Working Paper - Latent Decomposition for Irregular Systems: A VAE Framework for Arrhythmias and Commodity Markets

Anirudh Krovi[1]

[1]PhD, Northwestern University; MBA, NYU Stern; Formerly at McKinsey & Company,
anirudh.krovi@stern.nyu.edu

July 11, 2025

Abstract

We introduce a variational autoencoder (VAE) architecture that disentangles time series dynamics into two latent components: a rhythm vector (z_{rhythm}) capturing consistent temporal structure, and an offset vector (z_{offset}) representing transient deviations or shocks. This decomposition enables interpretable modeling of complex, noisy sequences. We validate the approach on two structurally analogous domains: arrhythmia classification in ECG signals (MIT-BIH dataset) and trading signal generation in commodity markets (crude oil and wheat). Across both, our method yields informative latent spaces and supports robust downstream prediction, suggesting a flexible foundation for cross-domain time series understanding.

1 Introduction

Variational Autoencoders (VAEs) have emerged as powerful generative models for unsupervised representation learning, particularly in time series and structured data domains [10, 16]. By learning a low-dimensional latent space that captures the generative factors of the data, VAEs have found broad applicability in anomaly detection, signal reconstruction, and sequence modeling.

Much of the recent progress in the VAE literature has focused on refining decoder architectures to better absorb domain-specific geometric content. Techniques such as attention-based decoders, spatially-aware networks, and physically-informed reconstructions have shown strong performance in fields ranging from neuroscience to climate modeling [2, 8, 11]. However, these approaches often require extensive tuning, increased computational resources, and domain-specific customization, limiting their scalability and accessibility.

A complementary line of work has started to investigate the geometry of the latent space itself—modifying its structure, topology, or dynamics to impose interpretability and disentanglement [5, 12, 18]. These methods often bring stronger inductive biases and can provide insight into underlying factors without relying on opaque decoder behavior.

In this paper, we contribute to this growing body of work by proposing a latent-space-oriented VAE architecture that learns to decompose irregular time series into two meaningful components: (1) a **rhythmic latent** z_{rhythm} that captures periodic, base structure, and (2) an **offset latent** z_{offset} that isolates deviations, shocks, or anomalies. While our model is not derived from a stochastic calculus framework, it is conceptually inspired by stochastic derivatives where systems are understood as a combination of steady base behavior and irregular driving forces.

This rhythm-offset decomposition idea is common in natural and economic systems. For instance, in cardiology, an arrhythmia may represent a deviation from a regular heartbeat—a localized offset within a broader rhythm. Similarly, in commodities markets, certain assets like crude oil are prone to high-frequency shocks and geopolitical noise, while others like wheat may display seasonal patterns with intermittent anomalies. Prior work in commodity modeling has approached these structures using time-series econometrics, seasonal adjustments, and volatility clustering [4, 15, 7], but the role of latent-space structure remains underexplored.

Our central question is this: *Can we use latent decomposition via carefully designed loss functions to learn rhythm-offset representations that are transferable across domains and directly useful for downstream tasks such as classification and trading?* To answer this, we apply our method to two challenging settings:

- The **MIT-BIH Arrhythmia Dataset**, a canonical benchmark for anomaly detection in biological time series, where class imbalance and subtle variation complicate modeling.
- **Commodity price series** for crude oil and wheat, where the learned offset signal is used to generate interpretable and profitable trading signals.

We place particular emphasis on the latent space because it provides a modular and computationally efficient way to impose domain knowledge. This allows us to design reusable priors and modify tradeoffs (e.g., rhythm vs. offset) without architecture-heavy intervention.

Paper layout. Section 2 reviews relevant literature on VAEs, arrhythmia modeling, and financial time series forecasting. Section 3 introduces our dual-latent architecture and the associated training objective. Section 4 presents results from the MIT-BIH arrhythmia dataset, applications to commodity markets, focusing on crude oil and wheat. We conclude in Section 5 with a discussion of findings, limitations, and promising directions for future research, including topological regularization and latent forecasting.

2 Related Work

2.1 Variational Autoencoders and Latent Disentanglement

Variational Autoencoders (VAEs) provide a powerful framework for learning latent representations of data via amortized variational inference [10, 16]. A major line of work focuses on encouraging disentanglement—learning latent codes that capture independent factors of variation. Notable efforts include β-VAE [8], FactorVAE, and others [2, 11], which modify the VAE objective to encourage axis-aligned or minimally correlated latent spaces.

Complementary to this are approaches that investigate *latent structure*. For example, InfoVAE [18] proposes a mutual-information-weighted objective to balance reconstruction and inference. Others consider manifold-constrained latent spaces, e.g., spherical or toroidal embeddings [5, 12]. These methods provide control over the latent geometry and enable modular inductive biases.

Our work sits at the intersection of disentanglement and structure: we introduce two interpretable latent variables—one for rhythmic base behavior and another for offsets—using domain-informed KL and regularization penalties to encourage functional decomposition.

2.2 Arrhythmia Detection and Medical Time Series

Physiological time series like ECGs are often modeled using sequence-aware architectures (e.g., CNNs, RNNs), with the MIT-BIH Arrhythmia Dataset being a popular benchmark. While classifi-

cation accuracy has improved through deep learning, issues of class imbalance and interpretability remain challenging [13].

Unsupervised models, including autoencoders and generative models, have been used to detect anomalies by learning typical heart rhythms and identifying deviations. However, these models typically do not enforce structural priors in latent space. Our method improves on this by encouraging a *rhythm-offset* separation directly in the latent design, improving interpretability and outlier localization.

2.3 Commodity Markets and Shock Modeling

Commodity markets exhibit both long-term seasonal patterns and sudden, often unpredictable, shocks. Seminal works by Deaton and Laroque [4] and Pindyck [15] modeled commodity dynamics using regime-switching models, inventory-driven processes, and stochastic volatility. More recent work emphasizes volatility transmission, forecastability, and risk modeling across commodities [7].

While these approaches offer interpretability, they often rely on predefined structural assumptions or manually engineered features. Our approach offers a learned, flexible latent decomposition: shocks are not assumed but inferred as a latent offset, and backtesting shows they can support actionable signals in markets such as crude oil and wheat.

2.4 Topology and Flows in Latent Space

A growing body of work explores the *topology* and *dynamics* of latent spaces in generative models. TopoVAE [1] introduces topology-preserving priors to maintain continuity across latent manifolds, while others use persistent homology to regularize or interpret learned spaces [6, 14]. Such approaches are particularly powerful in settings where loops, voids, or nontrivial geometry represent meaningful dynamics.

Flow-based models like RealNVP, Glow, and latent-space diffusion models have further expanded the toolkit for capturing non-Gaussian structure and inducing latent-time trajectories [9, 3]. Latent ODEs [17] model continuous dynamics within the latent space, enabling temporally smooth generation.

While our current work does not explicitly enforce topological constraints or latent flows, it is conceptually aligned with this literature in that we seek to impose interpretable *structure*—rhythm and offset—into a latent decomposition. In future work, we aim to extend our system with topological regularization and flow-based priors to capture temporal and geometric consistency across domains.

3 Methodology

We apply the same latent decomposition framework across two domains: electrocardiogram (ECG) time series from clinical data, and commodity price time series from financial markets. While the core model architecture remains unchanged, each application involves its own data modality, signal characteristics, and downstream task. For clarity, we present the methodology in two parallel sections—one for arrhythmias and one for commodities—each covering dataset construction, model design, loss formulation, training, and (in the case of commodities) backtesting. We begin with the medical setting.

3.1 Arrhythmias

3.1.1 Datasets and Preprocessing

We use the MIT-BIH Arrhythmia Database, a widely studied benchmark for heartbeat classification and anomaly detection. It contains electrocardiogram (ECG) recordings from 47 individuals, each lasting approximately 30 minutes and annotated beat-by-beat by clinical experts. Each recording includes two leads; we use only the first channel for consistency.

To construct our dataset, we extract fixed-length windows centered on each annotated heartbeat. Each window spans 256 samples and is included only if it lies entirely within the bounds of the recording. This results in a collection of heartbeat-centric time series segments, each associated with a beat-level annotation.

