodes were also chosen here, aligning with PCA for comparability and model parsimony.

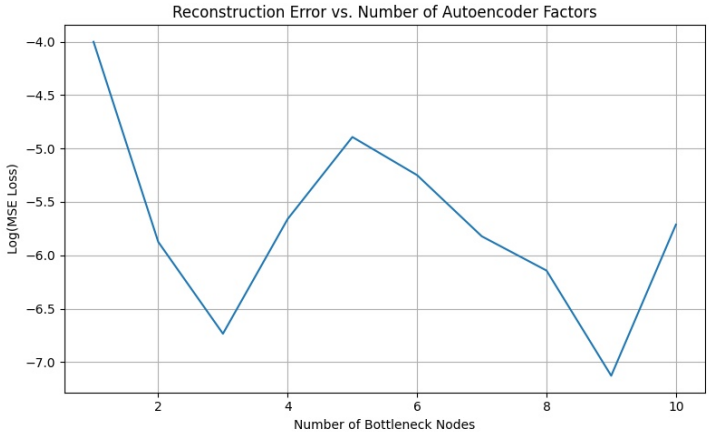


Figure 5: Reconstruction error (mean squared error) of the autoencoder as a function of the number of latent nodes. While higher dimensions slightly reduce error, the improvement beyond three latent nodes is marginal, supporting the choice of a 3-dimensional bottleneck for comparability and parsimony.

Stationarity Testing: ADF as Primary Criterion To assess whether the latent factor series were stationary, both the Augmented Dickey-Fuller (ADF) and KPSS tests were applied. While both provide complementary insights with the ADF testing for the presence of a unit root and the KPSS assuming stationarity under the null , the ADF test was taken as the primary criterion in practice. This aligns with standard econometric time series practice, where the ADF test is widely accepted for determining integration order (Harris, 1995).

In nearly all cases, the latent series (i.e., $\boldsymbol{\beta}_t$ in Nelson-Siegel, $\mathbf{f}_t$ in PCA, and $\mathbf{z}_t$ in AE) failed the ADF and KPSS tests, and were therefore considered non-stationary. A small fraction of the series did pass the test without differencing, but to ensure methodological consistency, both across models and across time , all series were uniformly first-differenced (i.e., differencing order $n = 1$). After this transformation, all α-series passed the ADF test for stationarity without exception. This was not the case for the KPSS test. As argued in various papers, including Boswijk and Franses (1992), the preferred test is one that avoids unnecessary differencing, which makes ADF more efficient in practice. This approach was applied to all three models: the Nelson-Siegel factors ($\boldsymbol{\beta}$), the PCA components ($\mathbf{f}$), and

the Autoencoder latent codes ($\mathbf{z}$).

This uniform approach ensures that the autoregressive modelling of the α-series is carried out consistently, avoiding model- or data-specific exceptions that could otherwise introduce bias into the forecasting or simulation procedures.

Model Selection: BIC Over AIC To determine the optimal lag order p in the AR(p) models, both the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) were considered. Ultimately, BIC was chosen as the leading metric due to its stricter penalty on model complexity. AIC tends to favour more flexible models with higher order, which may fit the training data better but are also more vulnerable to overfitting. BIC is more conservative and promotes parsimony, which is desirable in a high-frequency setting and when generating out-of-sample scenarios (Burnham and Anderson, 2004).

Residual Simulation: Copula vs. Multivariate Normal In the Monte Carlo forecasting step, two approaches were evaluated for simulating residuals while preserving the dependence between the three α-series: Gaussian copula sampling and multivariate normal simulation.

For the Nelson-Siegel and Autoencoder models, a Gaussian copula combined with non-parametric marginal distributions (ECDFs) was used. This allows for a flexible, non-Gaussian dependence structure while preserving the empirical behaviour of each marginal. The copula ensures that simulated shocks reflect the historical correlations between components, a crucial requirement when simulating joint dynamics.

For the PCA model, however, the copula approach was deliberately avoided. Empirical tests showed that the historical correlation structure of PCA residuals was better preserved using direct multivariate normal sampling combined with inverse ECDF transformation. Using a copula in this case introduced unnatural dependencies and instability in the simulations. Therefore, a simpler method was chosen: standard normally distributed samples were drawn jointly, transformed to uniform margins via the standard normal CDF, and then mapped back using the ECDF. As a result, the copula approach was not applied to PCA.

Training Windows To test model robustness across different economic environments, a rolling forecast structure was applied. For each model, training was performed using four different historical start dates: 1990, 2000, 2010, and 2020.

For each of these starting points, a four-step prediction cycle was run across 2024. In each step, the model is retrained up to the current point and then used to forecast the following quarter. The full cycle is structured as follows:

- Train up to end of 2023 → Forecast Q1 2024
- Train up to end of Q1 2024 → Forecast Q2 2024
- Train up to end of Q2 2024 → Forecast Q3 2024
- Train up to end of Q3 2024 → Forecast Q4 2024

This rolling forecast procedure was repeated for each of the four training windows. As a result, model performance can be assessed in both calm and turbulent periods, strengthening the generalisability of the conclusions.

5 Results

This section evaluates the performance of the three scenario generation models: Nelson-Siegel, Principal Component Analysis (PCA), and Autoencoder , introduced in Section 4. Each model is assessed using a consistent framework that includes: reconstruction accuracy, time series modeling of latent factors, residual simulation, scenario quality, and probabilistic forecast evaluation. The empirical focus is on forecasting Q1 2024, consisting of 61 trading days. Forecasts are generated using 10,000 Monte Carlo simulations per model. Two evaluation setups are employed. First in Section 5.1, we assess each model using a fixed historical training window (1990–2023). Second, in Section 5.2, we evaluate robustness under varying economic regimes using a rolling forecast procedure across multiple training periods.

5.1 Single-Window Results

This subsection presents the results of each model based on a single, fixed training window spanning the full historical period from 1990 to 2023. By holding the training period constant, the individual behavior of each model can be assessed in detail, without the confounding effects of changing economic regimes or shifting calibration windows.

Nelson-Siegel (NS) The Nelson-Siegel model describes the yield curve using three latent factors: level, slope, and curvature. To verify whether these factors are properly identified over the entire sample period, both the average observed yield curve from 1990–2023 and a reconstructed version based on the average β-values were generated, where β refers to the factor loadings defined in Paragraph 4.1. Figure 6 shows that the reconstructed curve closely tracks the empirical average across all maturities, with only minor deviations at the long end. This close alignment suggests that the NS model is flexible enough to capture the dominant structural features of the yield curve, and that the three latent factors provide a stable and interpretable basis for scenario generation.

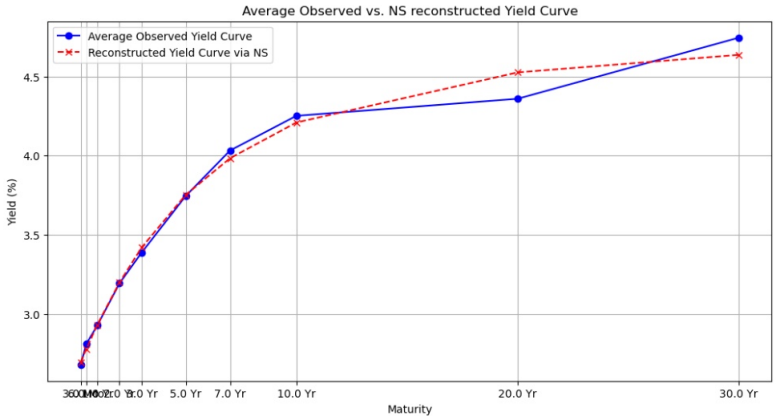


Figure 6: Average observed yield curve (1990–2023) and reconstructed yield curve based on the average $\boldsymbol{\beta}$-values from the Nelson-Siegel model. The figure shows both curves across the full maturity spectrum, allowing visual assessment of the model's ability to replicate the average term structure.

After making the time series stationary as discussed in Paragraph 4.2, the optimal lag order p was selected for each factor α_{β_i} using the Bayesian Information Criterion (BIC). This resulted in an AR(1) model for α_{β_1} and AR(10) models for α_{β_2} and α_{β_3}.

Furthermore, the Nelson-Siegel model's decay parameter λ was calibrated by minimizing the RMSE over the full sample of historical yield curves. The optimal value was found to be $\lambda = 0.4263$, resulting in an in-sample RMSE of approximately 0.3099.

The estimated models are:

AR(1) model for α_{β_1}

$$\alpha_{\beta_1,t} = -0.00020 - 0.02927 \cdot \alpha_{\beta_1,t-1} + \varepsilon_{1,t}$$

AR(10) model for α_{β_2}

$$\begin{aligned}\alpha_{\beta_2,t} = &-0.00003 - 0.09979 \cdot \alpha_{\beta_2,t-1} - 0.04004 \cdot \alpha_{\beta_2,t-2} - 0.05415 \cdot \alpha_{\beta_2,t-3} + 0.02181 \cdot \alpha_{\beta_2,t-4} \\ &- 0.02123 \cdot \alpha_{\beta_2,t-5} + 0.03059 \cdot \alpha_{\beta_2,t-6} - 0.00708 \cdot \alpha_{\beta_2,t-7} - 0.02905 \cdot \alpha_{\beta_2,t-8} \\ &+ 0.00580 \cdot \alpha_{\beta_2,t-9} + 0.08731 \cdot \alpha_{\beta_2,t-10} + \varepsilon_{2,t}\end{aligned}$$

AR(10) model for α_{β_3}

$$\begin{aligned}\alpha_{\beta_3,t} = &-0.00118 - 0.13505 \cdot \alpha_{\beta_3,t-1} - 0.06948 \cdot \alpha_{\beta_3,t-2} - 0.07850 \cdot \alpha_{\beta_3,t-3} + 0.04021 \cdot \alpha_{\beta_3,t-4} \\ &- 0.07179 \cdot \alpha_{\beta_3,t-5} + 0.04526 \cdot \alpha_{\beta_3,t-6} - 0.01382 \cdot \alpha_{\beta_3,t-7} - 0.00510 \cdot \alpha_{\beta_3,t-8} \\ &- 0.01997 \cdot \alpha_{\beta_3,t-9} + 0.06855 \cdot \alpha_{\beta_3,t-10} + \varepsilon_{3,t}\end{aligned}$$

To evaluate whether the simulated residuals preserve the characteristics of the historical residuals, Figure 7 compares the empirical cumulative distribution function (CDF) of historical residuals for α_{β_1} to the CDF of simulated residuals at forecast horizon $t + 6$. The close overlap between the two curves indicates that the marginal distribution of residuals is well preserved by the simulation process.

