Journal of Economic Theory and Econometrics, Vol. 35, No. 4, Dec. 2024, 33–54

Prediction and Prediction Intervals in Exchange Rates: A Generative Approach Using Variational Autoencoders*

Soohyon Kim†

Abstract In this study, we explore the use of generative models for time series prediction and construction of prediction intervals, addressing the challenge of quantifying uncertainty in deep learning models. Specifically, we employ a Variational Autoencoder (VAE), a form of a Bayesian neural network also a part of generative AI models, to model and generate latent factors for the exchange rates of ten currencies. These latent factors enable the approximate reconstruction of the exchange rate series through the decoder part of VAE. By generating a thousand sets of latent factors and reconstructing exchange rates, we create prediction intervals through a Multi-Layer Perceptron (MLP) applied to the reconstructed series. This approach provides valuable insights into evaluation of the uncertainty associated with time series predictions using neural networks.

Keywords Time series prediction, prediction interval, generative model, uncertainty measure.

JEL Classification C45, F17, F31.

*I express my sincere gratitude to the two anonymous reviewers and the editor for their insightful comments and constructive feedback, which have greatly contributed to improving the quality of this paper. This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea (NRF-2022S1A5A8056688).

†Department of Economics, Chonnam National University, 77 Yongbong-ro, Bukgu, Gwangju, Republic of Korea 61186. E-mail: soohyon.kim@jnu.ac.kr.

Received October 21, 2024, Revised November 1, 2024, Accepted December 16, 2024

1. INTRODUCTION

The accurate prediction of time series is a fundamental challenge in various fields, especially in economics and finance. With the growing complexity and volume of data, traditional statistical methods often struggle to capture the intricate patterns and dependencies present in time series, especially when faced with non-linearity, non-stationarity, and volatility with high-frequencies. This has led to the increased adoption of deep learning models, which have shown significant promise in capturing complex temporal dynamics and improving predictive performance.

However, despite their successes, deep learning models are often criticized for their "black-box" nature, which obscures the ability to quantify the uncertainty inherent in their predictions. In many application, such as time series forecasting, understanding and quantifying the uncertainty—both the inherent variability in the data, aleatoric uncertainty and the uncertainty due to the model's limitations, epistemic uncertainty (Abraham and Ledolter, 2009) - of predictions is as important as the predictions themselves. To address this challenge, we turn to generative models, which have gained prominence in recent years for their ability to model complex data distributions and generate new, realistic data points. Among these, Variational Autoencoders (VAEs) have emerged as a powerful tool for learning low-dimensional representations (latent factors) of high-frequency data. VAEs combine the flexibility of deep learning with the rigorous probabilistic framework of Bayesian inference, making them well-suited for tasks where uncertainty quantification is essential. By modelling the underlying distribution of the data, VAEs allow for the construction of prediction intervals that reflect the true variability in predictions, offering a more comprehensive understanding of the confidence we can place in model outputs.

In this study, we apply VAEs to construct prediction intervals for daily exchange rate forecasting, presenting a significant advancement over traditional point forecasts by directly addressing prediction uncertainty. Exchange rates are inherently complex and nonlinear, necessitating models that can represent a range of possible future outcomes rather than a single forecast. Distinct from prior applications of VAEs in fields like image and text generation, which emphasize precise reconstructions, our approach leverages the VAE to generate a set of approximately reconstructed series, each representing a plausible trajectory of future exchange rates.

By training a VAE on the exchange rates of ten major currencies, we extract essential latent factors that capture potential core patterns in the historical data. These latent factors are then used to produce multiple reconstructed series, which serve as diverse inputs for subsequent prediction modeling. The reconstructed series are fed into a Multi-Layer Perceptron (MLP) model, trained specifically for short-term exchange rate forecasting. The MLP processes each VAE-generated reconstruction, providing a range of point forecasts across the ensemble of inputs. This range allows us to calculate prediction intervals that quantify the forecast uncertainty, reflecting both data variability and model-based uncertainty.

This combined use of VAE and MLP introduces a novel framework for exchange rate forecasting by moving beyond point predictions to construct robust prediction intervals. From an econometric perspective, this approach offers an advancement in uncertainty quantification by simulating a range of latent representations that approximate the stochastic nature of exchange rates. In contrast to traditional econometric models, which typically rely on bootstrapping or statistical assumptions for interval estimation, our method directly incorporates the VAE's generative nature to capture a broader scope of potential outcomes, even under highly variable market conditions. Consequently, this framework enriches econometric analysis by enabling a probabilistic view of future exchange rates.

Positioning this study within the broader landscape of deep learning research, our approach underscores the potential of generative models like VAE to move beyond point predictions and facilitate uncertainty quantification across a variety of macroeconomic and financial forecasting tasks. To our knowledge, this is among the first studies to apply VAEs specifically for the purpose of constructing prediction intervals in financial time series forecasting, offering a unique contribution to the literature on deep learning applications in econometrics. The remainder of this paper is structured as follows: Section 2 reviews relevant literature on time series prediction and uncertainty quantification. Section 3 details the methodology, including the architecture of the VAE and MLP models, and presents experimental results, demonstrating the effectiveness of our approach in exchange rate forecasting and interval construction. Finally, Section 4 discusses the implications of our findings and concludes with suggestions for future research.

2. LITERATURE REVIEW

In the realm of econometrics, time series forecasting is one of critical areas of study, particularly due to its applications in economic modeling and decision-making. Various approaches have been developed to enhance the accuracy of predictions and to assess the uncertainties associated with these forecasts. One of the foundational approaches to time series forecasting is the Autoregressive Integrated Moving Average (ARIMA) model, which has been extensively studied for its flexibility in handling univariate time series data (Hamilton, 1994; Enders, 2014). In addition to traditional models like ARIMA, modern techniques such as fuzzy time series models, including K-means clustering have been reported to show promising results in enhancing prediction performance (Zhang *et al.*, 2022).