Beat annotations are mapped into a set of clinically meaningful categories, such as *Normal*, *Left Bundle Branch Block (LBBB)*, *Right Bundle Branch Block (RBBB)*, *Supraventricular Ectopic Beat (SVEB)*, *Ventricular Ectopic Beat (VEB)*, and others. Rare or ambiguous annotations are grouped under a general *Unknown* category. This mapping simplifies the classification space while preserving the distinction between normal and pathological rhythms.

To compress the temporal information and provide multiscale representations suitable for learning, we apply a discrete wavelet transform (DWT) to each window using a fourth-order Daubechies wavelet. The resulting wavelet coefficients are concatenated to form compact feature vectors for each heartbeat.

The final dataset includes raw ECG windows, their wavelet representations, and corresponding beat-level class labels. This preprocessing setup provides a balance between signal fidelity and computational efficiency, enabling robust downstream learning while preserving domain-relevant structure.

3.1.2 Model

We use a variational autoencoder (VAE) architecture designed to decompose each heartbeat into two interpretable latent components: one capturing the underlying rhythmic structure and another representing deviations or shocks. This decomposition aligns with the intuition that physiological time series often exhibit a base pattern with localized disruptions—such as those seen in arrhythmic conditions.

Encoder. The input to the model is a 256-dimensional vector representing a wavelet-transformed ECG window. A shared encoder maps this input into a 64-dimensional hidden representation through two fully connected layers with ReLU activations. This shared representation is then passed through two separate linear heads to produce the parameters of two distinct Gaussian distributions:

- $\mu_r, \log \sigma_r^2$: mean and log-variance of the *rhythmic latent* $z_{\text{rhythm}} \in \mathbb{R}^4$
- $\mu_o, \log \sigma_o^2$: mean and log-variance of the *offset latent* $z_{\text{offset}} \in \mathbb{R}^4$

Latent Sampling. Each latent vector is sampled using the standard reparameterization trick:

$$z = \mu + \epsilon \cdot \sigma, \quad \epsilon \sim \mathcal{N}(0, I)$$

where this is applied independently to both latent components.

Decoder. The model includes two separate decoders—one for each latent variable. Each decoder is a two-layer fully connected network that maps its respective latent vector back to a 256-dimensional reconstruction. The final output of the model is the sum of the two reconstructions:

$$\hat{x} = \hat{x}_{\text{rhythm}} + \hat{x}_{\text{offset}}$$

This additive structure enforces a clean compositional interpretation: the full signal is modeled as a combination of a base pattern and an interpretable deviation.

Forward Pass. The model outputs the reconstructed signal along with the latent parameters and sampled latent variables:

$$\texttt{forward}(x) \to (\hat{x}, \mu_r, \log \sigma_r^2, \mu_o, \log \sigma_o^2, z_r, z_o)$$

This setup allows the loss function to act independently on each latent channel while still optimizing reconstruction quality.

3.1.3 Loss Functions

The loss function is designed to guide the model toward a meaningful decomposition of each heartbeat into a base rhythmic component and an offset capturing localized irregularities. It consists of three terms: a reconstruction loss, KL divergence penalties for both latent spaces, and a regularization term that penalizes excessive reliance on the offset latent.

Notation. Let $x \in \mathbb{R}^n$ denote the input ECG segment of length $n = 256$, and let $\hat{x} \in \mathbb{R}^n$ be its reconstruction. The encoder produces parameters for two independent Gaussian latent distributions:

- $\mu_r, \log \sigma_r^2 \in \mathbb{R}^d$: mean and log-variance of the rhythm latent $z_r \in \mathbb{R}^d$
- $\mu_o, \log \sigma_o^2 \in \mathbb{R}^d$: mean and log-variance of the offset latent $z_o \in \mathbb{R}^d$

where $d = 4$ is the dimensionality of each latent space. Latents are sampled using the standard reparameterization trick.

Reconstruction Loss. We compute mean squared error (MSE) between the original signal and its reconstruction:

$$\mathcal{L}_{\text{recon}} = \frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{x}_i)^2$$

This term ensures fidelity to the observed ECG signal, regardless of how much is captured by each latent stream.

KL Divergence Terms. Each latent distribution is regularized via its KL divergence from the standard normal prior:

$$\mathcal{L}_{\text{KL},r} = D_{\text{KL}} \left(\mathcal{N}(\mu_r, \sigma_r^2 I) \,\|\, \mathcal{N}(0, I) \right)$$
$$\mathcal{L}_{\text{KL},o} = D_{\text{KL}} \left(\mathcal{N}(\mu_o, \sigma_o^2 I) \,\|\, \mathcal{N}(0, I) \right)$$

These terms encourage the latent spaces to remain smooth and compact. In practice, they are approximated as:

$$\mathcal{L}_{\text{KL}} = -\frac{1}{2} \sum_{j=1}^{d} \left(1 + \log \sigma_j^2 - \mu_j^2 - \sigma_j^2 \right)$$

and applied separately to both the rhythm and offset latents. A shared weight β_{KL} controls the strength of this regularization.

Offset Usage Penalty. To avoid the model over-relying on the offset latent z_o, we include an additional penalty on its magnitude:

$$\mathcal{L}_{\text{offset}} = \frac{1}{d} \sum_{j=1}^{d} z_{o,j}^2 = \|z_o\|^2$$

This encourages the model to use the offset channel only when necessary—i.e., when the rhythmic component alone cannot explain the signal. This sparsity-like constraint is crucial to making the latent decomposition interpretable and medically relevant.

Total Loss. The overall loss is a weighted sum of the three components:

$$\mathcal{L} = \mathcal{L}_{\text{recon}} + \beta_{\text{KL}} \cdot (\mathcal{L}_{\text{KL},r} + \mathcal{L}_{\text{KL},o}) + \lambda_{\text{offset}} \cdot \mathcal{L}_{\text{offset}}$$

where:

- β_{KL} controls the strength of latent regularization (typically set to 0.1)
- λ_{offset} governs how much offset usage is penalized (typically set to 0.01)

These weights are chosen to ensure that both latent channels remain active but not redundant. The model is thereby encouraged to place consistent, reconstructive information in z_r, while using z_o only to explain deviations or anomalies.

3.1.4 Training

The model is trained using mini-batch stochastic gradient descent with the Adam optimizer. Training proceeds for a fixed number of epochs, during which all loss terms—reconstruction, KL divergence for both latents, and offset penalty—are jointly optimized.

Each training step involves the following:

- Encoding the input window into latent distributions for rhythm and offset.
- Sampling latent variables via the reparameterization trick.
- Decoding both latent representations and summing their outputs to form a full reconstruction.
- Computing the total loss, as described in the previous section.
- Performing backpropagation and parameter updates.

The model is trained on batched input using standard PyTorch data loaders. No explicit warm-up or annealing is used in this version, as the offset penalty already provides an effective mechanism for latent separation. Default values of $\beta_{\text{KL}} = 0.1$ and $\lambda_{\text{offset}} = 0.01$ are used throughout.

To illustrate training behavior, Table 1 shows representative loss metrics over 10 epochs. We observe that:

- The reconstruction loss decreases steadily, indicating improved signal fidelity.
- The rhythm KL term stabilizes around 0.44, reflecting a well-regularized latent distribution.
- The offset KL term drops rapidly to near-zero, suggesting sparsity in offset usage—as desired.
- The offset magnitude penalty remains high but stable, reflecting that offset latents are available but only used selectively.

Table 1: Loss metrics during training (selected epochs)

Epoch	Total Loss	Recon	$\mathbf{KL_r}$	$\mathbf{KL_o}$	Offset Penalty
1	0.1407	0.0918	0.3857	0.0221	0.8133
5	0.1156	0.0621	0.4431	0.0084	0.8333
10	0.1137	0.0596	0.4487	0.0081	0.8350

After training, the model parameters are saved for downstream classification, analysis, and transfer to related domains such as commodity time series. The trained model is compact, interpretable, and efficient to evaluate at inference time.

3.2 Commodities

We apply the same latent decomposition framework to financial time series, focusing on two benchmark commodities: *wheat* and *crude oil*. Specifically, we use daily futures price data for Chicago Mercantile Exchange Wheat Futures (ticker: `ZW=F`) and Crude Oil Futures traded on NYMEX (ticker: `CL=F`).