To assess this more systematically, Appendix A.1 shows additional CDF comparisons for α_{β_2} and α_{β_3} at the same forecast horizon $(t + 6)$. These confirm that the marginal distributions of simulated residuals for these components also closely align with their historical counterparts.

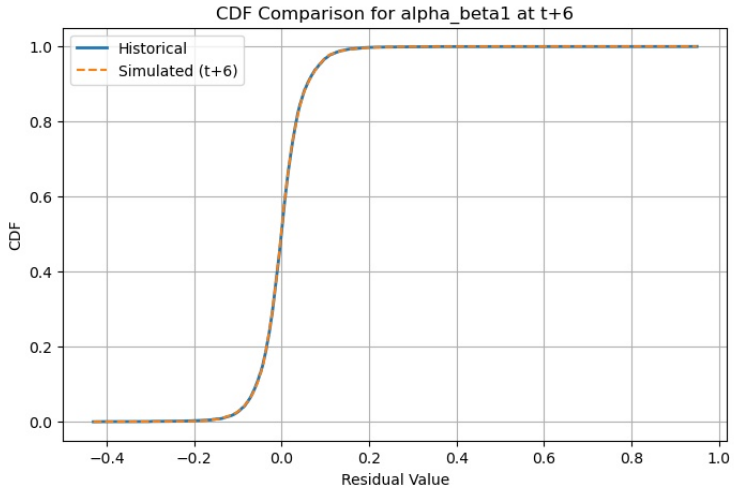
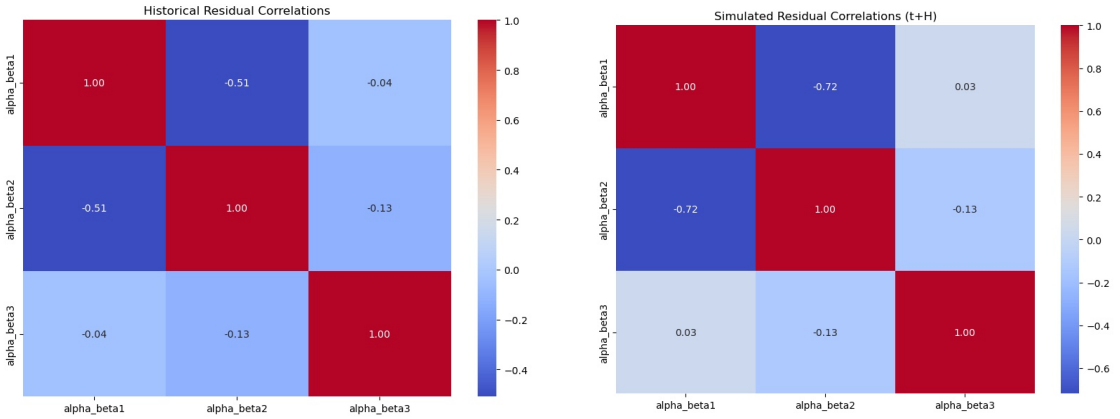


Figure 7: Empirical cumulative distribution function (CDF) of historical residuals for α_{β_1} and the corresponding simulated residuals at $t + 6$.

Beyond marginal behavior, it is also important to preserve the dependence structure between residual components. Figure 8 presents the historical correlation matrix of the three differenced latent factors α_{β_1}, α_{β_2}, and α_{β_3} (Panel 8a), alongside the corresponding matrix computed from simulated residuals (Panel 8b). The comparison reveals that the overall sign and magnitude of correlations are maintained, particularly the strong negative correlation between α_{β_1} and α_{β_2}. However, the simulated correlation appears slightly stronger than in the historical data.

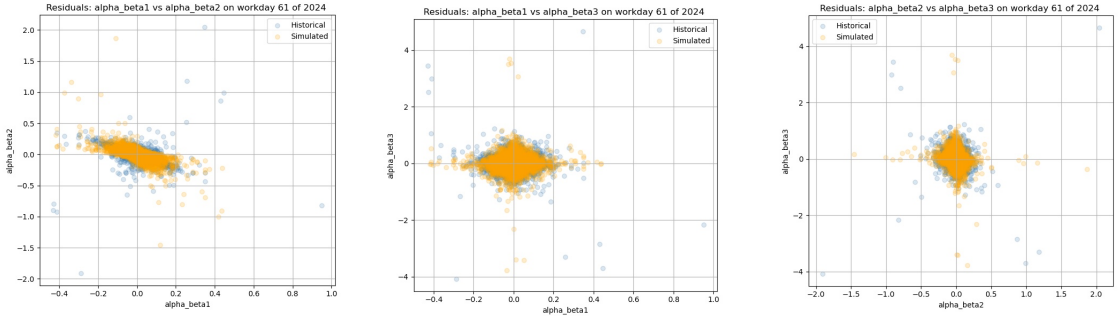


(a) Historical correlation matrix. **(b)** Simulated correlation matrix at $t + H$.

Figure 8: Correlation structure of residuals from the Nelson-Siegel model. Panel 8a shows the empirical correlation matrix of the historical residuals from the first-differenced latent factors α_{β_1}, α_{β_2}, and α_{β_3}, based on data from 1990–2023. Panel 8b shows the correlation matrix of simulated residuals at forecast horizon $t + H$, where $H = 61$ (corresponding to 61 business days ahead). Residuals are simulated using inverse empirical CDF transformations, with dependence preserved via a Gaussian copula as described in Section 4.3.

A more detailed inspection is shown in Figure 9, which presents scatterplots of the pairwise relationships between residuals. The simulated residuals (orange) closely resemble the historical residuals (blue) not only in overall shape but also in the behavior of extreme values. This comparison suggests that the tail dependence between latent factor innovations is preserved in the simulated data.

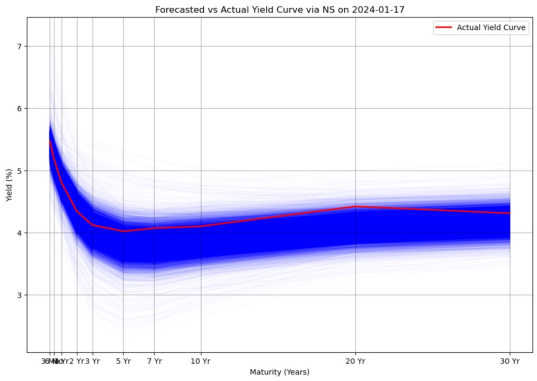
This visual impression is supported numerically by the tail dependence coefficients reported in Appendix A.2. Both lower and upper tail values are of similar magnitude between historical and simulated residuals, indicating that the dependence in the extremes is broadly retained across the forecast horizon.



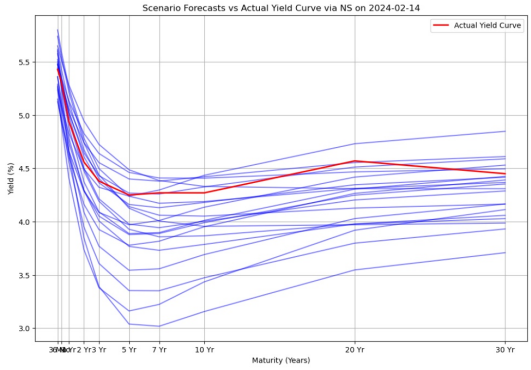
(a) α_{β_1} vs α_{β_2} **(b)** α_{β_1} vs α_{β_3} **(c)** α_{β_2} vs α_{β_3}

Figure 9: Pairwise scatterplots of residuals from the Nelson-Siegel model. Each subplot compares historical and simulated residuals for a pair of first-differenced latent factors: 9a α_{β_1} vs. α_{β_2}, 9b α_{β_1} vs. α_{β_3}, and 9c α_{β_2} vs. α_{β_3}. Historical residuals (1990–2023) are shown in blue; simulated residuals at forecast horizon $t + H$ (with $H = 61$ business days) are shown in orange.

To visualize the quality of the scenarios, Figure 10 presents simulated yield curves for a forecast day in Q1 2024. Subfigure 10a shows the full set of 10,000 simulated yield curves, which captures a broad range of possible outcomes. The actual observed yield curve is overlaid in red for reference. Subfigure 10b presents 20 randomly selected simulations, allowing a closer look at individual forecast trajectories and their alignment with the observed curve.



(a) Simulated yield curves from the Nelson-Siegel model for a forecast horizon of $t + 10$ business days. The full simulation cloud contains 10,000 generated yield curves for a specific forecast date in Q1 2024. The actual observed yield curve on that day is shown in red for comparison.

