Generative machine learning models for scenario simulation

Vadim Strizhov

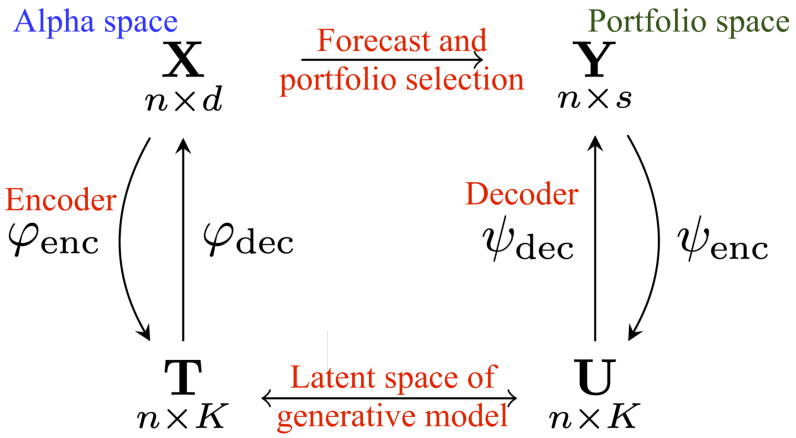
This paper describes the principles of generative modeling. Its goal is to create a dynamic portfolio scheduling system. It selects subsets of alphas with significant predictive power to indicate the optimal portfolio. Below, the main parts of the system are listed.

1. Canonical correlation analysis

We have two domains of time series: the space of alphas and the space of portfolios. The problem is forecasting a change in the portfolio itself, given the historical trajectories of alphas. This problem includes the selection of alphas, the selection of portfolio items, and the analysis of time series cross-dependencies in alpha space and in portfolio space. A useful solution to this problem is Canonical correlation analysis. In deep learning, it is called the sequence or autoregressive transformer approach.

Benefits of the CCA:

1. It approximates both the design (alpha) space and the behavior (portfolio) space
2. It reduces the dimensionality of these two spaces, selecting the most informative alphas and the optimal structure of the portfolio
3. It selects a subset of alphas with significant predictive power to indicate the portfolio



The figure shows the forecasting and portfolio model. The alpha space collects phase trajectories or graph representations of the alphas. Portfolio space collects the state of the instruments that form a portfolio. The encoder and decoder extract dependencies in both spaces.

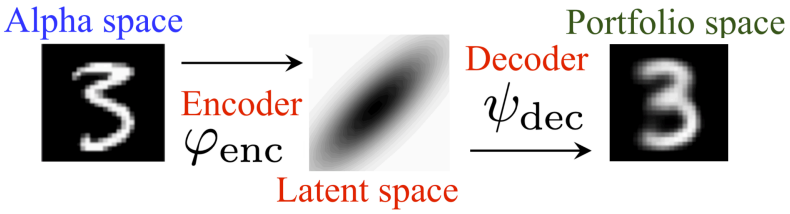
2. Generative models

Generative models decompose the deterministic and stochastic components in the time series. They have useful mechanisms to model possible scenarios of dynamic trading. We propose to estimate changes in expected returns, changes in parameters of forecasting models, and changes in distributions of resudy. It delivers more detailed scenarios of dynamic trading.

Classes of generative models
Models with tractable density: Normalizing flows and Autoregressive models
Models with approximate density: Probabilistic diffusion models and Variational autoencoders

Benefits of the generative models:

1. It creates new data points that resemble the training data and simulates realistic scenarios, keeping the original data distributions
2. It transfers dependencies in data spaces from one model to another, and assigns initial parameters for pre-training of alternative models



The figure shows the reconstruction of the data distribution and the data generation in the Canonical correlation analysis. The encoder part extracts and analyses distributions and dependencies in the alpha state space. The decoder part generates realizations of time series in the portfolio state space. Since the CCA assumes a forward-backward inference scheme, the latent distribution represents dependencies from both spaces. It is fully compatible with the Normalizing Flows and Probabilistic Diffusion Models.