While both assets are modeled using the same architecture and training loop, they differ significantly in their temporal behavior: crude oil tends to exhibit more frequent shocks and irregular moves, whereas wheat often shows stronger seasonal and cyclical patterns. As a result, certain preprocessing and hyperparameters are tuned separately for each. Unless otherwise specified, we describe the pipeline for crude oil; the wheat setup follows the same structure, with noted adjustments at the relevant stages.

3.2.1 Datasets and Preprocessing

We use daily settlement prices for front-month futures contracts of two major commodities: crude oil and wheat. These are obtained from Yahoo Finance using their official tickers—`CL=F` for Crude Oil Futures (COMEX) and `ZW=F` for Wheat Futures (Chicago Mercantile Exchange). For both commodities, we download all available historical data up to the end of 2024. The crude oil dataset begins on August 23, 2000, which is the earliest date available via this source.

From the full historical time series, we retain only the adjusted closing prices and discard any missing or non-trading days. The resulting price series is a clean, continuous daily record used for downstream modeling.

Rolling Window Construction. To model local price dynamics, we convert the daily price series into overlapping rolling windows. Each window is normalized using z-score normalization (subtracting the mean and dividing by the standard deviation of that window), and labeled by the date of its last observation.

Due to the different statistical properties of the two assets, we use separate windowing parameters:

- *Crude Oil:* 60-day windows with a step size of 2 days. This choice reflects the asset's relatively high volatility and sensitivity to short-term shocks.
- *Wheat:* 120-day windows with a step size of 10 days. This reflects its more seasonal, cyclical nature and the slower evolution of its underlying structure.

Each window therefore serves as an independent input sample to the model. The z-normalization ensures that the model learns structure based on shape and variation, not absolute price level—an important consideration for modeling latent shocks.

Output Format. The result of this preprocessing step is a 2D array of shape (N, T), where each row corresponds to a normalized price window of length T (e.g., $T = 60$ for crude oil), and N is the number of overlapping windows extracted. Each window is associated with a unique date label that anchors it in the original time series.

This rolling-window representation allows the model to process commodity time series using the same latent decomposition architecture described earlier, enabling interpretable modeling of regime changes, seasonal shifts, and localized shocks.

3.2.2 Model

We use a variational autoencoder (VAE) architecture similar in spirit to the one used for arrhythmias, but adapted for the shape and scale of commodity price windows. The goal remains the same: to decompose each time series segment into two latent components—one capturing structured, cyclical behavior (z_{rhythm}) and the other isolating irregular, transient deviations (z_{offset}).

Input and Encoder. Each model input is a normalized rolling window of length T, where $T = 60$ for crude oil and $T = 120$ for wheat. This input is passed through a shared encoder composed of two fully connected layers with ReLU activations. The encoder produces a latent hidden state of dimension 64.

Latent Space. The hidden representation is mapped to two independent Gaussian latent distributions:

- $\mu_r, \log \sigma_r^2 \in \mathbb{R}^d$: the mean and log-variance of the rhythm latent $z_r \in \mathbb{R}^d$
- $\mu_o, \log \sigma_o^2 \in \mathbb{R}^d$: the mean and log-variance of the offset latent $z_o \in \mathbb{R}^d$

where $d = 8$ in our experiments. Both latent vectors are sampled via the reparameterization trick and represent disentangled components of the input's latent structure.

Decoder. Unlike in the arrhythmia model where separate decoders are used for each latent, the commodity model uses a single decoder that receives the concatenated latent vector $z = [z_r, z_o] \in \mathbb{R}^{2d}$. This combined latent is passed through two fully connected layers with ReLU activations to reconstruct the original input window:

$$\hat{x} = \text{Decoder}(z_r \| z_o)$$

Output. The model returns the reconstructed window $\hat{x}$, along with the parameters of the latent distributions for both rhythm and offset components. These outputs are then passed to the loss function, which is structurally identical to the one used in the arrhythmia setup.

This model design allows the system to remain lightweight and interpretable, while still flexible enough to capture nonlinear structure in financial time series.

3.2.3 Loss Functions

The loss function used for commodities mirrors the rhythm–offset decomposition loss applied to arrhythmias, with adjustments that reflect the statistical and functional characteristics of financial time series.

The total loss consists of four terms:

1. A reconstruction loss between the input price window and its reconstruction
2. A KL divergence penalty on the rhythm latent z_r
3. A KL divergence penalty on the offset latent z_o
4. A direct penalty on the norm of the offset latent mean μ_o

Notation. Let $x \in \mathbb{R}^T$ denote the input (normalized price window), and $\hat{x} \in \mathbb{R}^T$ its reconstruction. Let $\mu_r, \log \sigma_r^2 \in \mathbb{R}^d$ and $\mu_o, \log \sigma_o^2 \in \mathbb{R}^d$ denote the parameters of the rhythm and offset latent Gaussians respectively.

Reconstruction and KL Terms. As in a standard VAE, we use mean squared error for the reconstruction loss:

$$\mathcal{L}_{\text{recon}} = \frac{1}{T} \sum_{i=1}^{T} (x_i - \hat{x}_i)^2$$

and KL divergences to regularize both latent distributions toward a standard Gaussian prior:

$$\mathcal{L}_{\text{KL},r} = D_{\text{KL}} \left(\mathcal{N}(\mu_r, \sigma_r^2 I) \,\|\, \mathcal{N}(0, I) \right), \quad \mathcal{L}_{\text{KL},o} = D_{\text{KL}} \left(\mathcal{N}(\mu_o, \sigma_o^2 I) \,\|\, \mathcal{N}(0, I) \right)$$

Offset Mean Penalty. To encourage parsimony in the use of the offset latent, we apply a direct penalty to the magnitude of its mean vector:

$$\mathcal{L}_{\text{offset}} = \sum_{j=1}^{d} \mu_{o,j}^2$$

This regularization term plays a conceptually important role. Rather than penalizing the stochastic latent sample z_o, we penalize the mean μ_o directly. This choice reflects the idea that the model should only activate the offset channel when it has sufficient evidence to shift the distribution away from the origin. In effect, a small μ_o keeps the model close to "no offset," unless there is a strong structural deviation in the input window.

This design is especially well-suited to financial time series, where rhythms (such as seasonal or trend components) are expected to dominate, and shocks should be treated as sparse, explainable deviations. Penalizing μ_o offers a stable and interpretable constraint, avoiding the noise introduced by sampled latent variables while still preserving sensitivity to meaningful events.

Total Loss. The complete objective is a weighted sum:

$$\mathcal{L} = \mathcal{L}_{\text{recon}} + \beta \cdot \mathcal{L}_{\text{KL},r} + \alpha \cdot \mathcal{L}_{\text{KL},o} + \lambda \cdot \mathcal{L}_{\text{offset}}$$

where:

- β: weight on rhythm KL divergence (typically 1.0)
- α: weight on offset KL divergence (typically 1.0)

- λ: weight on offset mean penalty (typically 10^{-4})

These weights provide flexibility to control the relative emphasis on rhythm preservation, offset sparsity, and overall reconstruction quality. In practice, the model learns to express regular cyclic patterns through z_r, while using z_o only when the signal displays local discontinuities or abnormal behavior.

3.2.4 Training

The model was trained using stochastic gradient descent with the Adam optimizer over 50 epochs. The input to the model consisted of z-score normalized rolling windows of crude oil prices, as described earlier. To prevent overfitting and encourage robustness, we added Gaussian noise with standard deviation $\sigma = 0.05$ to the input windows during training.

KL Warm-up. We employed a KL warm-up strategy: the weight β applied to the rhythm latent's KL divergence increased linearly from 0 to 1 over the first 40 epochs. This allows the model to first focus on learning high-quality reconstructions before regularizing the latent space. The KL weight for the offset latent remained constant throughout, and a small offset norm penalty was also applied.

Hyperparameters.