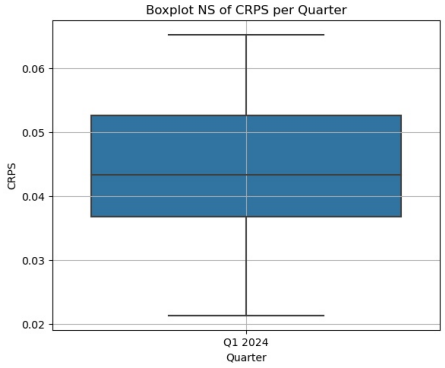


(b) Subset of 20 simulated yield curves randomly selected from the full scenario set shown in (a), plotted alongside the actual yield curve for forecast day $t + 10$ in Q1 2024. This provides a more detailed view of individual forecast paths.

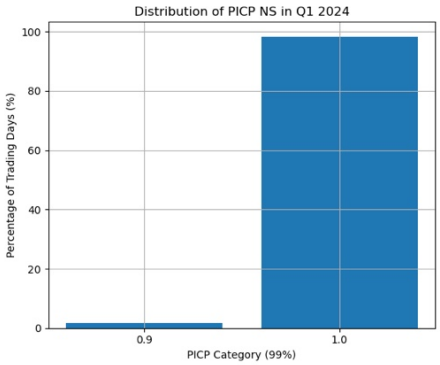
Figure 10: Simulated yield curve scenarios using the NS model.

Finally, the probabilistic performance of the NS model is assessed using the CRPS and PICP metrics. Figure 11 shows the distribution of CRPS scores for Q1 2024. The CRPS distribution in Subfigure 11a has a median of approximately 0.0446, with an interquartile range from about 0.0370 to 0.0537. The mean CRPS is 0.0452 with a standard deviation of 0.0102.

Subfigure 11b shows the distribution of PICP at the 99% confidence level. that in almost all cases, the realized rates fall within the 99% prediction intervals. Nearly all forecast days achieve full coverage. One day seems to have one maturity outside the 99% prediction interval.



(a) Boxplot of CRPS (Continuous Ranked Probability Score) values computed for the Nelson-Siegel model over all forecast days in Q1 2024. Scores are calculated based on simulated yield curve scenarios compared to observed yield curves, as described in Section 4.2. The CRPS is computed for each day and maturity and summarized across the quarter.



(b) Distribution of Prediction Interval Coverage Probability (PICP) values at the 99% confidence level for the NS model over Q1 2024. For each forecast day, the percentage of maturities where the observed rate falls within the predicted 99% interval is computed.

Figure 11: Evaluation metrics for the NS model in Q1 2024.

Principal Component Analysis (PCA) The PCA model offers a statistical alternative to the Nelson-Siegel specification. Instead of relying on predefined basis functions, PCA reduces the yield curve to three principal components, which are linear combinations of the original yields.

Figure 12 shows the average observed yield curve over the period 1990–2023, along with its reconstruction based on the three average component values and corresponding loading vectors. The two curves are nearly indistinguishable, suggesting that the first three components capture the key structural features of the yield curve. This visual agreement supports the suitability of a three-dimensional latent representation for the PCA model.

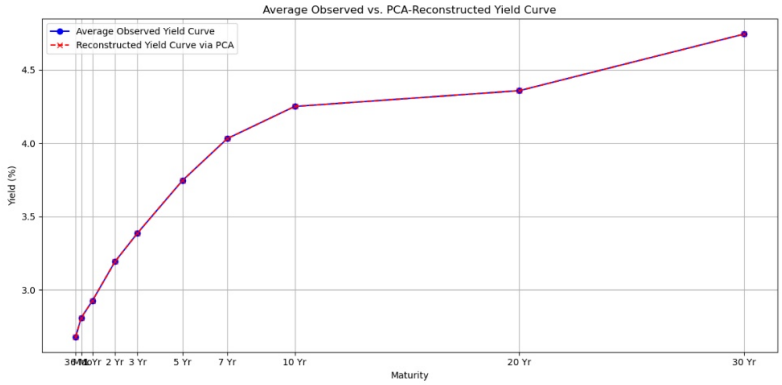


Figure 12: Average observed yield curve (1990–2023) and reconstructed yield curve based on the first three principal components from the PCA model. The reconstruction is obtained using the average component scores and corresponding loading vectors across the full sample period. The figure displays both curves over the maturity spectrum for a visual assessment of the model's ability to approximate the average term structure.

As explained in Paragraph 4.2, the first differences of the PCA latent factors α_{f_i} were tested for stationarity. For each series, the optimal lag order p was selected based on the Bayesian Information Criterion (BIC). This resulted in an AR(2) model for α_{f_1}, an AR(10) model for α_{f_2}, and an AR(1) model for α_{f_3}. The estimated specifications are:

AR(2) model for α_{f_1}

$$\alpha_{f_1,t} = -0.00152 - 0.19076 \cdot \alpha_{f_1,t-1} - 0.05444 \cdot \alpha_{f_1,t-2} + \varepsilon_{1,t}$$

AR(10) model for α_{f_2}

$$\begin{aligned}\alpha_{f_2,t} = {} & 0.00023 - 0.03607 \cdot \alpha_{f_2,t-1} - 0.03733 \cdot \alpha_{f_2,t-2} - 0.06026 \cdot \alpha_{f_2,t-3} \\ & + 0.02959 \cdot \alpha_{f_2,t-4} - 0.03706 \cdot \alpha_{f_2,t-5} + 0.02873 \cdot \alpha_{f_2,t-6} \\ & + 0.01520 \cdot \alpha_{f_2,t-7} - 0.03336 \cdot \alpha_{f_2,t-8} - 0.00076 \cdot \alpha_{f_2,t-9} + 0.08149 \cdot \alpha_{f_2,t-10} + \varepsilon_{2,t}\end{aligned}$$

AR(1) model for α_{f_3}

$$\alpha_{f_3,t} = -0.00032 - 0.02329 \cdot \alpha_{f_3,t-1} + \varepsilon_{3,t}$$

To assess whether the simulated residuals are realistic, Figure 13 compares the empirical cumulative distribution function of the historical residuals of α_{f_1} with the simulated distribution on a randomly selected future day $t + 6$. The close similarity confirms that the simulation procedure captures the marginal distributions. Additional results for α_{f_2} and α_{f_3} at the same horizon are shown in Appendix A.1, and yield similarly close matches.

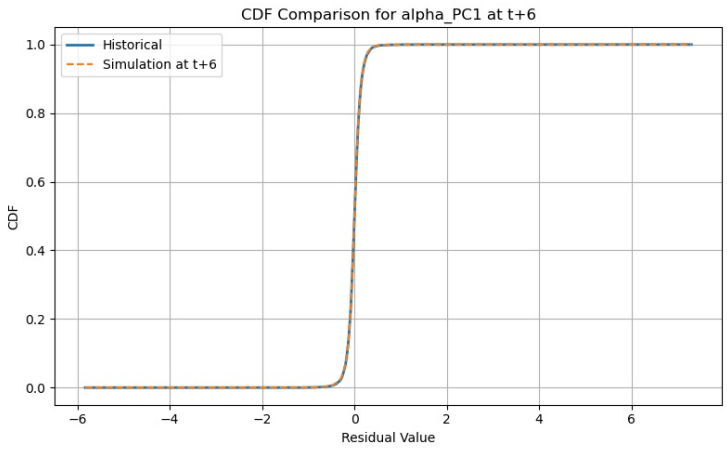
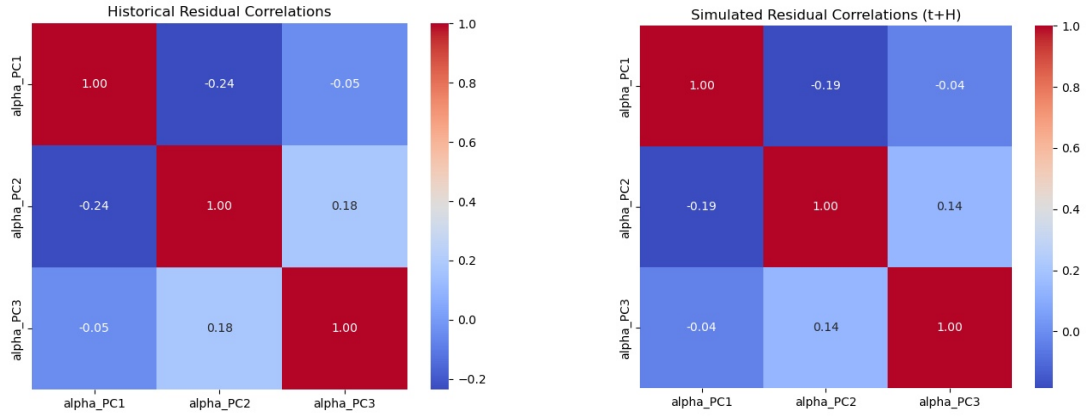


Figure 13: Empirical cumulative distribution function (CDF) of historical PCA residuals for α_{f_1} and the corresponding simulated residuals at forecast horizon $t + 6$.

In addition to the marginal distributions, the dependence structure between PCA residuals is also examined. Figure 14 displays the correlation matrices of the first-differenced latent factors α_{f_1}, α_{f_2}, and α_{f_3}, comparing the historical estimates to those derived from simulated scenarios. Panel 14a shows the historical correlation matrix computed from the full sample period (1990–2023), while Panel 14b displays the corresponding matrix based on simulated residuals at forecast horizon $t + H$. The overall structure, including the signs and relative strengths of the correlations, is largely preserved in the simulated data. Some minor deviations appear in the off-diagonal entries, but the general dependence pattern is maintained.