In addition to traditional econometric models, machine learning approaches have gained traction in time series forecasting due to their ability to capture complex patterns in data. Techniques such as recurrent neural networks (RNNs) and deep learning models have been reported to perform well in certain multi-step ahead forecasting tasks. These models can learn from extensive datasets and adapt to changing patterns, which can make them effective in dynamic environments like financial markets. However, a significant challenge remains in quantifying the uncertainty associated with these predictions. (Flunkert *et al.*, 2020)

The construction of confidence bands and prediction intervals in time series analysis is a critical area of research where accurate forecasting and assessing uncertainty is essential for the prediction model performance. Brockwell and Davis (2002) discuss the theoretical underpinnings of time series models and the importance of uncertainty quantification in forecasts. They emphasize that confidence bands are crucial for understanding the reliability of predictions, particularly in the context of autoregressive models. Abraham and Ledolter (2009) elaborates on statistical techniques for constructing prediction intervals, highlighting the role of statistical inference in forecasting. This work underscores the necessity of robust methods for interval estimation, particularly when dealing with non-stationary time series data. Recent advancements in the construction of prediction intervals have been made by Karmakar *et al.* (2021), who explore long-term prediction intervals with many covariates. Their work highlights the challenges of constructing simultaneous prediction intervals for multivariate time series, particularly in applications such as electricity price forecasting. Many works in statistics on bootstrap prediction intervals for autoregressive time series further illustrates the effectiveness of resampling techniques in interval estimation. The bootstrap method is particularly advantageous as it does not rely on strict parametric assumptions about the underlying data distribution, making it suitable for a wide range of time series applications. (Davison and Hinkley, 2013; Franco *et al.*, 2001; Hwang and Shin, 2010; Novoa and Mendez, 2009) Their findings indicate that bootstrap methods can provide more reliable prediction intervals compared to traditional methods, particularly in the presence of model uncertainty. Deep learning models can capture complex relationships and nonlinearities that traditional approaches may not be able to address. In particular, neural networks have been shown to provide more robust and reliable predictions by quantifying uncertainty in the predictions. (Blundell *et al.*, 2015; Gal and Ghahramani, 2016; Kendall and Gal, 2017; Louizos and Welling, 2017)

In this study, we pick exchange rates among millions of time series for prediction and constructing prediction intervals since exchange rate prediction is a crucial area of research with significant implications for econometrics and financial economics. Traditional approaches to exchange rate prediction include time series analysis, regression models, and some structural models. In spite of such variability, these models often fail to capture the complex relationships and nonlinearities that exist in currency markets, leading to poor performance in predicting exchange rates. Recently, deep learning models, such as neural networks and convolutional neural networks, have shown promising results in predicting exchange rates (Fischer and Krauss, 2018; Galeshchuk and Mukherjee, 2017).

3. METHODOLOGY

In the realm of time series forecasting, particularly in econometric and financial applications, accurate predictions are essential. However, equally important is the ability to quantify the uncertainty associated with these predictions. Uncertainty in predictions arises from two primary sources: aleatoric uncertainty and epistemic uncertainty. Aleatoric uncertainty, also known as data uncertainty, is inherent in the variability and noise present in the data itself. This type of uncertainty is irreducible; no matter how sophisticated the model or how extensive the data, there will always be an element of randomness or unpredictability in the outcome. Prediction intervals are essential in addressing aleatoric uncertainty, as they provide a range within which future observations are expected to fall. On the other hand, epistemic uncertainty, also referred to as model uncertainty, is the uncertainty related to the model's knowledge of the data. This type of uncertainty is reducible and can be diminished with more data, better models, or enhanced training techniques.

In this study, we address both aleatoric and epistemic uncertainties in the context of predicting daily exchange rates and constructing prediction intervals. We employ a VAE, a generative model that captures the underlying structure of the data by learning a latent representation of the exchange rates. The VAE is particularly effective in managing epistemic uncertainty, as it enables the model to generalize across different scenarios, even in regions where traditional time series model might struggle to specify a clear data structure. The VAE's ability to generate multiple sets of latent factors from the learned distribution is central to our methodology. By sampling from these latent factors, we reconstruct thousands of exchange rate series, each representing different possible realizations of the underlying data. This approach inherently captures aleatoric uncertainty by reflecting the variability due to noise, which is especially critical for high-frequency data.

To further quantify uncertainty in our predictions, we construct prediction intervals by utilizing a MLP initially trained on observed exchange rates. To form these intervals, we input a series of reconstructed exchange rate data generated by the VAE into the trained MLP, yielding a distribution of predicted values. The mean of these predictions is used as the final point forecast, while the variability across predictions provides the basis for the prediction intervals, capturing aleatoric uncertainty in the forecasts. While this approach does not employ traditional bootstrapping, the use of multiple VAE-generated reconstructions serves a similar purpose by simulating a variety of plausible future series, allowing us to capture the inherent data variability. This strategy enables the construction of robust prediction intervals, offering a structured insight into the level of confidence associated with the model's forecasts.

3.1. METHOD OF PREDICTION AND PREDICTION INTERVALS BUILDING

In this section, we propose a methodological framework for predicting daily exchange rates and constructing prediction intervals by combining a VAE and a MLP. The VAE is used to extract latent factors from the daily exchange rates of ten major currencies—Euro (EUR), British Pound (GBP), Korean Won (KRW), Japanese Yen (JPY), Mexican Peso (MXN), Canadian Dollar (CAD), Australian Dollar (AUD), Chinese Yuan (CNY), Brazilian Real (BRL), and Indian Rupee (INR)—as well as the US Dollar Index (USD). These 11 series represent major global currency interactions and are sufficient to capture the primary structural and dynamic components that drive fluctuations in exchange rates. Including additional variables could introduce noise without significantly enhancing predictive performance, as these currencies already encompass diverse economic contexts and trading relationships central to global currency markets.