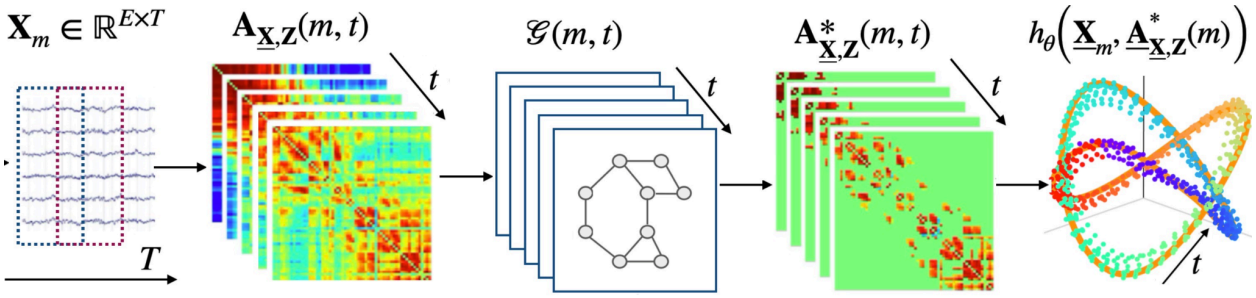
3. Graph and dynamic barycenter models

There are three assumptions about the data to construct a graph model:

1) each time series is relatively short, with thousands of samples,
2) it has a significant variance,
3) there is a significant covariance in clusters of time series.

So, the model is a dynamic graph convolutional neural network, which keeps relations between time series and tracks changes in these relations. We use Riemannian geometry methods, connecting the curvature of space and the graph structure.

To decide on portfolio structure, we propose to generate changes in the time series relation graphs.

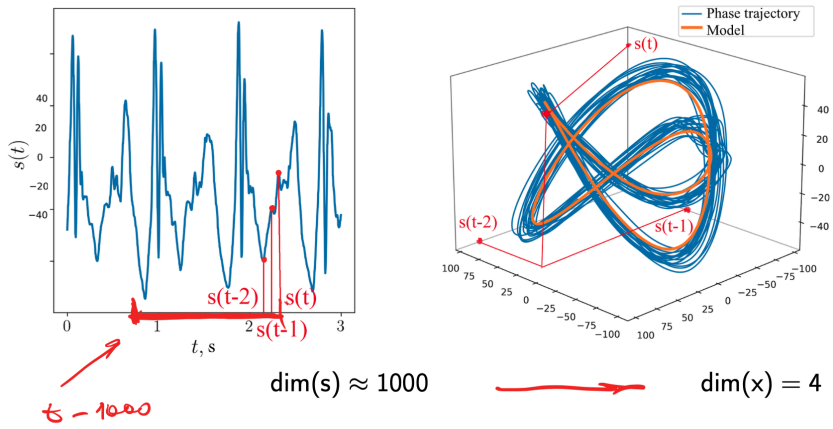


The figure shows the composition of models to forecast a set of time series and changes in their cross-dependencies. First, we construct the state space of the time series and construct a cross-dependency matrix. It changes in time. Second, we extract and prune the graph of clusters and reconstruct the behavior to select alphas with significant forecasting power.

4. State-space models

Due to the high covariation in time series, we will use Singular spectrum analysis. It defines the phase trajectories of the system in its state space. The dimensionality of these spaces is excessive. To reveal dependencies in data (reconstruct embedding manifolds), we reduce this dimensionality.
We use discrete and continuous time and space representations to reconstruct embedding manifolds. In the discrete case, we use tensor decomposition. We use a tensor structure to predict a set of trajectories instead of a single time series. Namely, tensor-train decomposition brings decent reconstruction quality and easily extends toward deep neural networks.