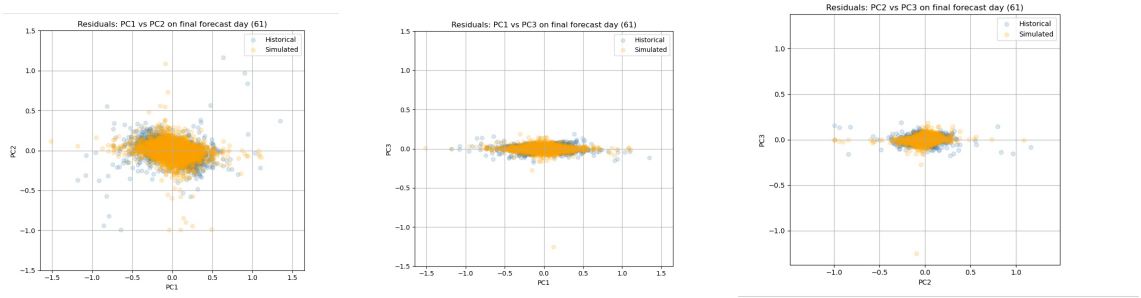
(a) Historical correlation matrix.

(b) Simulated correlation matrix at $t + H$.

Figure 14: Correlation structure of residuals from the PCA-based model. Panel 14a shows the empirical correlation matrix of the historical residuals from the first-differenced latent factors α_{f_1}, α_{f_2}, and α_{f_3}, based on data from 1990–2023. Panel 14b shows the correlation matrix of simulated residuals at forecast horizon $t + H$, where $H = 61$ (corresponding to 61 business days ahead). Residuals are simulated using inverse empirical CDF transformations without the use of a Gaussian copula.

A more detailed inspection is provided in Figure 15, which shows scatterplots of the pairwise residual relationships. The direction and spread of the simulations align well with the historical patterns.

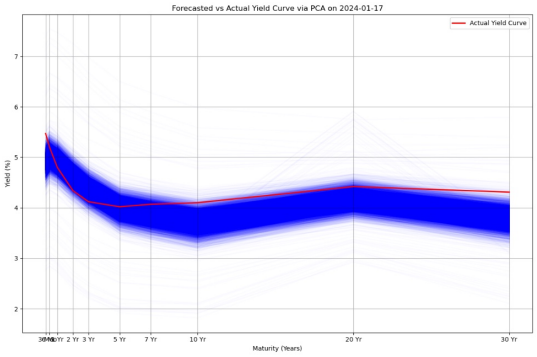
A more detailed inspection is provided in Figure 15, which shows scatterplots of the pairwise residual relationships. The direction and spread of the simulations align well with the historical patterns. However, the formal tail dependence statistics in Appendix A.2 indicate that the simulated residuals tend to understate the joint tail behavior, particularly for the pair α_{f_2} and α_{f_3}.



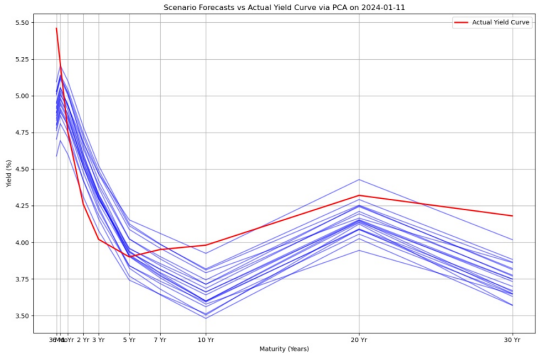
(a) α_{f_1} vs. α_{f_2} **(b)** α_{f_1} vs. α_{f_3} **(c)** α_{f_2} vs. α_{f_3}

Figure 15: Pairwise scatterplots of residuals from the PCA-based model. Each subplot compares historical and simulated residuals for a pair of first-differenced principal component scores: a α_{f_1} vs. α_{f_2}, b α_{f_1} vs. α_{f_3}, and c α_{f_2} vs. α_{f_3}. Historical residuals (1990–2023) are shown in blue; simulated residuals at forecast horizon $t + H$ (with $H = 61$ business days) are shown in orange.

The quality of the simulated scenarios is illustrated in Figure 16. Subfigure 16a shows the full simulation cloud of 10,000 generated yield curves for a single forecast day in Q1 2024. Subfigure 16b presents a random subset of 20 scenarios alongside the actual yield curve, allowing a closer inspection of the individual forecast paths.



(a) Simulated yield curves from the PCA model for a forecast horizon of $t + 10$ business days. The full simulation cloud contains 10,000 generated yield curves for a specific forecast date in Q1 2024. The actual observed yield curve on that day is shown in red for comparison.



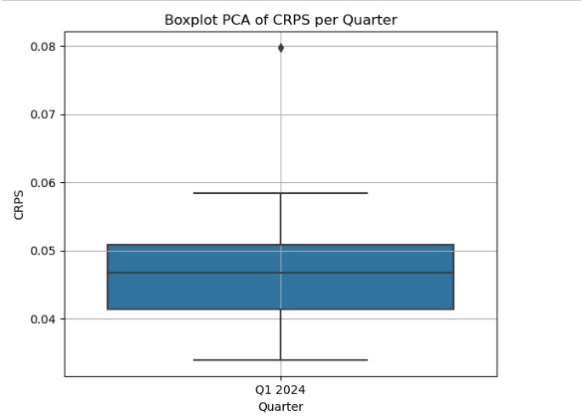
(b) Subset of 20 simulated yield curves randomly selected from the full scenario set shown in (a), plotted alongside the actual yield curve for forecast day $t + 10$ in Q1 2024. This provides a more detailed view of individual forecast paths.

Figure 16: Simulated yield curve scenarios using the PCA model.

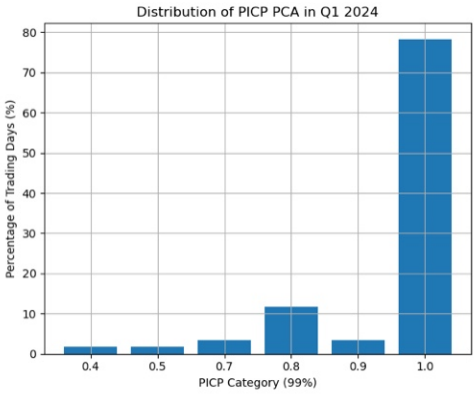
Lastly, the probabilistic performance of the PCA model is evaluated using the CRPS and PICP metrics. Figure 17 shows the distributions of both metrics for the first quarter of 2024.

The CRPS distribution in Figure 17a has a median of approximately 0.0467, with an interquartile range between 0.0414 and 0.0508. The mean CRPS is about 0.0465, with a standard deviation of 0.0078.

Figure 17b shows the daily distribution of the PICP scores (99% interval). While most predictions still provide full coverage, a few maturities fall outside the prediction bands on some days.



(a) Boxplot of CRPS (Continuous Ranked Probability Score) values computed for the PCA model over all forecast days in Q1 2024. Scores are calculated based on simulated yield curve scenarios compared to observed yield curves, as described in Section 4.2. The CRPS is computed for each day and maturity and summarized across the quarter.



(b) Distribution of Prediction Interval Coverage Probability (PICP) values at the 99% confidence level for the PCA model over Q1 2024. For each forecast day, the percentage of maturities where the observed rate falls within the predicted 99% interval is computed.

Figure 17: Evaluation metrics for the PCA model in Q1 2024.

Autoencoder (AE) The Autoencoder (AE) model represents a data-driven approach in which the yield curve is compressed into three latent variables via a non-linear transformation. These so-called **z**-factors are learned through a neural network, without imposing a predefined functional form.

Figure 18 presents the average observed yield curve over the period 1990–2023, together with its reconstruction based on the mean values of the latent z-factors. The close alignment between the two curves indicates that the AE model successfully captures the long-run structure of the yield curve.

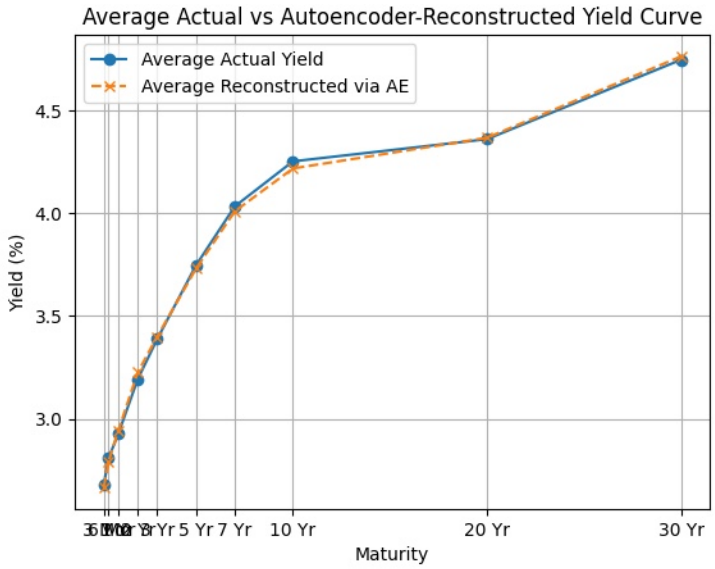


Figure 18: Average observed yield curve over the period 1990–2023 alongside the reconstructed yield curve using the Autoencoder model. The reconstruction is based on the average values of the learned latent representations.