Once trained, the VAE generates a thousand distinct reconstructions of the KRW exchange rate series by sampling from the latent space. Each reconstruction represents a potential realization of the exchange rate, capturing variability within the series. These reconstructions serve as inputs to an MLP model, which is then trained to produce short-term predictions. By utilizing multiple VAE-generated scenarios, we construct prediction intervals that reflect uncertainty in future exchange rates, offering a robust framework to better understand and quantify the inherent variability within exchange rate predictions. To construct prediction intervals, we train an MLP model on the observed KRW series and then use it to predict exchange rates based on the VAE-generated reconstructions. Each reconstruction is passed through the trained MLP to produce an individual forecast, yielding a distribution of predicted values for each future time step. The final point prediction is taken as the mean of these predictions, while the prediction interval bounds are calculated based on the variability across the distribution, capturing the uncertainty inherent in the data. This combination of VAE and MLP provides a flexible method for creating prediction intervals, quantifying the aleatoric uncertainty within the exchange rate forecasts.

3.2. DATA PREPROCESSING

The data preprocessing begins with collecting daily historical exchange rate data for ten major currencies (EUR, GBP, KRW, JPY, MXN, CAD, AUD, CNY, BRL and INR) and the US Dollar Index (DXY) from the FRED database maintained by the Federal Reserve Bank of St. Louis. The dataset spans from January 2006. when the US Dollar Index was first published, to July 2024. To prepare the data for analysis, we compute the log differences of the exchange rates and the US Dollar Index, transforming them into day-on-day (DoD) series. This transformation captures the relative changes in the exchange rates on a daily basis. Finally, we apply min-max scaling to normalize the data, ensuring that it is standardized within a rage suitable for the VAE model.

$$x_i' = \frac{\Delta \log(x_i) - \min(\Delta \log(X))}{\max(\Delta \log(X)) - \min(\Delta \log(X))} \tag{1}$$

where $\Delta \log(x_i)$ represents the log difference of the original exchange rate value at time i, and x_i' represents the normalized value after applying log differencing and min-max scaling. In this process, $\Delta \log(x_i) = \log(x_i) - \log(x_{i-1})$ is the day-on-day log difference of the exchange rate, $\min(\Delta \log(X))$ is the minimum value within the log-differenced series, and $\max(\Delta \log(X))$ is the maximum value. This normalization is crucial to scale the data into a consistent range, typically between 0 and 1, which enhances the stability and effectiveness of the VAE during training.

	mean	std	min	25%	50%	75%	max	n
EUR	1.2321	0.1348	0.9616	1.1149	1.2125	1.3366	1.6010	4620
GBP	1.5002	0.2319	1.0703	1.2984	1.4929	1.6184	2.1104	4620
AUD	0.8069	0.1235	0.5755	0.7128	0.7662	0.9071	1.1026	4620
CNY	6.76	0.47	6.04	6.37	6.76	6.98	8.06	4620
JPY	108.76	17.77	75.72	98.61	109.09	117.22	161.73	4620
CAD	1.1908	0.1379	0.9168	1.0474	1.2378	1.3174	1.4592	4620
MXN	15.73	3.64	9.91	12.73	15.32	19.07	25.13	4620
BRL	3.1855	1.3441	1.5375	2.0130	3.0778	4.1766	5.9204	4620
INR	61.15	13.35	38.48	47.00	63.47	72.02	83.75	4620
KRW	1140.57	115.84	903.20	1080.82	1131.49	1192.61	1570.10	4620
DXY	104.77	11.81	85.46	93.42	106.15	115.11	128.45	4620

Table 1: DESCRIPTIVE STATISTICS EXCHANGE RATES AND US DOLLAR INDEX (JAN 2006 - JULY 2024). The table provides a comprehensive overview of the descriptive statistics for the exchange rates of ten major currencies and the US Dollar Index, covering the period from January 2006 to July 2024. The currencies included are the Euro (EUR), British Pound (GBP), Korean Won (KRW), Japanese Yen (JPY), Mexican Peso (MXN), Canadian Dollar (CAD), Australian Dollar (AUD), Chinese Yuan (CNY), Brazilian Real (BRL), and Indian Rupee (INR).

3.3. VAE MODEL CONSTRUCTION

The VAE model is designed to capture the underlying structure of exchange rates by mapping it into a distribution in a latent space and then reconstructing the original series from this latent representation. The distribution in the latent space is typically a Gaussian distribution and the encoder network outputs parameters (mean and log-variance) define this distribution. The VAE consists of three main components: an encoder, a decoder, and a loss function that balances reconstruction accuracy with regularization using Kullback-Leibler (KL) divergence.

- **Encoder:** The encoder is a neural network designed to take normalized and DoD-transformed exchange rate data as input. In this study, the input dimension is 11, corresponding to the ten currency pairs plus the US Dollar Index. The encoder outputs $2 \times d$ values, where d is the dimension of the latent variable z, set to 64 in this model. This dimension is chosen based on the network's structure, where the number of neurons is halved in each successive layer to effectively compress information in the latent space. Specifically, the layer before the latent layer contains 128 neurons, with the reduction to 64 balancing dimensionality reduction and model complexity. Of the $2 \times d$ output values, the first d values represent the mean (μ) of the latent variables, while the remaining d values correspond to the log-variance ($\log(\sigma^2)$), facilitating a probabilistic encoding of the data in the latent space. This structure supports robust feature extraction by capturing essential patterns in the exchange rate series while maintaining computational efficiency.