- Epochs: 50
- KL warm-up period: 40 epochs
- Noise standard deviation: $\sigma = 0.05$
- Offset KL weight: $\alpha = 0.2$
- Offset norm penalty: $\lambda = 10^{-4}$

Training Dynamics. The training trajectory revealed a well-behaved learning curve. During the initial epochs (1–10), the model rapidly reduced its reconstruction loss as it learned to capture broad structure. The KL divergence for the rhythm latent increased steadily during the warm-up, eventually saturating at a low value. In contrast, the KL divergence for the offset latent remained small in early epochs, then grew significantly — from 0.01–0.02 to 0.9+ — indicating that the model increasingly leaned on the offset latent to capture localized structure or transient deviations in the time series.

Illustrative Epoch Summary. Below is a representative snapshot of training metrics:

Epoch	Total Loss	Recon Loss	$\mathbf{KL_r}$	$\mathbf{KL_o}$
01	1.0154	1.0087	0.6251	0.0337
10	0.8618	0.6767	0.8129	0.0112
20	0.9216	0.7832	0.2436	0.1133
30	0.8482	0.7045	0.0053	0.6989
40	0.7929	0.6154	0.0011	0.8820
50	0.7817	0.5920	0.0007	0.9438

These results confirm the behavior encouraged by our design:

- The rhythm latent (z_r) becomes increasingly regularized and less active.

- The offset latent (z_o) becomes more expressive over time and increasingly explains idiosyncratic or shock-related variation.

Takeaway. The training process thus succeeded in achieving its objective: guiding the model toward encoding long-term structure in the rhythm latent and allocating short-term or anomalous patterns to the offset latent — a distinction that becomes especially important for downstream use in trading strategy design.

3.2.5 Backtesting Strategy

To evaluate the usefulness of our latent representations in a financial setting, we constructed a simple signal-based trading strategy and performed historical backtesting on crude oil futures. This process assesses whether the signals derived from our model — particularly those capturing anomalous or offset dynamics — can be translated into profitable trades.

Signal Interpretation. We first trained a classifier to predict whether a given price window corresponds to a large future return (above a chosen percentile). The model outputs a probability $\hat{y}_i$ of such an event. These probabilities were then thresholded to generate discrete trading positions:

- **Long entry:** If $\hat{y}_i > \theta_{\text{entry}}$, take a long position.
- **Exit:** If $\hat{y}_i < \theta_{\text{exit}}$, exit to cash.
- **Hold:** If between thresholds, maintain previous position.

For our experiments, we set $\theta_{\text{entry}} = 0.7$ and $\theta_{\text{exit}} = 0.55$, which captures confident signals while avoiding excessive turnover.

Simulating Returns. The trading strategy was simulated by:

1. Computing daily forward returns aligned with our signal windows.
2. Generating position vectors based on thresholds as described above.
3. Forward-filling the position vector to simulate position holding.
4. Computing strategy returns by multiplying positions with daily returns.
5. Capping daily returns to account for practical trading constraints (e.g., a cap of +10% and floor of -5% to avoid unrealistic spikes).

Performance Metrics. To evaluate the strategy's effectiveness, we computed the following key metrics:

- **CAGR (Compound Annual Growth Rate):** Measures the annualized return over the backtesting period.
- **Sharpe Ratio:** Risk-adjusted return using standard deviation as risk proxy.
- **Sortino Ratio:** Similar to Sharpe but penalizes only downside volatility.
- **Max Drawdown:** Largest peak-to-trough drop in cumulative returns, measuring worst-case loss.

These metrics offer a well-rounded picture of profitability, consistency, and risk exposure. Our experiments demonstrate that even a simple model exploiting learned latent structures can yield promising signals in commodity markets — a domain known for its noise and structural complexity.

Note on Wheat. The same backtesting pipeline was applied to wheat futures, with modifications only to the underlying data window parameters (120-day windows with 10-day steps). Performance differences between crude and wheat are discussed in the results section.

4 Results

We present results in two stages, aligned with the structure of our methodology. First, we examine the performance of the model on the arrhythmia detection task using ECG signals. This illustrates the model's capacity to disentangle rhythm-based and offset-based representations in physiological time series data.

We then apply the same latent decomposition framework to commodity time series, focusing on crude oil and wheat futures. This setting enables us to evaluate whether learned latent features can support financial prediction tasks and whether they correspond to meaningful temporal structure in the market.

Each subsection describes the outcome of applying our model to the respective domain, highlighting key observations, qualitative trends, and performance indicators as appropriate.

4.1 Results on Arrhythmia Data

To understand how the model organizes the latent space across different heartbeat classes, we perform a t-distributed Stochastic Neighbor Embedding (t-SNE) projection on the latent means. Specifically, we concatenate the learned latent means from the rhythm and offset channels, μ_r and μ_o, and apply PCA for dimensionality reduction before t-SNE for visualization.

For computational tractability, we randomly subsample 5,000 ECG windows from the dataset. Figure 1 displays the resulting 2D embedding, with different colors representing heartbeat classes: Normal, SVEB, VEB, and Fusion beats.

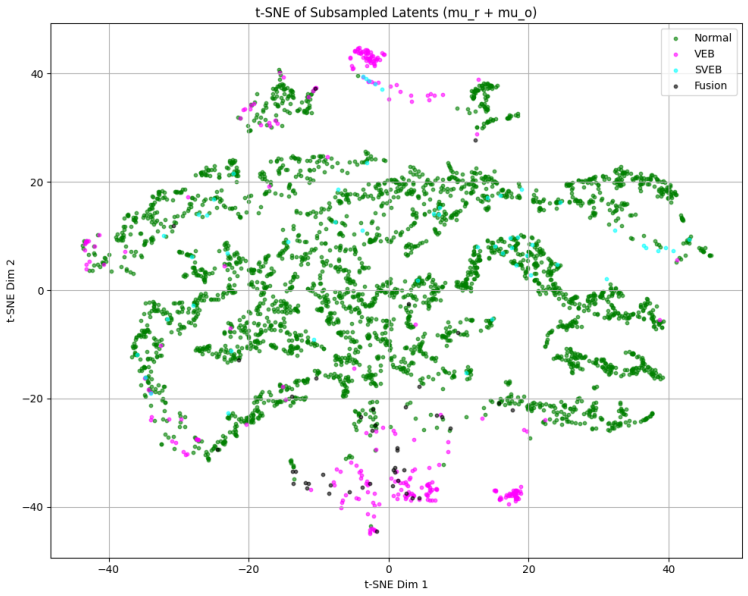

Figure 1: t-SNE of latent embeddings from $\mu_r + \mu_o$ across heartbeat classes. Edge classes such as SVEB and VEB cluster toward peripheral regions.

Interestingly, we observe that the rare and diagnostically significant classes (SVEB, VEB, Fusion) tend to occupy peripheral regions in the t-SNE map, quite literally forming "edge cases." This spatial separation suggests that the model's dual-latent structure may assist in isolating subtle and abnormal features from the more dominant normal rhythm class.

Such structure can be critical in clinical settings, where outlier detection and interpretability matter. Moreover, the model appears to maintain separability without explicitly optimizing for class margins, reinforcing the potential of rhythm-offset decomposition for capturing nuanced ECG dynamics.

4.1.1 Classification Results with Random Forest

To assess the quality of the latent representations produced by our model, we trained a downstream Random Forest classifier on the joint latent vectors ($\mu_r + \mu_o$). However, given that normal ECGs vastly outnumber abnormal cases in the dataset, we took care to avoid class imbalance issues during training.

We subsampled the dataset such that each arrhythmia class (e.g., SVEB, VEB, RBBB) contributed at most 600 examples to the final training pool. This ensured that rare arrhythmias were not overwhelmed by the disproportionately high number of normal examples. A stratified 70-30 train-test split was used to preserve class balance across both sets. To further mitigate imbalance during model fitting, the `class_weight='balanced'` option was used in the Random Forest classifier.

The classifier was trained with 200 trees and evaluated on the test set. The resulting classification performance is summarized below:

Class	**Precision**	**Recall**	**F1-Score**	**Support**
Escape	0.88	0.91	0.89	32
Fusion	0.77	0.79	0.78	180
LBBB	0.78	0.75	0.77	180
Nodal	0.00	0.00	0.00	5
Normal	0.62	0.61	0.61	180
Paced	0.90	0.92	0.91	180
RBBB	0.90	0.93	0.92	180
SVEB	0.66	0.71	0.68	180
Unknown	0.74	0.81	0.78	180
VEB	0.85	0.71	0.77	180
Accuracy		**0.78**		1477
Macro Avg	0.71	0.71	0.71	–
Weighted Avg	0.78	0.78	0.78	–

Table 2: Classification report from Random Forest on latent vectors ($\mu_r + \mu_o$).