As with the other models, the first differences of the AE factors α_{z_i} were made stationary. Based on the Bayesian Information Criterion (BIC), autoregressive models were then estimated: an AR(2) model for α_{z_1}, an AR(10) model for α_{z_2}, and an AR(1) model for α_{z_3}. The estimated models are given below:

AR(2) model for α_{z_1}

$$\alpha_{z_1,t} = 0.00085 - 0.01697 \cdot \alpha_{z_1,t-1} - 0.00979 \cdot \alpha_{z_1,t-2} + \varepsilon_{1,t}$$

AR(10) model for α_{z_2}

$$\begin{aligned}\alpha_{z_2,t} = &-0.00038 - 0.11558 \cdot \alpha_{z_2,t-1} - 0.04324 \cdot \alpha_{z_2,t-2} - 0.05657 \cdot \alpha_{z_2,t-3} + 0.03038 \cdot \alpha_{z_2,t-4} \\ &- 0.05990 \cdot \alpha_{z_2,t-5} + 0.03923 \cdot \alpha_{z_2,t-6} - 0.01923 \cdot \alpha_{z_2,t-7} - 0.00867 \cdot \alpha_{z_2,t-8} \\ &- 0.02881 \cdot \alpha_{z_2,t-9} + 0.06180 \cdot \alpha_{z_2,t-10} + \varepsilon_{2,t}\end{aligned}$$

AR(1) model for α_{z_3}

$$\alpha_{z_3,t} = -0.00002 - 0.07981 \cdot \alpha_{z_3,t-1} + \varepsilon_{3,t}$$

To evaluate the quality of the simulations, Figure 19 compares the empirical CDF of the residuals from α_{z_1} with that of the simulated values at forecast horizon $t + 6$. The

close agreement suggests that the marginal distribution of the residuals is well reproduced. Additional comparisons for α_{z_2} and α_{z_3} are included in Appendix A.1, and show similarly strong alignment.

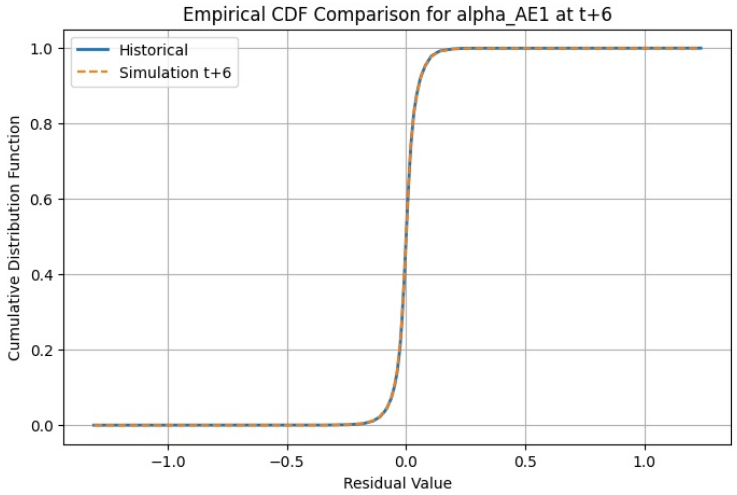
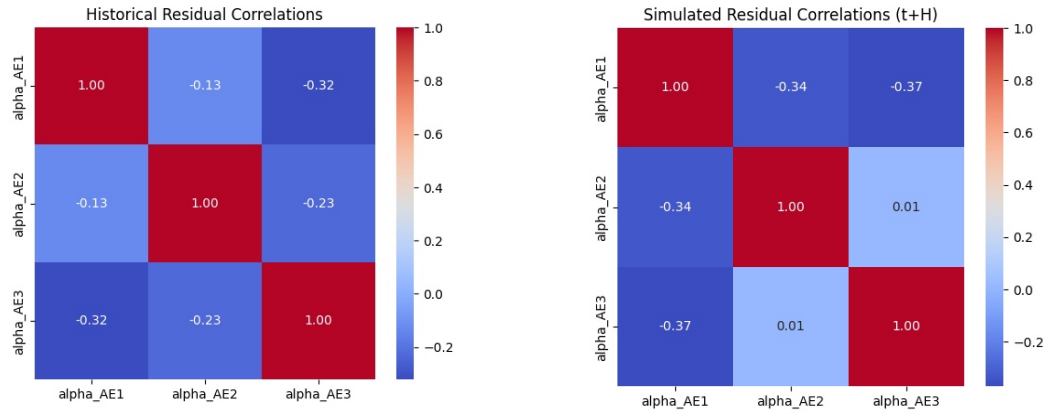


Figure 19: Empirical cumulative distribution function (CDF) of historical residuals for α_{z_1} and the corresponding simulated residuals at forecast horizon $t + 6$.

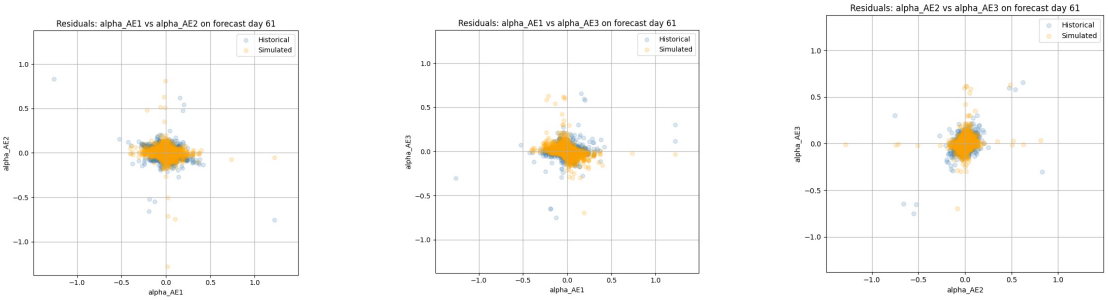
Figure 20 examines the dependence structure of the residuals. Panel 20a presents the historical correlation matrix of the first-differenced latent factors α_{z_1}, α_{z_2}, and α_{z_3}, while Panel 20b shows the corresponding matrix based on simulated residuals at forecast horizon $t + H$ (with $H = 61$ business days). The copula-based simulation procedure generally preserves the correlation structure. The signs of the correlations are consistent, and the strongest relationships, such as the positive correlation between α_{z_2} and α_{z_3} are retained. However, some deviations are evident, the negative historical correlation between α_{z_1} and α_{z_2} is weaker in the simulated data, and the correlation between α_{z_1} and α_{z_3} moves closer to zero. These differences suggest a slight underestimation of cross-factor dependence in the simulations.



(a) Historical correlation matrix. **(b)** Simulated correlation matrix at $t + H$.

Figure 20: Correlation structure of residuals from the AE-based model. Panel 20a shows the empirical correlation matrix of the historical residuals from the first-differenced latent factors α_{z_1}, α_{z_2}, and α_{z_3}, based on data from 1990–2023. Panel 20b shows the correlation matrix of simulated residuals at forecast horizon $t + H$, where $H = 61$ (corresponding to 61 business days ahead). Residuals are simulated using inverse empirical CDF transformations, with dependence preserved with use of a Gaussian copula.

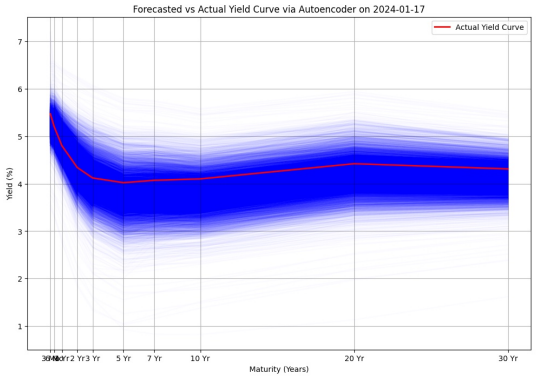
A visual inspection of pairwise residual relationships is shown in Figure 21. The scatterplots indicate that both the dispersion and directional structure of the historical residuals are well approximated by the simulations. However, the tail dependence statistics in Appendix A.2 reveal that the strength of joint extremes is notably weaker in the simulated data—especially for α_{z_2} and α_{z_3}.



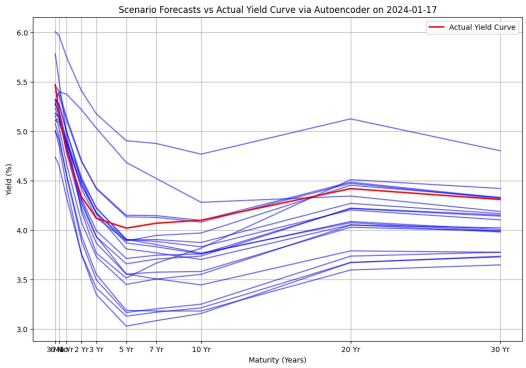
(a) α_{z_1} vs. α_{z_2} **(b)** α_{z_1} vs. α_{z_3} **(c)** α_{z_2} vs. α_{z_3}

Figure 21: Pairwise scatterplots of residuals from the Autoencoder model. Each subplot compares historical and simulated residuals for a pair of first-differenced latent factors: 21a α_{z_1} vs. α_{z_2}, 21b α_{z_1} vs. α_{z_3}, and 21c α_{z_2} vs. α_{z_3}. Historical residuals (1990–2023) are shown in blue; simulated residuals at forecast horizon $t + H$ (with $H = 61$ business days) are shown in orange.

The simulated yield curve scenarios are presented in Figure 22. Subfigure 22a displays the full simulation cloud on a selected day in Q1 2024, while Subfigure 22b compares 20 randomly selected scenarios with the observed curve.



(a) Simulated yield curves from the Autoencoder model for a forecast horizon of $t+10$ business days. The full simulation cloud contains 10,000 generated yield curves for a specific forecast date in Q1 2024. The actual observed yield curve on that day is shown in red for comparison.