- **Decoder:** The decoder is a neural network that reconstructs the exchange rate series from the latent variables. The input dimension of the decoder is d, and its output dimension is equal to the number of currency pairs plus the US Dollar Index, effectively reversing the encoding process to generate a reconstruction of the original series.

3.4. TRAINING THE VAE

The VAE is trained to minimize a loss function that combines a reconstruction loss, which measures the accuracy of the reconstructed exchange rates, and a regularization term, which ensures that the latent space is well-behaved by enforcing a Gaussian distribution on the latent variables.

$$L_{\text{total}} = L_{\text{recon}} + \beta L_{\text{KL}}$$

where:

- L_{recon} is the reconstruction loss, typically measured using Mean Squared Error (MSE), which quantifies the difference between the original and reconstructed data.

- L_{KL} is the Kullback-Leibler divergence, which penalizes the deviation of the latent variables' distribution from a standard Gaussian distribution.

- β is a hyperparameter that controls the trade-off between the reconstruction accuracy and the regularization of the latent space. We set $\beta = 0.5$, however, setting β to a different value, such as 1, would not result in a meaningful difference.

3.5. EXTRACTING LATENT FACTORS AND RECONSTRUCTION

Once the VAE model is trained, it is used to extract latent factors z from the normalized exchange rate data. The normalization process involves first taking the log difference of the original series to stabilize variance and then applying min-max scaling to ensure that the data lies within a range suitable for model training. The latent factors z capture the underlying dynamics of the exchange rate data. To generate multiple scenarios for prediction, we sample from the Gaussian distributions defined by the encoder's outputs (mean μ and log-variance $\log(\sigma^2)$) and use these samples to reconstruct a thousand different series of exchange rates through the decoder. Figure 2 shows mean of normalized and DoD transformed latent factors of Korean Won compared to DoD transformed original series. We specify the process of VAE train and extraction of the latent factors below in the pseudo code.

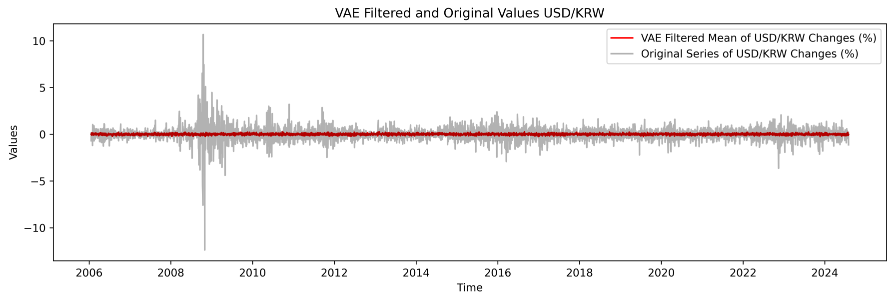

Figure 1: LOG-DIFFERENCED (DAY-ON-DAY) LATENT FACTORS AND ORIGINAL KRW EXCHANGE RATES. This figure compares DoD transformed latent factors and the original series of Korean won exchange rates agaisnt US dollar.

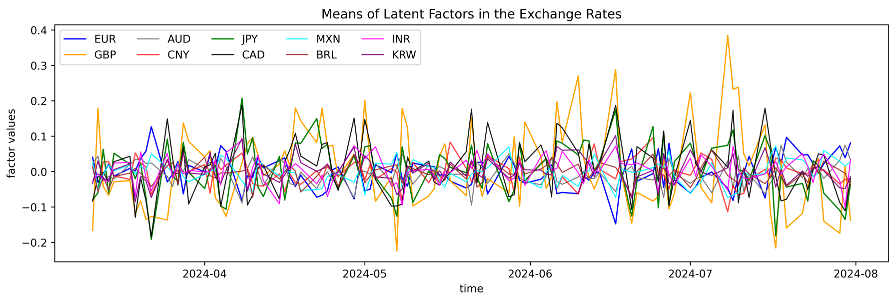

Figure 2: LOG-DIFFERENCED (DAY-ON-DAY) LATENT FACTORS OF EXCHANGE RATES OF TEN CURRENCIES. This figure exhibits DoD transformed latent factors of exchange rates of ten currencies against US dollar in recent periods. The currencies included are the Euro (EUR), British Pound (GBP), Korean Won (KRW), Japanese Yen (JPY), Mexican Peso (MXN), Canadian Dollar (CAD), Australian Dollar (AUD), Chinese Yuan (CNY), Brazilian Real (BRL), and Indian Rupee (INR).