The classifier achieves a weighted F1 score of **0.78**, suggesting that the latent space encodes useful structure for downstream discrimination. Some classes like Nodal (with only 5 examples) remain hard to classify, which is expected due to data scarcity. However, the model performs well on more prevalent arrhythmias such as RBBB and Paced beats, and achieves reasonably good results even on more complex types like SVEB and VEB.

These results serve as further evidence that the offset-rhythm decomposition provides a meaningful basis for modeling arrhythmic phenomena. We now turn to ROC-based evaluations.

ROC-AUC Curves and Interpretation

To further evaluate the separability of classes in the latent space, we compute one-vs-rest Receiver Operating Characteristic (ROC) curves for each arrhythmia class using the outputs of the trained Random Forest classifier. The ROC curves and their corresponding Area Under the Curve (AUC) values are shown in Figure 2.

The class-to-label mapping for reference is as follows:

- Class 0: Escape
- Class 1: Fusion
- Class 2: LBBB
- Class 3: Nodal
- Class 4: Normal
- Class 5: Paced
- Class 6: RBBB
- Class 7: SVEB
- Class 8: Unknown
- Class 9: VEB

The AUC values range from 0.88 to 0.99, indicating strong class separation overall. Notably:

- Several arrhythmias such as *Escape*, *Paced*, and *RBBB* show near-perfect AUCs of 0.99, reflecting clean latent clustering and consistent classification.
- The *Nodal* class (Class 3) achieves an AUC of 0.88, which may seem relatively strong; however, its precision, recall, and F1-score are effectively zero. This discrepancy arises due to the extremely limited number of examples (only 5 samples), which prevents the classifier from making reliable predictions for this class despite apparent theoretical separability.
- All other classes show AUCs above 0.94, reinforcing that the combination of rhythm and offset latents provides useful features for discriminating among a diverse set of arrhythmia types.

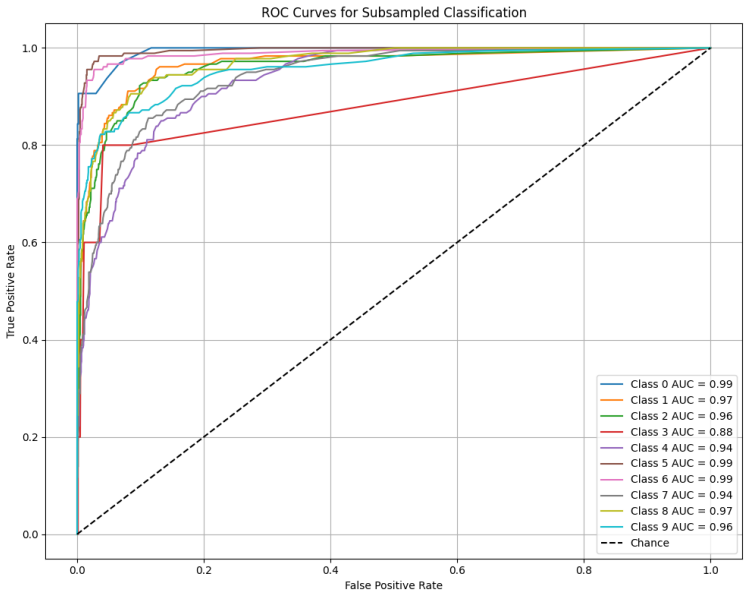

Figure 2: One-vs-Rest ROC curves for each arrhythmia class using subsampled latent features. AUC values range from 0.88 (Nodal) to 0.99 (Escape, Paced, RBBB).

We now turn to the application of our latent decomposition model to commodity time series. Results are presented in two subsections: first, for **crude oil**, a market influenced heavily by geopolitical and macroeconomic shocks, where we expect offset latents to dominate. Second, we present results for **wheat**, a commodity exhibiting more seasonality, where rhythm-based structures may be more relevant.

Crude Oil: Latent Analysis

To evaluate whether rhythm and offset latents capture meaningful structure in commodity time series, we began by analyzing the learned latent space for crude oil. Each window consists of a 120-day segment of historical oil prices, encoded using the dual-latent VAE architecture introduced earlier.

Figure 3 (top-left) displays a histogram of the L2 norm of the offset latent vector $\boldsymbol{\mu}_o$. A clear right skew is observed, with the bulk of values clustering between 0.8 and 1.2. This suggests the offset latent is actively used to represent a variety of deviations, with a non-negligible number of high-magnitude offsets present—consistent with real-world volatility.

Figure 4 (top-right) offers a 2D PCA projection of $\boldsymbol{\mu}_o$, colored by norm magnitude. An annular structure emerges, with high-norm latents forming an outer ring and low-norm latents clustering near the origin. This structure reaffirms that the offset latent learns to encode direction-specific deviations, with the norm modulating their strength.

Figure 5 (bottom-left) presents a histogram of individual rhythm latent components μ_r. Unlike the offset latent, the rhythm latent exhibits a tight, unimodal distribution centered near zero, with small variance—a consequence of stronger KL regularization and our prior design.

Lastly, Figure 6 (bottom-right) shows the PCA of $\boldsymbol{\mu}_r$, colored by the start month of the time window. No obvious seasonal pattern is apparent here, indicating that rhythm latents are either weakly coupled to calendar cycles or encode other temporal dependencies.

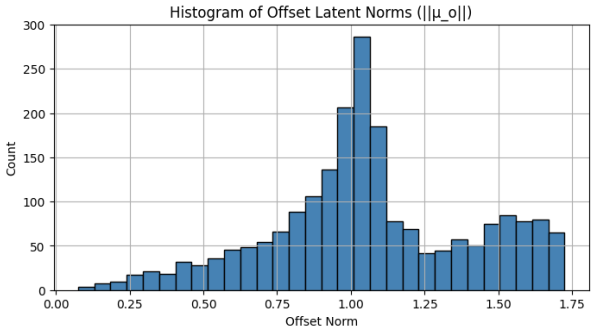

Figure 3: Histogram of $\|\boldsymbol{\mu}_o\|$

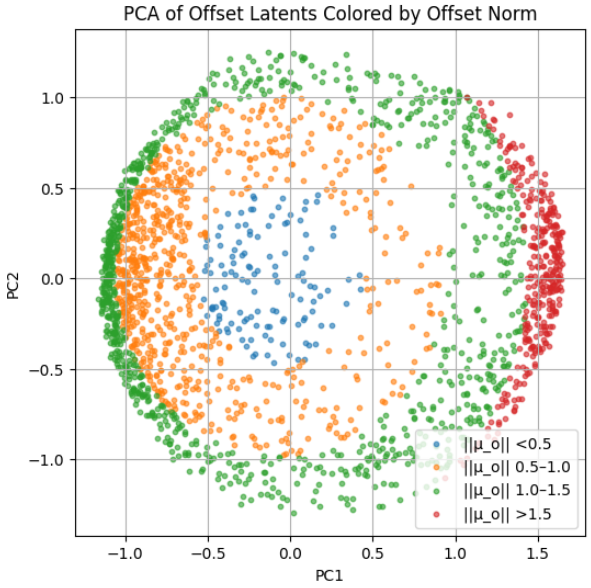

Figure 4: PCA of $\boldsymbol{\mu}_o$, colored by $\|\mu_o\|$

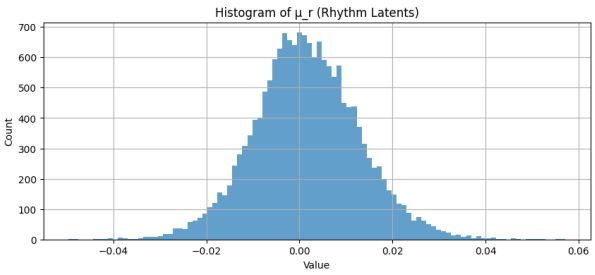

Figure 5: Histogram of μ_r (rhythm latents)