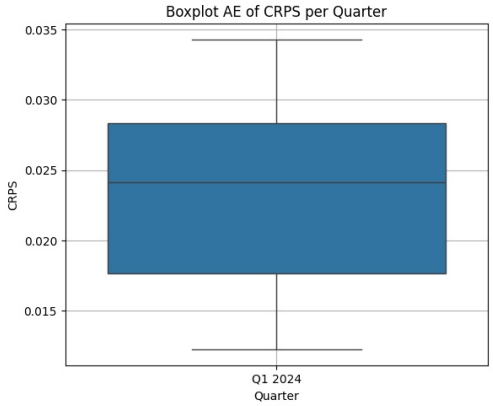


(b) Subset of 20 simulated yield curves randomly selected from the full scenario set shown in (a), plotted alongside the actual yield curve for forecast day $t + 10$ in Q1 2024. This provides a more detailed view of individual forecast paths.

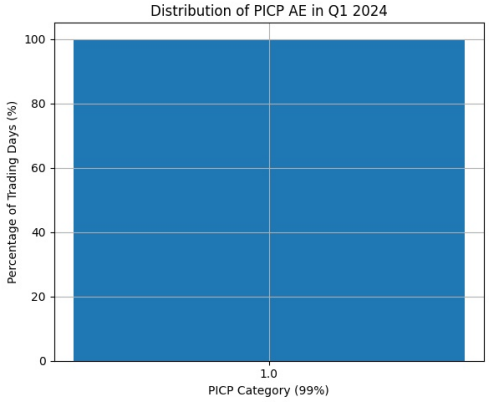
Figure 22: Simulated yield curve scenarios using the AE model.

Finally, Figure 23 evaluates the probabilistic performance of the AE model. The CRPS distribution in Subfigure 23a shows a median of approximately 0.0241, with an interquartile range from roughly 0.0177 to 0.0283. The mean CRPS is about 0.0237, with a standard deviation of approximately 0.0060.

The PICP distribution in Subfigure 23b shows that all forecasts continue to provide full coverage within the 99% prediction interval.



(a) Boxplot of CRPS (Continuous Ranked Probability Score) values computed for the Autoencoder model over all forecast days in Q1 2024. Scores are calculated based on simulated yield curve scenarios compared to observed yield curves, as described in Section 4.2. The CRPS is computed for each day and maturity and summarized across the quarter.



(b) Distribution of Prediction Interval Coverage Probability (PICP) values at the 99% confidence level for the AE model over Q1 2024. For each forecast day, the percentage of maturities where the observed rate falls within the predicted 99% interval is computed.

Figure 23: Evaluation metrics for the AE model in Q1 2024.

5.2 Performance Across Rolling Forecasts

To evaluate the robustness and generalizability of the three models, a rolling forecast structure was implemented. For each model, Nelson-Siegel, PCA, and Autoencoder, training was conducted using four different historical windows: starting in 1990, 2000, 2010, and 2020. From each of these starting points, the models produced quarterly forecasts across 2024, following the procedure outlined in Section 4.6

Table 3 reports the mean CRPS values across all forecast quarters and training periods for each of the three models. Several patterns are immediately apparent. The Autoencoder achieves the lowest CRPS values in the majority of cases and generally demonstrates stronger performance compared to Nelson-Siegel and Principal Component Analysis. This suggests that the AE model produces probabilistic forecasts that, on average, lie closer to the realized yield outcomes. However, this advantage is not universal. In a few settings, notably in Q2 and Q4 for select training windows, PCA or NS slightly outperform AE, indicating that no single model dominates in all scenarios.

A second observation is the clear deterioration in forecast quality during the third quarter (Q3) across all models. CRPS values rise significantly in Q3 compared to other quarters, particularly when models are trained on shorter and more recent windows (e.g., starting in 2010 or 2020). This may reflect structural features of Q3, such as reduced market liquidity or delayed policy actions, which tend to increase uncertainty and reduce the predictability of interest rate dynamics during this period.

Third, the results indicate that models trained on shorter and more recent datasets, especially those starting in 2010 or 2020, generally perform worse across all quarters. This trend is most visible in PCA and NS, where CRPS scores increase compared to models trained on the longer 1990 or 2000 windows. Conversely, training on longer and more diverse economic periods tends to yield more robust out-of-sample performance.

When comparing PCA and NS, PCA tends to outperform NS in terms of CRPS, particularly in Q2 and Q4. This suggests that PCA more effectively captures variation in the term structure under certain conditions. However, both linear models are frequently outperformed by AE, especially when trained on more extensive historical data.

	NS – Mean CRPS				PCA – Mean CRPS				AE – Mean CRPS			
Quarter	1990	2000	2010	2020	1990	2000	2010	2020	1990	2000	2010	2020
Q1	0.0452	0.0392	0.0487	0.0510	0.0465	0.0460	0.0462	0.0459	0.0237	0.0243	0.0481	0.0796
Q2	0.0493	0.0349	0.0534	0.0622	0.0368	0.0411	0.0576	0.0550	0.0281	0.0392	0.0592	0.0551
Q3	0.0811	0.0765	0.1350	0.1641	0.0659	0.0934	0.1423	0.1503	0.0492	0.0803	0.1071	0.1049
Q4	0.0908	0.0612	0.0893	0.0922	0.0512	0.0927	0.0894	0.0876	0.0589	0.0499	0.0896	0.1000

Table 3: Mean CRPS values across forecast quarters (Q1–Q4) and training windows starting in 1990, 2000, 2010, and 2020, for the NS, PCA, and AE models.

PICP Category	**1990**				**2000**				**2010**				**2020**			
	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
NS																
1.0	98.33	92.06	84.38	98.39	93.33	96.83	95.31	88.71	96.67	92.06	31.25	91.94	98.94	96.83	31.81	98.39
0.9	1.67	7.94	15.62	-	1.67	1.59	1.56	8.06	3.33	6.35	10.94	4.84	1.67	3.17	23.44	1.61
0.8	-	-	-	-	3.33	1.59	3.12	1.61	-	1.59	21.88	3.23	-	-	9.38	-
0.7	-	-	-	1.61	-	-	-	-	-	-	4.69	-	-	-	7.81	-
≤ 0.6	-	-	-	-	1.61	-	-	1.61	-	-	31.25	-	-	-	26.56	-
PCA																
1.0	78.33	84.13	87.50	77.42	90.00	95.24	96.88	85.48	91.67	90.48	36.06	74.19	98.23	100	40.62	100
0.9	3.33	4.76	4.69	9.68	1.67	-	1.56	9.68	3.33	1.59	12.50	11.29	1.67	-	23.44	-
0.8	11.67	7.94	4.69	9.68	5.00	3.17	1.56	3.23	1.67	1.59	20.31	9.68	-	-	9.38	-
0.7	3.33	1.59	1.56	-	3.33	1.59	-	-	3.33	6.35	12.50	4.84	-	-	18.75	-
≤ 0.6	3.33	1.59	1.56	3.23	-	-	-	1.61	-	-	15.62	-	-	-	7.81	-
AE																
1.0	100	92.06	98.44	90.32	100	100	96.88	100	100	93.65	65.62	85.48	98.33	98.41	79.69	100
0.9	-	7.94	1.56	6.45	-	-	3.12	-	-	1.59	6.25	8.06	1.67	1.59	12.50	-
0.8	-	-	-	-	-	-	-	-	-	4.76	20.31	4.84	-	-	7.81	-
0.7	-	-	-	-	-	-	-	-	-	-	4.69	1.61	-	-	-	-
≤ 0.6	-	-	-	3.23	-	-	-	-	-	-	3.12	-	-	-	-	-
.																

Table 4: Percentage of trading days per PICP category (rows), forecast quarter (columns), and training window (year blocks) for NS, PCA, and AE models.

Table 4 summarizes the distribution of daily PICP values across forecast quarters and training windows for each of the three models. The rows reflect categories of daily PICP values, where a value of 1.0 denotes that on that particular day, the realized yield for each of the ten maturities fell within the corresponding 99% prediction interval. Lower categories indicate that one or more maturities fell outside the predicted range.

The Autoencoder (AE) exhibits the highest concentration of days with PICP = 1.0 across nearly all quarters and training windows, particularly when trained on data starting in 1990, 2000, or 2020. This indicates that the AE model consistently produces prediction intervals that are well calibrated to the empirical variation in yields. Even in quarters where other models show difficulty maintaining high PICP, such as Q3, the Autoencoder still achieves a relatively high proportion of days where all maturities fall within their 99%

prediction intervals. This suggests that AE is better able to maintain calibration when forecasting becomes more challenging relative to other quarters.

By contrast, the Nelson-Siegel (NS) and PCA models show greater variation in their daily PICP distributions. For NS, training windows from 1990 and 2000 result in a high proportion of days where all maturities lie within the predicted intervals. However, this performance deteriorates for windows starting in 2010 and especially in 2020. In Q3 of those years, the proportion of days with full containment drops substantially, with some days falling below a PICP of 0.6, meaning fewer than six of the ten maturities were accurately bounded. This pattern suggests that the NS model may underestimate uncertainty during structurally different or turbulent regimes.

PCA follows a similar trend: relatively strong calibration under early training windows, but declining performance in later periods. The drop in fully contained days is particularly visible in Q3 and Q4 of 2010 and 2020, where a growing number of observations fall into intermediate or lower PICP categories. In these instances, a lower share of maturities falls within the predicted 99% intervals, indicating reduced calibration performance relative to earlier training periods. The consistently lower PICP values in Q3 across all models support the earlier observation from the CRPS results: this quarter poses structural forecasting challenges. The concurrent decline in both accuracy (CRPS) and calibration (PICP) implies that models trained on recent data may not fully capture the heightened variability and complexity typical of this period.