Pseudo Code for training VAE and Extrating Latent Factors

```
Initialize parameters for the VAE model:
- input_dim: dimension of the input data (number of currencies +
  USD Index)
- latent_dim: dimension of the latent space
- intermediate_dims: dimensions of the intermediate hidden layers

Define the Encoder:
- Input: Exchange rate data (X) of ten currencies
- Layer 1: Dense layer with 512 neurons and ReLU activation
          function
- Layer 2: Dense layer with 256 neurons and ReLU activation
          function
- Layer 3: Dense layer with 128 neurons and ReLU activation
          function
- Output 1: Latent mean in latent dimension
- Output 2: Latent log-variance in latent dimension

Define the Sampling Function:
- Input: mean and log-variance
- Sample from standard normal distribution:
  epsilon = random_normal(shape=latent dimension)
- Output: Sampled latent vector (z) = mean + exp(0.5* log-
  variance) * epsilon

Define the Decoder:
- Input: Latent vector (z)
- Layer 1: Dense layer with 128 neurons and ReLU activation
          function
- Layer 2: Dense layer with 256 neurons and ReLU activation
          function
- Layer 3: Dense layer with 512 neurons and ReLU activation
          function
- Output: Reconstructed exchange rate data (X')

Define the VAE Model:
- Input: Exchange rate data (X)
- Encode: Pass X through the encoder to get mean and log-variance
- Sample: Pass mean and log-vararariance through the sampling
          function to get z
- Decode: Pass z through the decoder to get X'
- Output: Reconstructed exchange rate data (X')

Define the VAE Loss Function:
- Reconstruction Loss: Compute the Mean Squared Error (MSE)
  between X and X'
- KL Divergence Loss: Compute the Kullback-Leibler divergence
```

```
  between the learned distribution (mean, log-variance) and the
  standard normal distribution
- Total Loss: total loss = Reconstruction Loss + beta
              * KL Divergence Loss

Train the VAE Model:
- Initialize optimizer (e.g., Adam)
- For each epoch:
- Forward pass: Pass the input data X through the VAE model
- Compute loss: Calculate the total loss
- Backpropagation: Compute gradients and update the model
  parameters using the optimizer

After training:
- Use the trained encoder to extract latent factors (z) from the
  normalized exchange rate data
- Generate multiple sets of latent factors by sampling from the
  learned latent distribution
- Use the decoder to reconstruct exchange rate series from the
  sampled latent factors

Output:
- Latent factors (z)
- Reconstructed exchange rate series
```

3.6. MLP MODEL TRAINING AND PREDICTION

The observed KRW exchange rates are used to train the MLP model, while the reconstructed series generated by the VAE serve as inputs during prediction. This approach allows the MLP to generate future values of the KRW exchange rate based on variations captured in the VAE's latent representations, which reflect the inherent dynamics of the original data.

Data Splitting: The dataset consists of KRW exchange rates, transformed into day-on-day differences and scaled for model compatibility. The data is divided into training, validation, and test sets based on the following proportions:

- Training Set: 80% of the dataset, covering January 19, 2006, to October 16, 2020
- Validation Set: 10% of the dataset, covering October 19, 2020, to August 22, 2022
- Test Set: 10% of the dataset, covering August 23, 2022, to July 31, 2024

Let $\mathbf{X}_{\text{train}}, \mathbf{X}_{\text{val}}, \mathbf{X}_{\text{test}}$ represent the input matrices for the training, validation, and test sets, respectively, and let $\mathbf{y}_{\text{train}}, \mathbf{y}_{\text{val}}, \mathbf{y}_{\text{test}}$ denote the corresponding target vectors. This proportional division ensures a robust training process, with validation and test sets that support both model tuning and out-of-sample performance evaluation, as shown in Figure 3.

MLP Architecture: The MLP model is structured as a feed forward neural network with multiple hidden layers. The model architecture can be mathematically represented as follows:

$$\mathbf{h}^{(1)} = \tanh(\mathbf{W}^{(1)}\mathbf{X} + \mathbf{b}^{(1)}),$$

$$\begin{aligned} \mathbf{h}^{(2)} &= \tanh(\mathbf{W}^{(2)}\mathbf{h}^{(1)} + \mathbf{b}^{(2)}), \\ \mathbf{h}^{(3)} &= \tanh(\mathbf{W}^{(3)}\mathbf{h}^{(2)} + \mathbf{b}^{(3)}), \\ \hat{y} &= \mathbf{W}^{(4)}\mathbf{h}^{(3)} + \mathbf{b}^{(4)}, \end{aligned}$$

where:

- $\mathbf{X} \in \mathbb{R}^{n_{\text{lags}} \times n_{\text{vars}}}$ is the input matrix, where n_{lags} is the number of lagged observations and n_{vars} is the number of variables (in this case, exchange rates).
- $\mathbf{W}^{(1)}, \mathbf{W}^{(2)}, \mathbf{W}^{(3)}, \mathbf{W}^{(4)}$ are weight matrices for each layer.
- $\mathbf{b}^{(1)}, \mathbf{b}^{(2)}, \mathbf{b}^{(3)}, \mathbf{b}^{(4)}$ are bias vectors for each layer.
- $\mathbf{h}^{(1)}, \mathbf{h}^{(2)}, \mathbf{h}^{(3)}$ are the activations of the hidden layers, each using the tanh activation function.
- $\hat{y}$ is the predicted output of the model, representing the future value of KRW exchange rates

Training Process: The MLP model is trained by minimizing the following loss function:

$$L(\mathbf{W}, \mathbf{b}) = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2,$$

where N is the number of training samples, y_i is the actual target value, and $\hat{y}_i$ is the predicted target value. The training process employs backpropagation, a method used to compute the gradients of the loss function with respect to the model parameters. These gradients are then used by the Adam optimizer to adjust the weights and biases of the network in order to minimize the loss. Backpropagation works by propagating the error from the output layer back through the network, layer by layer, allowing the model to learn the appropriate adjustments to make for improving its predictions. The MSE serves as the primary loss function, while the MAE is used as an evaluation metric. Additionally, the training process incorporates early stopping and learning rate scheduling to prevent overfitting and ensure optimal model performance.