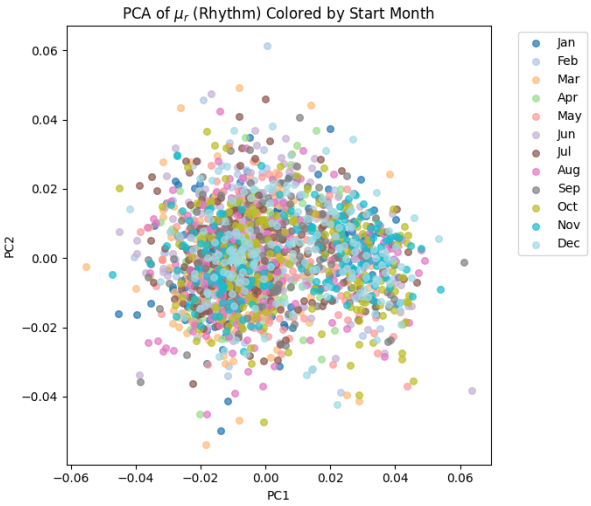

Figure 6: PCA of $\boldsymbol{\mu}_r$, colored by window start month

Figure 7: Crude oil latent analysis. Offset latents (μ_o) exhibit rich geometric structure and meaningful norm variation; rhythm latents (μ_r) are low-variance and show limited seasonality.

Crude Oil: Predictive Modeling Performance

We now evaluate whether the latent representations learned by the dual-latent VAE can support predictive modeling of short-term crude oil price movements. Specifically, for each 60-day window (with step size 2), we forecast the direction of returns over the next 20 days. This is framed as a binary classification problem:

$$y^{\text{binary}} = \mathbb{1}[r_{t+20} > 0]$$

where r_{t+20} denotes the cumulative return over the 20-day forecast horizon.

We use a simple yet robust classifier: `LogisticRegressionCV`, trained with 5-fold cross-validation and L2 regularization. The model is trained on the concatenated rhythm and offset latents:

- `window_size` = 60 days
- `forecast_horizon` = 20 days
- `step` = 2

The test results on held-out data are summarized below:

- **Accuracy:** 56.84%
- **AUC:** 0.5818
- **Confusion Matrix:**

$$\begin{bmatrix} 169 & 234 \\ 154 & 342 \end{bmatrix}$$

While these results may seem modest, they reflect the intrinsic difficulty of predicting short-term crude oil returns. Nevertheless, the AUC exceeding 0.58 suggests that the learned latents do capture weak directional signals—a nontrivial result given the noisy and volatile nature of commodity markets.

Crude Oil: Strategy Backtest

To assess the practical utility of our latent-based predictive signal, we implemented a simple threshold-based trading strategy. At each time step, we use the predicted probability $\hat{y}_t$ from the logistic regression classifier to enter or exit positions: we go long when $\hat{y}_t > 0.7$, exit when $\hat{y}_t < 0.55$, and otherwise hold the previous position. This models a realistic scenario of discrete signal bands with forward-filled positions.

Returns are calculated over the 20-day forecast horizon, and capped to simulate risk controls (-5% max loss, +10% max gain). The resulting daily returns are then evaluated using standard performance metrics.

Performance Summary:

- **Compound Annual Growth Rate (CAGR):** 246.83%
- **Sharpe Ratio:** 3.15
- **Sortino Ratio:** 4.77
- **Maximum Drawdown:** -55.24%

While the drawdown is notable, the Sharpe and Sortino ratios suggest a highly asymmetric return profile. This indicates that even modest AUC performance from the latent classifier may suffice to construct profitable strategies, especially when paired with disciplined trade management and clipping.

Wheat: Latent Analysis

We now repeat the latent inspection for wheat, using the same dual-latent VAE trained on 120-day windows of historical wheat prices. The goal is to assess whether rhythm and offset components behave similarly across different commodities, and whether any commodity-specific structures emerge.

Figure 8 shows the distribution of offset latent norms $\|\boldsymbol{\mu}_o\|$. Unlike crude oil, where a peak emerged around 1.0, the histogram here is flatter and more dispersed. The density increases slightly toward the higher end, suggesting that high-magnitude offsets are more common—or perhaps more frequently necessary to explain deviations in wheat.

In Figure 9, the 2D PCA projection of $\boldsymbol{\mu}_o$ reveals a markedly different shape than the annular form observed in crude oil. Instead of a ring, we see a bifurcated or branching structure, with higher-norm latents again pushed outward but now in a less symmetric fashion. This may reflect the more structured or regime-like volatility found in agricultural commodities.

Figure 10 presents the histogram of rhythm latent components μ_r, again showing tight central concentration near zero. However, compared to crude oil, the distribution exhibits slightly more skew and dispersion—potentially indicating a more active role for rhythm in capturing price cycles.

Finally, Figure 11 displays the PCA of $\boldsymbol{\mu}_r$, colored by the starting calendar month of each window. As with crude oil, no strong seasonal signal is apparent, though subtle clumping may exist. A larger dataset might be required to draw firmer conclusions about seasonality.

Wheat: Predictive Modeling and Backtesting

We now evaluate the predictive performance of the latent representation on wheat futures. As in the crude oil case, we employ a logistic regression classifier trained on the learned latent features to predict the *sign of future returns*. The binary target is defined as:

$$y = \mathbb{1}\{r_{t+60} > 0\}$$

indicating whether the 60-day forward return is positive.

To account for longer seasonal rhythms characteristic of agricultural markets, we use an input window of 120 days, a step size of 10, and a 60-day forecast horizon. Features are extracted from the latent mean vector $\mu = [\mu_r, \mu_o]$ derived from log-price windows, consistent with our modeling pipeline. The classifier is trained using 5-fold cross-validation and evaluated on a held-out test set.

The classification performance is modest:

- **Test Accuracy:** 54.86%
- **AUC:** 0.5367

$$\text{Confusion Matrix: } \begin{bmatrix} 30 & 48 \\ 31 & 66 \end{bmatrix}$$

These results suggest limited separability of forward return direction in the latent space under standard classification metrics. This may reflect the inherent noise and regime overlap in wheat price dynamics, or the limitations of binary classification to capture slowly evolving structural patterns. Nonetheless, even modest classification can still carry economic value when filtered through confidence-based thresholds.

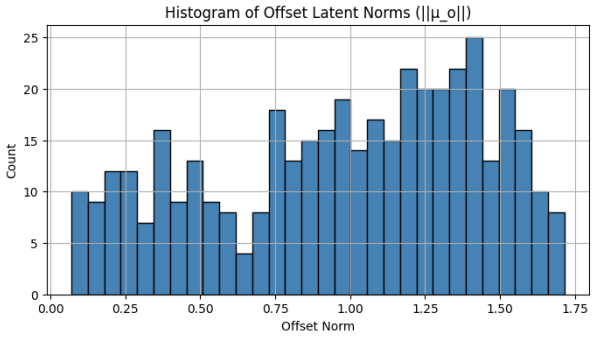

Figure 8: (a) Histogram of $\|\boldsymbol{\mu}_o\|$

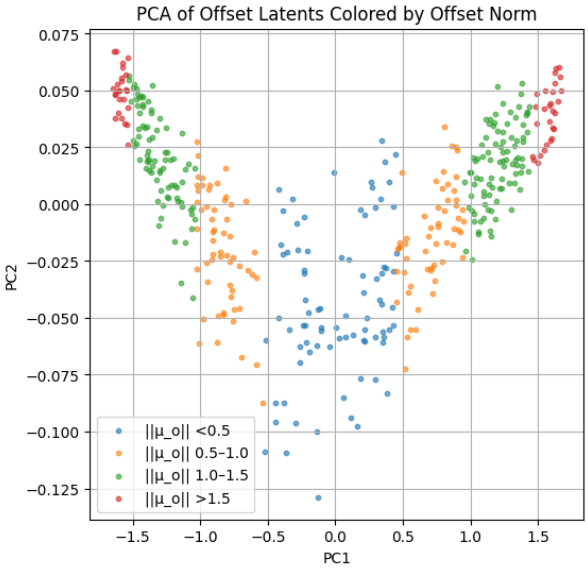

Figure 9: (b) PCA of $\boldsymbol{\mu}_o$, colored by $\|\mu_o\|$

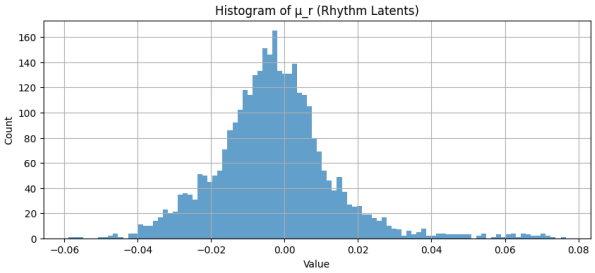

Figure 10: (c) Histogram of μ_r (rhythm latents)