Overall, the AE model demonstrates relatively strong calibration across a wide range of conditions. Especially when trained on longer historical samples, AE tends to maintain higher proportions of days with full interval containment compared to the other models. While NS and PCA perform adequately in many settings, their calibration appears more sensitive to the choice of training window and forecast quarter.

6 Discussion

This study addresses the central research question: *How do Nelson-Siegel, Principal Component Analysis, and Autoencoders perform in generating plausible scenarios for the U.S. Treasury yield curve?*

To evaluate how plausible the generated scenarios are, this study not only considers forecast accuracy and calibration via CRPS and PICP, but also incorporates the quality of simulated residuals, their underlying correlation structure, and robustness across different economic regimes via a rolling forecast framework with varying training windows.

6.1 Preservation of Residual Structure

A crucial part of generating realistic yield curve scenarios is simulating residuals that preserve the dependency structure between latent components. In all models, the marginal distributions of the residuals were handled in the same way: empirical cumulative distribution functions (ECDFs) were used to transform the simulated values back into the original scale. The only difference between the approaches lies in how the joint dependency between the components was captured.

For the Nelson-Siegel and Autoencoder models, a copula-based approach was used. Specifically, a Gaussian copula was fitted to the empirical residuals after transforming them to uniform distributions. This allows for flexible modelling of dependence. In contrast, the PCA model relied directly on the historical correlation matrix of the residuals. Multivariate normal draws were generated using this correlation structure, and these were then transformed through the ECDFs. Although this is conceptually related to a Gaussian copula, no explicit copula model was fitted.

Empirical results show that, in terms of tail behavior, all three models generate residuals that visually resemble the historical data, including the spread, direction, and clustering patterns seen in scatterplots. However, the simulated correlation matrices reveal some distortion. In particular, the copula-based methods (NS and AE) tend to slightly overstate the correlation between the first and second components compared to the historical residuals. By contrast, PCA maintains a more neutral correlation pattern, with modest negative correlations remaining relatively stable across simulation and historical data. Still, tail dependence results in Appendix A.2 suggest that only the Nelson-Siegel model captures joint extreme events reliably, while PCA and AE slightly underestimate such dependencies.

These findings suggest that while the Gaussian copula effectively preserves broad joint behavior, it may impose overly strong dependence in some latent structures. Alternative copula families could potentially yield better calibration of the dependence structure. For example, alternative copula families such as the t-copula may offer stronger tail dependence

and better flexibility in capturing asymmetric co-movements. Future work could explore such models to enhance the realism of multivariate residual simulations.

6.2 Overall model performace

The following paragraphs evaluate how each model performs in terms of forecast accuracy and calibration, highlighting both strengths and limitations.

Autoencoder as Best Performing Model Across the full rolling forecast setup, covering all quarters and training windows, the Autoencoder (AE) generally emerges as the best-performing model. It consistently outperforms the other approaches on both key evaluation metrics: the Continuous Ranked Probability Score (CRPS) and the Prediction Interval Coverage Probability (PICP).

In terms of CRPS, AE achieves the lowest scores in the vast majority of quarters and training years. This indicates that its simulated distributions are typically the most concentrated around the actual observed interest rates, suggesting a high degree of precision in capturing yield curve dynamics.

Regarding PICP, AE frequently attains a score of exactly 1.0, and 0.9 in most of the remaining cases. These values imply that nearly all realized outcomes fall within the model's 99% prediction interval, reflecting a high level of calibration. AE remains robust even in volatile economic periods (e.g., post-2020), suggesting the model is well-suited to capture non-linear dynamics and structural breaks better than linear alternatives.

It should be noted, however, that PCA and Nelson-Siegel occasionally outperform the Autoencoder in specific quarters or training windows, particularly in terms of either CRPS or PICP. This indicates that the superiority of the AE model is not absolute across all scenarios.

PCA vs. Nelson-Siegel : Accuracy vs. Calibration Both PCA and Nelson-Siegel use three latent factors to model the yield curve, but their performance differs in meaningful ways. PCA generally achieves lower CRPS scores than NS, particularly in more volatile quarters such as Q3 and Q4. This indicates that PCA is more effective at concentrating the predicted distribution around the realized rate, possibly due to its data-driven structure, which captures variation in the term structure efficiently.

In contrast, Nelson-Siegel tends to achieve slightly higher PICP scores, indicating that its prediction intervals more frequently succeed in capturing the actual observed yields. This suggests that the model is more conservative in its uncertainty estimation, producing wider intervals that are better calibrated to the true variability in the data.

This highlights a common trade-off: PCA produces more concentrated forecasts, whereas NS provides more conservative but better-calibrated intervals. In both respects, however, the Autoencoder still performs best overall.

Structural Weakness in Q3 One consistent pattern in the results is that all models perform significantly worse in the third quarter (Q3):

- CRPS scores are highest in Q3 across models, indicating a decrease in forecast accuracy.
- PICP scores also tend to drop during Q3, meaning the models are more likely to miss the actual realization outside of the predicted intervals.

This dip in performance may be linked to structural features of Q3, such as lower liquidity during summer months, a pause in monetary policy activity before Q4 decisions, or post-crisis uncertainty. Haubrich and Dombrosky (1996) also show that the predictive power of the yield curve deteriorates during such transitional periods.

Economic Regime Effects: Impact of Recent Training Periods Models trained on more recent periods, such as those starting in 2010 or 2020, tend to perform less in terms of PICP and CRPS, particularly in the third and fourth quarters. These periods reflect structurally atypical macroeconomic environments, including prolonged phases of low interest rates, post-crisis recovery dynamics following the 2008 financial crisis, unprecedented monetary interventions during the COVID-19 pandemic, and broader structural shifts in financial markets. Forecast accuracy, as measured by CRPS, declines under these conditions for all models, most notably for the Autoencoder, highlighting the difficulty of scenario generation based on unstable and relatively short historical windows. Nevertheless, AE still appears to be the most robust model in these settings, likely due to its ability to capture complex, non-linear patterns in limited data. This finding underscores the importance of using rich and representative training periods when building scenario models. Haubrich and Dombrosky (1996) similarly note that yield curve predictability weakens during regime shifts in macroeconomic and financial conditions.

6.3 Recommendations

Autoencoders demonstrate strong overall performance and should be considered in applications where predictive precision and robust scenario generation are critical. However, due to their complex, non-transparent structure, they may be less suitable in settings that require model interpretability or regulatory transparency. In such cases, Principal Component Analysis or Nelson-Siegel may be preferred, as they offer clearer economic interpretation of latent components, albeit with some trade-offs in forecast sharpness or

calibration.

Furthermore, the choice of dependency modeling plays a critical role in the realism of simulated scenarios. While Gaussian copulas offer flexibility, they may impose overly strong dependence between latent factors. Exploring alternative copula families, such as the t-copula or vine copulas, may improve the fidelity of the residual correlation structure, particularly in capturing asymmetric or tail dependence.

Finally, models trained on short or unstable economic periods tend to perform less reliably. Using longer and more representative historical samples is therefore advisable to enhance scenario robustness across regimes.

7 Conclusion

This study evaluated the suitability of three models, Nelson-Siegel, Principal Component Analysis, and Autoencoders for generating plausible scenarios for the U.S. Treasury yield curve. Using more than 30 years of U.S. Treasury data across ten maturities, each yield curve was reduced to a three-factor latent representation. These latent series were differenced to ensure stationarity, modeled with autoregressive processes, and extended forward using simulated residuals. The simulation procedure explicitly preserved both marginal distributions and the dependence structure between latent components. Full yield curves were reconstructed from the simulated latent paths.

To evaluate the plausibility of the generated scenarios, two metrics were applied: the Continuous Ranked Probability Score (CRPS), measuring how closely the forecasts clustered around realized values, and the Prediction Interval Coverage Probability (PICP), indicating the proportion of observed yields falling within model-generated intervals. These metrics were assessed within a rolling forecast framework, allowing for comparison across multiple economic periods and training windows.

The results show that the Autoencoder model performs most strongly across the majority of configurations. It produces narrower forecast distributions that remain close to observed yield levels, while still retaining sufficient range to contain realized values. This advantage is particularly evident when trained on longer and more diverse historical samples, highlighting the model's ability to capture non-linear patterns in the data.

PCA tends to achieve more concentrated predictive distributions, as reflected in lower CRPS scores, particularly during volatile quarters. Nelson-Siegel, on the other hand, produces wider but more reliably calibrated intervals, resulting in slightly higher PICP scores. These results reflect a typical trade-off between accuracy and coverage in linear models, with each method offering strengths under different evaluation criteria.

All models show weaker performance during the third quarter of the year, regardless of training window. This suggests that seasonal factors such as low market activity or delayed policy shifts introduce greater uncertainty, making reliable forecasting more difficult. Additionally, models trained on shorter or more recent samples tend to perform worse, especially when structural breaks dominate recent interest rate behavior.

Overall, this study demonstrates that data-driven, non-linear methods such as Autoencoders can substantially improve the generation of realistic interest rate scenarios, provided that sufficient historical information is available and dependence between latent factors is carefully modeled. While such models are less transparent, they offer significant benefits in settings where simulation realism is critical. Future research could extend this framework by incorporating alternative dependence structures, such as t- or vine copulas, or by

including macro-financial covariates to broaden the applicability of scenario generation.

References

Annaert, J., A. G. P. Claes, M. J. K. De Ceuster, and H. Zhang (n.d.). "Estimating the yield curve using the Nelson–Siegel model: A ridge regression approach". Unpublished manuscript. Accessed June 2025.

Bank, D., N. Koenigstein, and R. Giryes (2021). *Autoencoders*. `https://arxiv.org/abs/2003.05991`. arXiv preprint arXiv:2003.05991. Tel Aviv University. Accessed July 1, 2025.