The pseudo code of specific structure of the MLP model and how to train and validate the model is as follows:

Pseudo Code for MLP Training

```
1. Initialize Parameters:
- Set input dimension to (number of input variables)
- Define batch_size (e.g., 256)
- Define the number of epochs (e.g., 50)
- Initialize the learning rate (e.g., 0.001)

2. Define the MLP Model Architecture:
- Create a sequential model
- Add a dense layer with 512 neurons with hyperbolic tangent
  activation function and set input dimension
- Add a dense layer with 128 neurons and hyperbolic tangent
  activation function
- Add a dense layer with 32 neurons and hyperbolic tangent
  activation function
- Add an output dense layer with 1 neuron and a linear
  activation function

3. Compile the Model:
- Use the Adam optimizer with the defined learning rate
- Set the loss function to Mean Squared Error (MSE)
- Set the evaluation metric to Mean Absolute Error (MAE)

4. Implement Callbacks:
- Implement early stopping to monitor validation loss with a
  patience of 10 epochs, and restore the best weights
- Implement a learning rate scheduler to reduce the learning
  rate by a factor of 0.1 if validation loss does not improve
  after 5 epochs, with a minimum learning rate of 1e-6

5. Train the Model:
- Train the MLP model on the training data
- Use the validation data to monitor performance
- Train for the specified number of epochs, with early stopping
  and learning rate scheduling enabled

6. Evaluate the Model:
```

```
- Evaluate the trained model on the test data
- Output the test loss (MSE) and test mean absolute error (MAE)
```

3.7. CONSTRUCTING PREDICTION INTERVALS

To quantify the uncertainty in the predictions, we construct prediction intervals based on the distribution of predictions obtained from the MLP model. The process involves generating multiple reconstructions of the exchange rate series using the VAE and training the MLP on these reconstructions to derive a distribution of predicted values.

3.7.1. Generating Multiple Predictions

Let $\hat{y}_t^{(k)}$ denote the prediction at time t obtained from the k-th MLP model trained on the k-th reconstructed series. We generate K different reconstructions using the VAE, leading to K predictions for each time point t. This can be expressed as:

$$\hat{y}_t^{(k)} = \mathrm{MLP}_k(\mathbf{z}^{(k)}),$$

where $\mathbf{z}^{(k)}$ represents the latent factors from the k-th VAE reconstruction.

3.7.2. Calculating Mean and Standard Deviation

For each time point t, the mean prediction μ_t and the standard deviation σ_t of the predictions are computed as:

$$\mu_t = \frac{1}{K}\sum_{k=1}^{K} \hat{y}_t^{(k)},$$

$$\sigma_t = \sqrt{\frac{1}{K-1}\sum_{k=1}^{K}\left(\hat{y}_t^{(k)} - \mu_t\right)^2}.$$

3.7.3. Constructing Prediction Intervals

The prediction intervals are constructed based on the mean and standard deviation. For a confidence level α, the prediction interval PI_t at time t can be calculated as:

$$PI_t = \left[\mu_t - z_{\alpha/2}\cdot\sigma_t, \mu_t + z_{\alpha/2}\cdot\sigma_t\right],$$

where $z_{\alpha/2}$ is the critical value from the standard normal distribution corresponding to the desired confidence level.

3.7.4. Pseudocode for Prediction Interval Construction

```
Input: Reconstructed exchange rate series from VAE, MLP model,
       confidence level
Output: Prediction intervals for each time point t

1. Initialize K ← Number of reconstructed series
2. Initialize predictions ← Empty list to store MLP predictions

3. For each k in {1, 2, ..., K} do:
a. Train MLP on the k-th reconstructed series
b. Predict future exchange rates using the trained MLP
c. Store predictions in predictions[k]

4. For each time point t do:
a. Compute mean (predictions at time t)
b. Compute standard deviation (predictions at time t)
c. Compute prediction intervals

5. Return prediction intervals
```

3.8. RESULTS AND DISCUSSION

The proposed methodology effectively addresses the challenge of prediction uncertainty in time series data through deep learning model such as MLP. By leveraging the strengths of generative models, specifically the VAE for latent factor extraction and the MLP for predicting exchange rates, we generate accurate predictions accompanied by well-calibrated uncertainty measures. The prediction intervals are constructed in two different ranges: 90% and 95%. 90% (95%) prediction intervals are built by adding and subtracting by 1.65 (1.96) times of a standard deviation of a thousand scenarios of predictions in each point of prediction. The prediction intervals constructed using this approach provide a comprehensive understanding of the potential range of future exchange rates as shown in Figure 3. The VAE's ability to generate multiple sets of latent factors allows us to capture both the underlying structure of the data and the inherent noise (aleatoric uncertainty). By training the MLP in these diverse reconstructions, we account for epistemic uncertainty, reflecting the model's confidence in its predictions. The resulting prediction intervals are not only indicative of the expected values but also provide a probabilistic measure of the uncertainty in its prediction, making it more robust and reliable.

Model	MSE	MAE
VAE + MLP	0.004513	0.051676
ARIMA	0.007187	0.073190
Factor Model	0.008516	0.080993

Table 2: PREDICTION PERFORMANCE COMPARISON. The table provides performance measures in MSE (Mean Squared Error) and MAE (Mean Absolute Error) of time series models. The Dynamic Factor Model (DFM) and ARIMA model are used as benchmarks to compare against the proposed VAE-MLP method for exchange rate forecasting. The DFM captures common latent factors driving the dynamics of multiple time series. Specifically, one latent factor was specified ($K = 1$) with a first-order autoregressive process ($p = 1$), representing the shared structure across 11 series, including the Korean Won (KRW) exchange rate and the US Dollar Index. The ARIMA model was fitted to the KRW exchange rate series individually, using an order of $(1,1,1)$, which combines an autoregressive term, a single differencing operation, and a moving average term. Both models were estimated using maximum likelihood methods, and their forecasts were compared against those of the VAE-MLP approach in terms of prediction accuracy and uncertainty quantification.