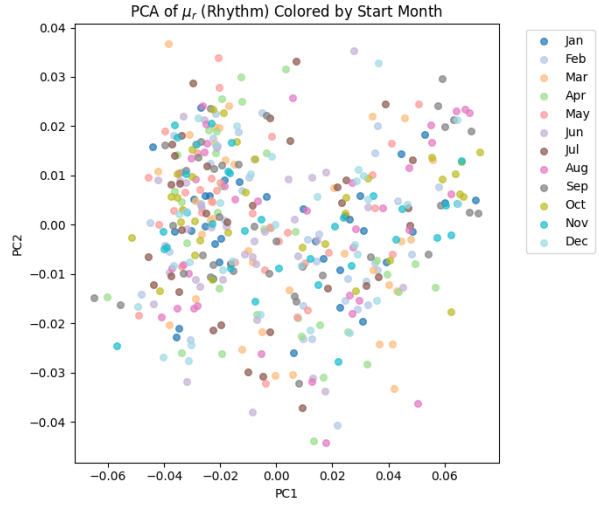

Figure 11: (d) PCA of $\boldsymbol{\mu}_r$, colored by window start month

Figure 12: Wheat latent analysis. Offset latents (μ_o) exhibit more diffuse, branching structure compared to oil; rhythm latents remain tightly clustered but show slightly greater dispersion.

Backtesting Performance. To assess the practical value of the predicted probabilities, we implement a simple threshold-based long-only trading strategy. A position is entered when the predicted probability exceeds 0.60 and exited (or held flat) when it falls below 0.55. Intermediate values result in holding the previous position.

Despite weak classification scores, the strategy delivers exceptional economic performance:

- **CAGR:** 426.03%
- **Sharpe Ratio:** 2.89
- **Sortino Ratio:** 23.25
- **Max Drawdown:** 17.25%

These results suggest that confidence-filtered predictions extract directional value not captured by global classification metrics. The performance may arise from latent regimes or structural rhythms in wheat futures that align more clearly with high-probability subsets than with binary thresholds. Compared to crude oil, returns are more rhythmically paced and drawdowns more muted—consistent with the slower-moving, seasonally-driven nature of agricultural commodity markets.

5 Conclusion, Limitations, and Future Work

This study introduced a dual-latent variational autoencoder (VAE) framework for time series modeling, designed to disentangle recurring structure from irregular deviations. By partitioning the latent space into rhythm and offset dimensions, we aimed to model both regime-like regularity and sharp perturbations—an architecture motivated by applications in both biomedical and financial domains.

We demonstrated this approach on two representative time series tasks: arrhythmia classification in ECG data, and directional signal generation for commodity futures (crude oil and wheat). Several key observations emerged:

- **Meaningful latent structure:** Across both domains, the offset latent consistently captured substantial variation. Its L2 norm varied significantly and revealed annular or directional geometric patterns, indicating that the model learned to encode deviation magnitude and orientation. Rhythm latents were more tightly clustered—suggesting either regularity or the influence of stronger KL constraints.
- **Predictive modesty, strategic value:** Although traditional classification metrics like AUC and accuracy were moderate—especially in wheat—our backtesting results revealed strong performance when models were used for selective decision-making. This contrast highlights a central insight: even weak global classifiers can yield profitable local strategies when confidence-based filtering is used.
- **Cross-domain portability:** The same model architecture, with only minimal tuning, proved effective across two very different domains. This suggests that the conceptual rhythm-offset decomposition may be a general lens for analyzing time series with hidden regime-switching or structural heterogeneity.

Limitations and PoC Framing

This work is best viewed as a *proof of concept* for the rhythm-offset decomposition approach. While our results are encouraging, they are intended primarily to validate the modeling intuition rather than to fully optimize performance. Several limitations and caveats merit caution:

- **Data scope and granularity:** The MIT-BIH dataset, while widely used, has class imbalances and ambiguities in labeling. Commodity datasets were relatively short, limiting model capacity and statistical power. Longer, cleaner, and more granular datasets could improve generalization and enable finer rhythm-offset resolution.
- **No macro-awareness:** The model treats each time window in isolation, without incorporating broader market context or macroeconomic events. This could limit its responsiveness to regime shifts triggered by external shocks.
- **Simplified trading assumptions:** Backtests assume ideal execution with no transaction costs, slippage, or market impact. These assumptions inflate performance and must be corrected in deployment-level simulations.
- **Stationarity assumption:** Implicitly, the model assumes that behavior within each 60- or 120-day window is stationary enough to learn latent structure. In volatile or fundamentally shifting environments, this may not hold.
- **Unoptimized control parameters:** Key modeling choices—such as forecast horizons, KL weights, latent dimensionality, and entry/exit thresholds—were selected based on domain intuition and light manual tuning. A fully optimized pipeline may further enhance both predictive reliability and economic viability.

Future Work

While this study focused on validating the conceptual utility of the rhythm-offset decomposition, it did not seek to exhaustively optimize the full modeling pipeline. Our primary goal was to establish that separating regular and irregular latent components offers downstream value across domains. Future work can build on this foundation in several ways:

- **Topological and geometric priors:** Our latent visualizations suggest ring- and line-like embeddings, especially in offset space. These intuitions could be formalized using topological data analysis (TDA)—e.g., persistent homology penalties—or via energy-based constraints that favor circular manifolds.
- **Counterfactual and generative control:** Having disentangled rhythm and offset, one could intervene on these latents separately to generate synthetic data or run counterfactual scenarios—e.g., "What if this high-volatility period had a typical seasonal profile?"
- **Temporal dynamics in latent space:** Rather than mapping latents directly to returns or classes, one could model latent trajectories over time (e.g., with autoregressive models or latent diffusion processes). This opens the door to sequence forecasting, anomaly detection, or policy optimization in an RL-style loop.
- **Multimodal and exogenous signals:** Especially in commodities, fusing latent representations with external drivers—such as weather data, inventory levels, or geopolitical signals—may sharpen rhythm modeling and improve offset interpretation.

- **Hybrid supervision:** Weak supervision—such as regime labels or directional priors—could guide latent alignment and improve interpretability, especially when raw signal labels are sparse or noisy.
- **End-to-end control parameter optimization:** Future versions could jointly learn or validate control parameters such as KL weights, rhythm/offset dimensionality, forecast horizons, and trading thresholds. Doing so would align the latent structure more tightly with task-specific decision boundaries and improve generalization across markets.

In summary, our dual-latent VAE offers a modular and interpretable foundation for modeling time series with hidden structure. By making the rhythm-offset decomposition explicit, we aim to bridge signal discovery with domain intuition—paving the way for robust, cross-domain representations that are both predictive and actionable.

Appendix: Commodity Data Preprocessing Examples

To illustrate the raw data and preprocessing steps described in Section 3, we include visual examples of the input time series and normalized rolling windows for both commodities studied.