Boswijk, P. and P. H. Franses (1992). "Dynamic specification and cointegration". In: *Oxford Bulletin of Economics and Statistics* 54.3, pp. 369–381.

Burnham, K. P. and D. R. Anderson (2004). "Multimodel inference: Understanding AIC and BIC in model selection". In: *Sociological Methods Research* 33.2, pp. 261–304.

Diebold, F. X. and C. Li (2006). "Forecasting the term structure of government bond yields". In: *Journal of Econometrics* 130.2, pp. 337–364.

Duffee, G. R. (2002). "Term premia and interest rate forecasts in affine models". In: *The Journal of Finance* 57.1, pp. 405–443.

Estrella, A. and F. S. Mishkin (June 1996). "The yield curve as a predictor of U.S. recessions". In: *Current Issues in Economics and Finance* 2.7.

Gneiting, T. and M. Katzfuss (2014). "Probabilistic forecasting". In: *Annual Review of Statistics and Its Application* 1.1, pp. 125–151.

Gneiting, T. and A. E. Raftery (2007). "Strictly proper scoring rules, prediction, and estimation". In: *Journal of the American Statistical Association* 102.477, pp. 359–378.

Gusev, A., A. Chervyakov, A. Alexeenko, and E. Nikulchev (2023). "Particle swarm training of a neural network for the lower upper bound estimation of the prediction intervals of time series". In: *Mathematics* 11.19, p. 4342.

Harris, R. I. D. (1995). *Using cointegration analysis in econometric modelling*. London: Harvester Wheatsheaf / Prentice Hall.

Haubrich, J. G. and A. M. Dombrosky (Feb. 1996). *Predicting real growth using the yield curve*. Federal Reserve Bank of Cleveland Economic Commentary. Accessed July 1, 2025.

Hersbach, H. (Oct. 2000). "Decomposition of the continuous ranked probability score for ensemble prediction systems". In: *Weather and Forecasting* 15.5, pp. 559–570.

Ibáñez, F. (2015). *Calibrating the Dynamic Nelson-Siegel Model: A Practitioner Approach.* `https://mpra.ub.uni-muenchen.de/68377/`. Central Bank of Chile. Accessed July 1, 2025.

Jolliffe, I. T. and J. Cadima (2016). "Principal component analysis: A review and recent developments". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 374.2065, p. 20150202.

Kim, M. and J. Song (2020). "Unsupervised deep neural network for near-real-time damage assessment of structures subject to earthquake excitations". In: *Proceedings of the Seventh Asian-Pacific Symposium on Structural Reliability and Its Applications (APSSRA2020).* Tokyo, Japan.

Lopez, J. H. (1997). "The power of the ADF test". In: *Economics Letters* 57.1, pp. 5–10.

Nelson, C. R. and A. F. Siegel (Oct. 1987). "Parsimonious modeling of yield curves". In: *The Journal of Business* 60.4, pp. 473–489.

Oprea, A. (2022). "The use of principal component analysis (PCA) in building yield curve scenarios and identifying relative-value trading opportunities on the Romanian government bond market". In: *Journal of Risk and Financial Management* 15.6, p. 247.

Pathirage, C. S. N., J. Li, L. Li, H. Hao, W. Liu, and P. Ni (2018). "Structural damage identification based on autoencoder neural networks and deep learning". In: *Engineering Structures* 172, pp. 133–145.

Redfern, D. and D. McLean (Aug. 2014). *Principal Component Analysis for Yield Curve Modelling: Reproduction of Out-of-Sample Yield Curves.* Research Report. Accessed July 1, 2025. Moody's Analytics.

Reserve Bank of Australia (n.d.). *Bonds and the Yield Curve.* `https://rba.gov.au/education/`. Accessed July 1, 2025.

Sokol, A. (2022). *Autoencoder market models for interest rates.* `https://ssrn.com/abstract=4300756`. CompatibL. Available at SSRN: https://ssrn.com/abstract=4300756. Accessed July 1, 2025.

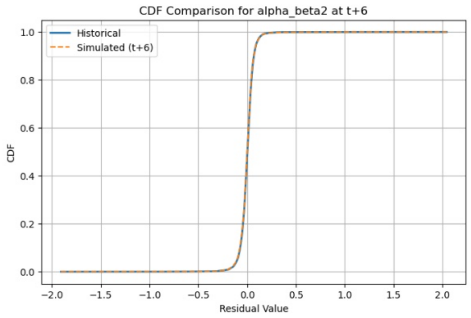
Suimon, Y., H. Sakaji, K. Izumi, and H. Matsushima (2020). "Autoencoder-based three-factor model for the yield curve of Japanese government bonds and a trading strategy". In: *Journal of Risk and Financial Management* 13.4, p. 82.

Trivedi, P. K. and D. M. Zimmer (2011). *Copula modeling: An introduction for practitioners.* Vol. 1. Foundations and Trends in Econometrics 1. Now Publishers Inc.

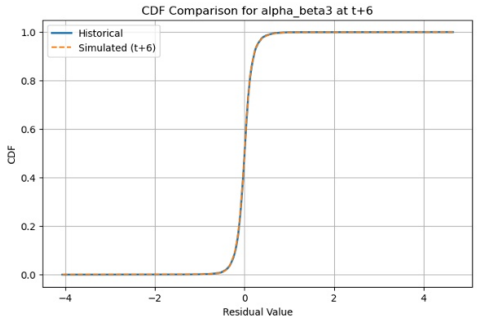
Tsay, R. S. (2010). *Analysis of financial time series.* 3rd. Hoboken, NJ: John Wiley Sons.

A Appendix A

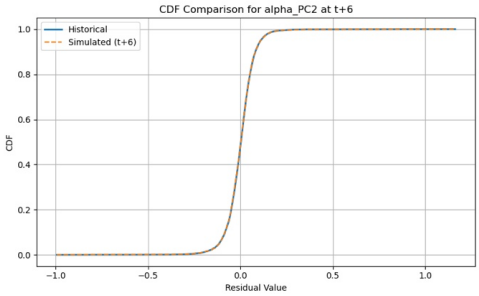
A.1



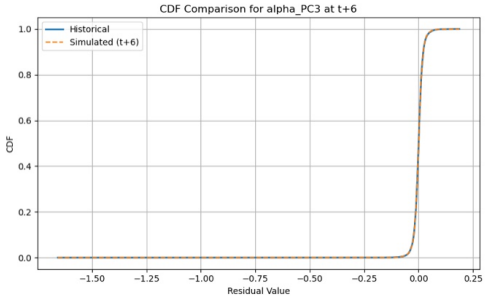
(a) NS – $\alpha_{\beta 2}$ at $t+6$



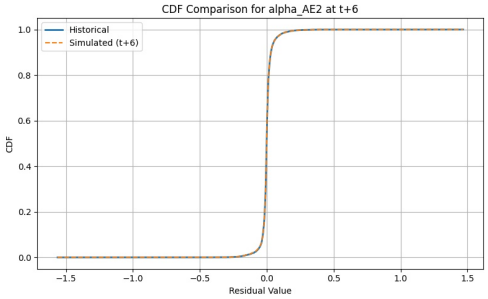
(b) NS – $\alpha_{\beta 3}$ at $t+6$



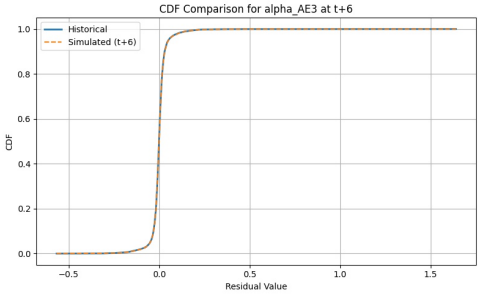
(c) PCA – $\alpha_{\mathrm{PC}2}$ at $t+6$



(d) PCA – $\alpha_{\mathrm{PC}3}$ at $t+6$



(e) AE – $\alpha_{\mathrm{AE}2}$ at $t+6$



(f) AE – $\alpha_{\mathrm{AE}3}$ at $t+6$

Figure 24: Comparison of empirical CDFs for selected latent residuals at forecast horizon $t+6$. Each plot compares the historical distribution with the simulated one for one of the three models.

A.2

Pair	Source	Lower	Upper
$\alpha_{\beta 1}$ vs. $\alpha_{\beta 2}$	Historical	0.001	0.002
	Simulated	0.000	0.000
$\alpha_{\beta 1}$ vs. $\alpha_{\beta 3}$	Historical	0.005	0.005
	Simulated	0.003	0.003
$\alpha_{\beta 2}$ vs. $\alpha_{\beta 3}$	Historical	0.003	0.005
	Simulated	0.002	0.001

Table 5: Tail Dependence (5%) – Nelson-Siegel Residuals

Pair	Source	Lower	Upper
α_{PC1} vs. α_{PC2}	Historical	0.004	0.002
	Simulated	0.001	0.001
α_{PC1} vs. α_{PC3}	Historical	0.006	0.007
	Simulated	0.002	0.002
α_{PC2} vs. α_{PC3}	Historical	0.010	0.009
	Simulated	0.005	0.004

Table 6: Tail Dependence (5%) – PCA Residuals

Pair	Source	Lower	Upper
α_{AE1} vs. α_{AE2}	Historical	0.008	0.008
	Simulated	0.001	0.001
α_{AE1} vs. α_{AE3}	Historical	0.009	0.009
	Simulated	0.001	0.001
α_{AE2} vs. α_{AE3}	Historical	0.032	0.034
	Simulated	0.019	0.018

Table 7: Tail Dependence (5%) – Autoencoder Residuals