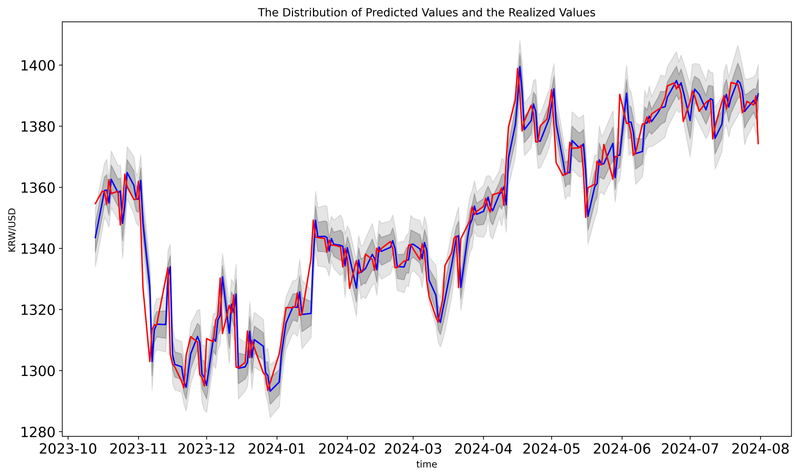

Figure 3: PREDICTION INTERVALS AND ORIGINAL SERIES OF USD/KRW. This figure illustrates the predicted USD/KRW exchange rates (blue line) along with the original series (red line). The shaded areas represent the prediction intervals: the dark grey area corresponds to the 95% prediction intervals, while the light grey area indicates the 90% prediction intervals. These intervals provide a probabilistic measure of the uncertainty surrounding the predictions with wider intervals indicating greater uncertainty.

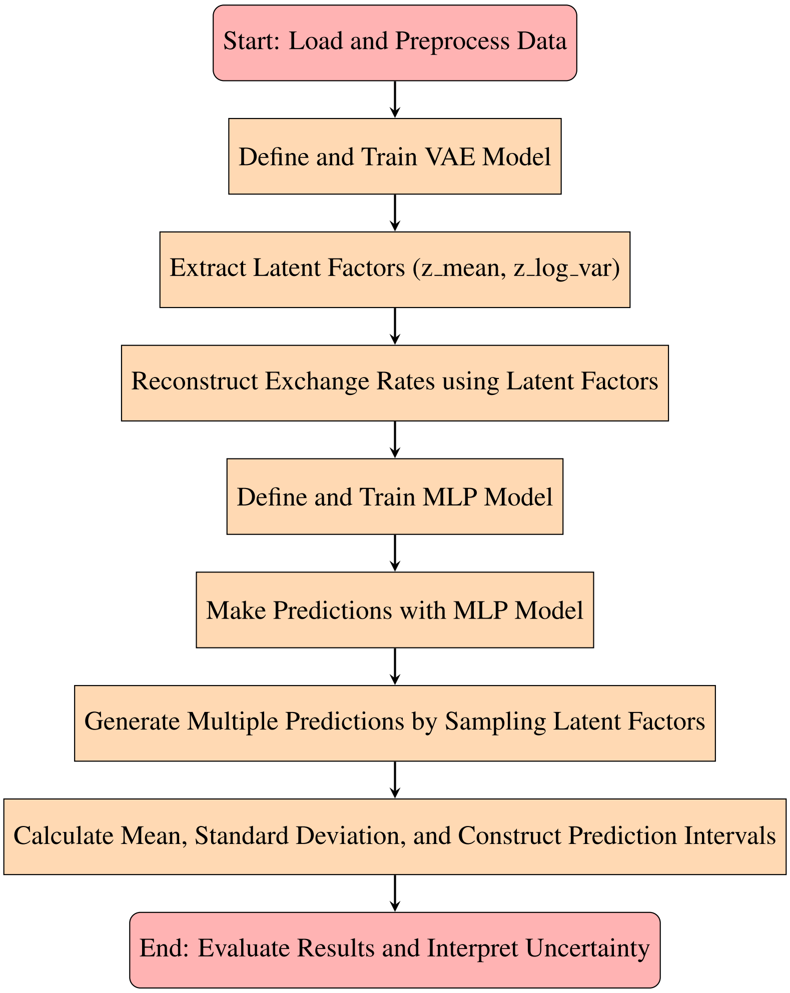

Figure 4: FLOWCHART OF THE PROCESS FROM PREPROCESSING TO PREDICTION INTERVAL CONSTRUCTION. This flowchart outlines the steps in constructing prediction intervals for exchange rates using a VAE and a MLP. The process begins with loading and preprocessing the data, followed by defining and training the VAE to extract latent factors. These factors are used to reconstruct the exchange rates, which are then used as inputs for the MLP model to make predictions. Multiple predictions are generated by sampling different latent factors, and the mean and standard deviation of these predictions are calculated to construct prediction intervals, providing a range of possible future exchange rates.

4. CONCLUSION

In this study, we explored the use of generative models to predict daily frequency data and construct prediction intervals, addressing the critical challenging of quantifying uncertainty in time series forecasting. Specifically, we employed a VAE to extract latent factors from the exchange rates of ten currencies and the US Dollar Index. By leveraging the VAE's generative capabilities, we were able to reconstruct multiple sets of exchange rates series, which were then used to train a MLP model. This approach allowed us to generate a distribution of predicted values, from which we constructed prediction intervals, providing a robust measure of uncertainty in the forecast exchange rates.

The results of out methodology demonstrate its effectiveness in addressing both aleatoric and epistemic uncertainties. The VAE, as a relative model to generative artificial intelligence (AI), capture the inherent noise and variability in the data, allowing the MLP model to generalize across different scenarios, even in the presence of high-frequency fluctuations. The prediction intervals constructed from the multiple reconstructions offered valuable insights into the range of possible future exchange rates. These intervals are often as crucial as the predictions themselves in policy decision and financial market decision-making. We used exchange rates but we can apply our methodology to any time series that we can study.