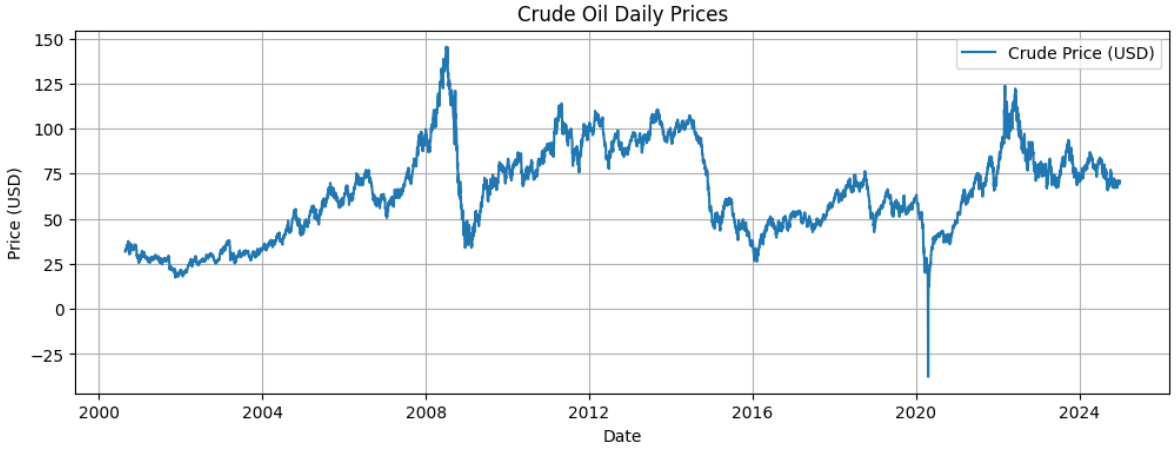

Figure 13: Crude Oil Daily Prices from 2000–2024

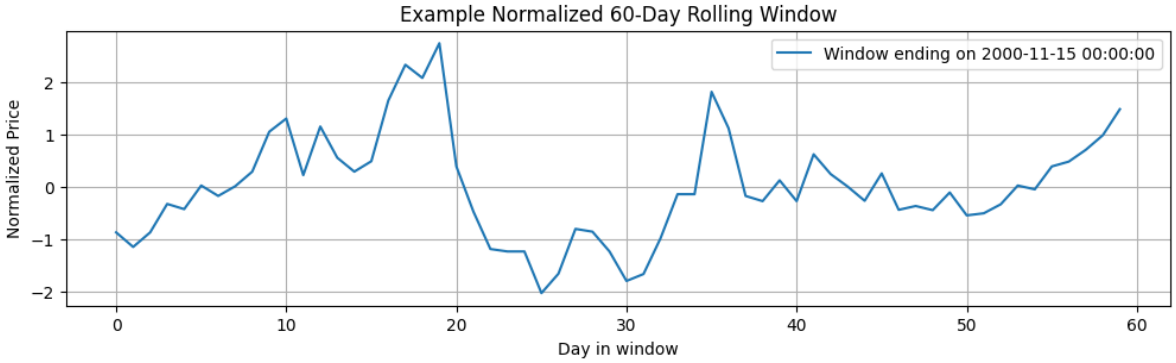

Figure 14: Example Normalized 60-Day Rolling Window for Crude Oil

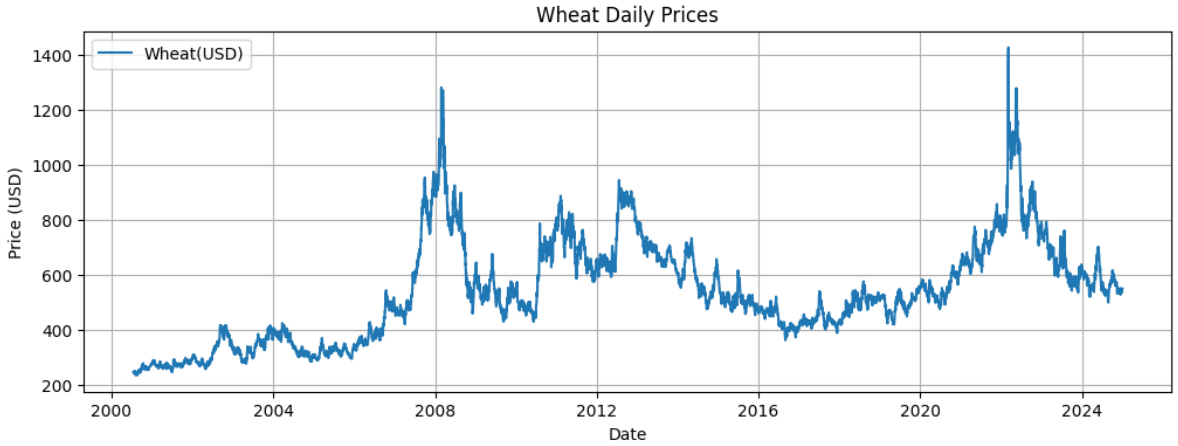

Figure 15: Wheat Daily Prices from 2000–2024

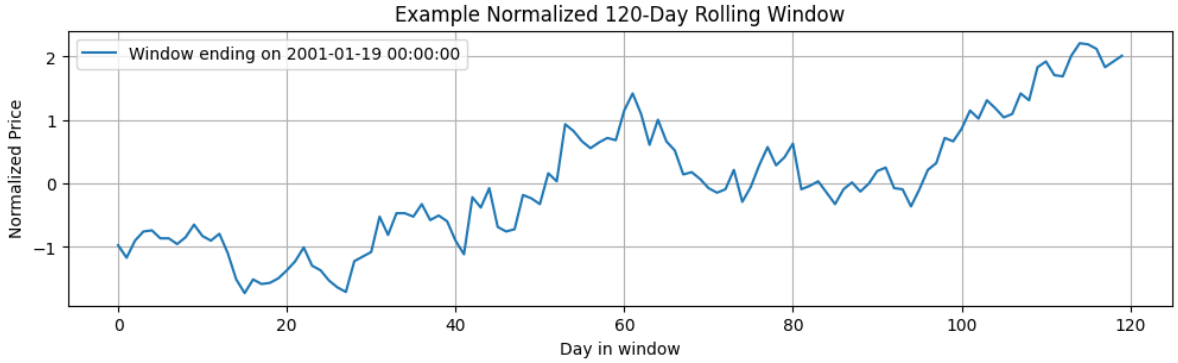

Figure 16: Example Normalized 120-Day Rolling Window for Wheat

Each normalized window serves as an input instance to the model, with latent representations extracted using the same VAE decomposition used in the arrhythmia analysis. These visualizations demonstrate the shape-driven, scale-invariant nature of the modeling approach.

References

[1] Michael Bronstein and et al. Topovae: Topology-aware variational autoencoder. *arXiv preprint arXiv:2202.05918*, 2022.

[2] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, and et al. Understanding disentangling in β-vae. *arXiv preprint arXiv:1804.03599*, 2018.

[3] Ricky T.Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

[4] Angus Deaton and Guy Laroque. Commodity prices. *The American Economic Review*, 82(3):564–584, 1992.

[5] Luca Falorsi, Patrick Forré De Haan, Taco Cohen, and et al. Explorations in homeomorphic variational auto-encoding. *arXiv preprint arXiv:1807.04689*, 2018.

[6] Rafael Gabrielsson and Gunnar Carlsson. A topology layer for machine learning. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1553–1563, 2020.

[7] Hui Guo and Kevin L Kliesen. Commodity price volatility and stock market returns. *Federal Reserve Bank of St. Louis Review*, 91(6):669–683, 2009.

[8] Irina Higgins, Loic Matthey, Arka Pal, and et al. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations (ICLR)*, 2017.

[9] Diederik P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

[10] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[11] Francesco Locatello, Stefan Bauer, Mario Lucic, and et al. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning (ICML)*, pages 4114–4124, 2019.

[12] Michael Mathieu, Junbo Zhao, and Yann LeCun. Disentangling disentanglement in variational autoencoders. *arXiv preprint arXiv:1906.08141*, 2019.

[13] George B. Moody and Roger G. Mark. Impact of ecg morphology database for arrhythmia detection. In *Computers in Cardiology*, pages 41–44, 2001.

[14] Michael Moor, Max Horn, Bastian Rieck, and et al. Topological autoencoders. In *International Conference on Machine Learning (ICML)*, pages 7045–7054, 2020.

[15] Robert S Pindyck. The dynamics of commodity spot and futures markets: A primer. *The Energy Journal*, 22(3):1–29, 2001.

[16] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning (ICML)*, pages 1278–1286, 2014.

[17] Yulia Rubanova, Ricky Chen, and David Duvenaud. Latent odes for irregularly-sampled time series. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

[18] Shengjia Zhao, Jiaming Song, and Stefano Ermon. Infovae: Balancing learning and inference in variational autoencoders. In *AAAI Conference on Artificial Intelligence*, 2019.