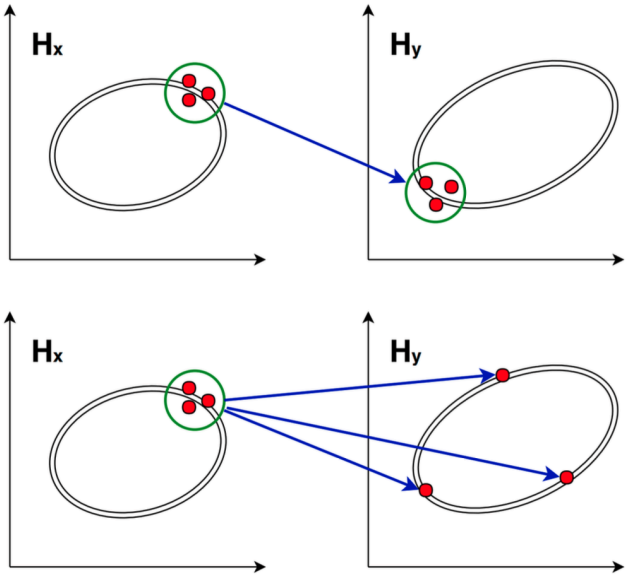
For the continuous case, we use controlled differential equations. They reflect the continuous topology of neural network structure involving automatic differentiation methods. The future combination of these two cases significantly impacts the modeling of heterogeneous signals.



The figure shows how a time segment is represented in the state space. The sequence of the time segments defines the phase trajectory. The dimensionality reduction reveals dependencies in the orange model.

5. Cross-convergence mapping

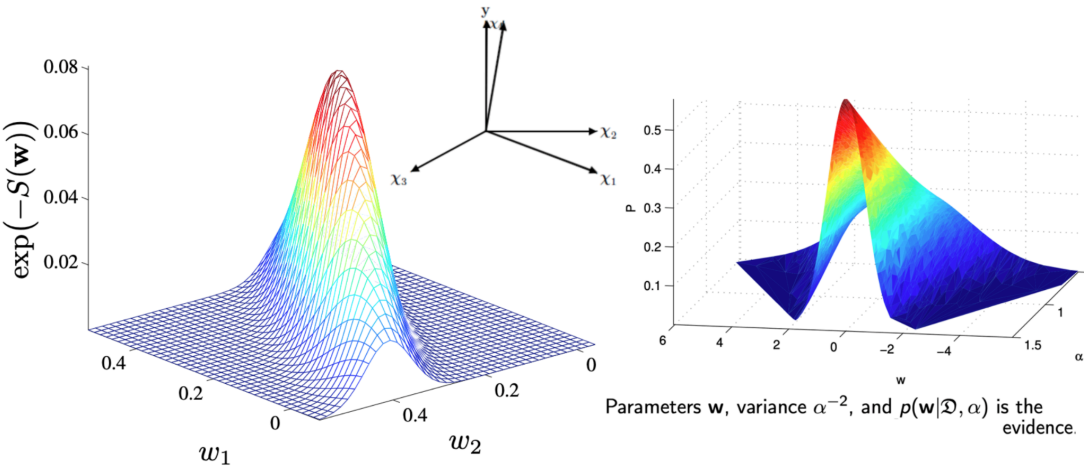
To estimate distributions in data, we shall construct a probabilistic metric space. The basic variant is the correlation or cointegration of the time series. A tricky variant is to estimate casualties between phase trajectories.



The figure shows that we observe casualties if we find a Lipschitz map between two phase trajectories. The Lipshitz coefficient serves as a distance between time series. We observe scattering in the neighborhood for independent trajectories, as shown below.

6. Generative Bayesian model selection

Bayesian model selection relies on the analysis of the model parameters. We shall analyze parameters, hyperparameters (distributions), and the model structure to select models. Optimization of neural networks with large amounts of hyperparameters is computationally expensive. We use Gaussian process regression models to analyze data uncertainties and transfer data distributions between models.



The left figure shows that two model parameters are dependent; the first brings precision over stability. The high correlation in the data reduces stability in model parameters. The right figure shows that the hyperparameter regulates the precision-stability tradeoff.

Benefits of the model selection

1. It elects robust alphas with significant predictive power
2. It elects a portfolio according to dependencies between its items
3. Controls the model complexity, consistent with the sample size

7. Practice of constructing deep learning models

It is a practical part of constructing pipelines and optimizing deep neural networks. Also, it highlights when we use ready-to-go Transformers or LSTMs or make them from scratch to obtain optimal complexity. Various pipelines will be discussed.

8. List of useful code repositories

A technical system shall be simple in construction, so the more code we find, the faster we obtain an initial result. We list and comment on useful frameworks to construct generative models.