However, this study is under its certain limitations. One of the primary challenges encountered was the complexity of training VAE and MLP models, particularly in handling high-dimensional financial data. The process of tuning hyperparameters, such as β in the VAE's loss function, and setting the architecture of the VAE and MLP was computationally intensive and required extensive experimentation. Moreover, the VAE's reliance in the assumption that the latent variables follow a Gaussian distribution may not always be appropriate for all types of financial data, potentially limiting the model's ability to capture more complex, non-Gaussian relationships in the data. Another limitation is related to the scope of the data used. While the study focused on ten major currencies and the UD Dollar Index, the model's performance and generalizability to other currencies or financial instruments were not explored. Additionally, the model's ability to adapt to structural change in the market or to perform in periods of extreme volatility was not extensively tested. Future research can address these limitations in several ways. First, exploring alternative generative models, such as Generative Adversarial Networks (GANs), could provide a more flexible framework for modeling the latent structure of financial data, potentially capturing more complex patterns than the VAE. Furthermore, incorporating additional macroeconomic indicators or market variables into the model could enhance its predictive power and robustness. Another promising direction for future research is the application of this methodology to a broader range of financial instruments and markets. Extending the model to forecast other asset classes, such as commodities, equities, or interest rates, could provide further insights into the generalizability and scalability of the approach. In addition, future research could investigate strategies to align these prediction intervals more dynamically with periods of extreme volatility. One potential direction could be to develop hybrid models that incorporate real-time volatility measures or dynamic adjustment mechanisms into the VAE-MLP framework, enhancing its responsiveness to sudden market shifts.

REFERENCES

Abraham, B. and J. Ledolter (2009). *Statistical Methods for Forecasting,* John Wiley & Sons.

Blundell, C., Cornebise, J., Kavukcuoglu, K., and D. Wierstra (2015). “Weight uncertainty in neural networks,” *Proceedings of the 32nd International Conference on Machine Learning* 37, 1613–1622.

Brockwell, P. J. and R. A. Davis (2002). *Introduction to time series and forecasting,* Springer.

Chen, S. and C. Hsu (2019). “Do stock markets have predictive content for exchange rate movements?,” *Journal of Forecasting* 38, 699–713.

Clements, M., Galvão, A., and J. Kim (2008). “Quantile forecasts of daily exchange rate returns from forecasts of realized volatility,” *Journal of Empirical Finance* 15, 729–750.

Davison, A. C. and D. V. Hinkley (1999). *Bootstrap methods and their application,* Cambridge University Press.

Enders, W. (2014). *Applied econometric time series,* 4th Edition, Wiley.

Fischer, T. and Krauss, C. (2018). “Deep learning with long short-term memory networks for financial market predictions,” *European Journal of Operational Research* 270, 654–669.

Flunkert, V., Salinas, D., and J. Gasthaus (2020). “DeepAR: Probabilistic forecasting with autoregressive recurrent networks,” *International Journal of Forecasting* 36, 1181–1191.

Frean, M., Lilley, M., and P. Boyle (2006). “Implementing Gaussian process inference with neural networks,” *International Journal of Neural Systems* 16, 321–327.

Gal, Y. and Z. Ghahramani (2016). “Dropout as a Bayesian approximation: Representing model uncertainty in deep learning,” in *Proceedings of the 33rd International Conference on Machine Learning* 48, 1050–1059.

Galeshchuk, S. and S. Mukherjee (2017). “Deep networks for predicting direction of change in foreign exchange rates,” *Intelligent Systems in Accounting, Finance, and Management* 24, 100–110.

Hamilton, J. D. (1994). *Time Series Analysis,* Princeton University Press.

Hwang, E. and D. W. Shin (2010). “Stationary bootstrapping for non-parametric estimator of nonlinear autoregressive model,” *Journal of Time Series Analysis* 32, 292–303.

Karmakar, S., Chudy, M., and W. B. Wu (2021). “Long-term prediction intervals with many covariates,” *Journal of Time Series Analysis* 43, 587–609.

Kendall, A. and Y. Gal (2017). “What uncertainties do we need in Bayesian deep learning for computer vision?,” *Advances in Neural Information Processing Systems* 30, 5574–5584.

Kysely, J. (2008). “A cautionary note on the use of nonparametric bootstrap for estimating uncertainties in extreme-value models,” *Journal of Applied Meteorology and Climatology* 47, 3236–3251.

Li, J. (2020). “Block bootstrap prediction intervals for parsimonious first-order vector autoregression,” *Journal of Forecasting* 40, 512–527.

Louizos, C. and M. Welling (2017). “Structured and efficient variational deep learning with matrix Gaussian posteriors,” in *Proceedings of the 34th International Conference on Machine Learning* 70, 1708–1716.

Novoa, C. and F. A. Mendez (2009). “Bootstrap methods for analyzing time studies and input data for simulations,” *International Journal of Productivity and Performance Management* 58, 460–479.

Franco, P. G. A. and R. C. Souza (2001). "A comparison of methods for bootstrapping in the local level model," *Journal of Forecasting* 21, 27–38.

Tambakis, D. and A. Royen (2002). "Conditional predictability of daily exchange rates," *Journal of Forecasting* 21, 301–315.

Wang, J. and J. Wu (2009). "The Taylor rule and interval forecast for exchange rates," Board of Governors of the Federal Reserve System, Technical Report 963r.

Zhang, J., Ye, H., Song, T., Wang, H., Xue, Z., Ma, R., and H. Guan (2022). "Improving Bayesian neural networks by adversarial sampling," *Journal of Artificial Intelligence Research* 36, 10110–10117.