ISyE8900 Project

Chi Zhang

Georgia Institute of Technology

I. Introduction

In this project, we implement a nonparametric approach for modeling interest rate curves using various manifold learning methods. The main focus will be the two spread curves: Swap Spread (LIBOR - Treasury) and Basis Spread (LIBOR - SOFR).

II. Literature Review

While term structure modeling has been well developed in past decades in the Treasury markets, advances in modeling the swap term structure or the spread term structure are comparatively small. Besides, for the purposes of managing risk or hedging derivatives, it is very common to see structural changes in local term structure areas, we would like to use adaptive time series forecasting techniques within the curve modeling framework. Diebold and Li (2006) use the AR family models to obtain encouraging results for long-horizon ex-ante forecasts by reformulating the Nelson and Siegel (1987) model. Duffee (2002) argues the random walk model is superior to the previous affine term structure models. Apart from these general parametric models, economists have also tried add some exogenous macroeconomic variables (e.g. real activity, inflation, and fed funds rate) to improve the predicting power, such as Ang and Piazzesi (2003) and Diebold, Rudebusch and Aruoba (2006). Though a large part of term structure model specification has been deployed, a uniform conclusion with regard to the factor selection has not been achieved yet.

As the booming of modern machine learning techniques, some elementary dimension reduction techniques like PCA have been applied in the fixed income market, such as Steeley (1990) and Litterman and Scheinkman (1991). Practitioners usually interpret the principle components as level, slope, and curvature effects. To link the factors with more understandable economic instances in the financial markets, Duffie and Singleton (1997) propose a multi-factor model for IRS (interest rate swap) that accommodates counterparty default risk and liquidity differences between the Treasury and Swap markets. By extending this work, Liu, Longstaff and Mandell (2006) estimate a five factor model to analyze swap spreads.

Since the movements of interest rates have been shown depend on the absolute level of rates: the curve behavior at low is different from the behavior when rates are high. This is exactly the limit of PCA: encoding a great amount of information regarding curve shapes into a covariance matrix. To better capture the non-linear relationship among neighbor tenors, Kondratyev (2018) (https://github.com/DarseZ/CurveFrcst-Using-ManifoldLrn/blob/main/papers/ANNLearnCurveDynamics.pdf) proposes a neural networks based term structure algorithm for brent oil forward price and USD swap curve.

To the best of our current knowledge, there is no work that aimed to forecast the spread term structure using non-linear dimension reduction, i.e. manifold learning. Inspired by the work of Kondratyev, we believe the heterogeneity of temporal evolution will be better captured by manifold learning compared with the widely deployed linear method. For the time series forecasting, we take the state space model (Kalman filter) as our main approach in estimating time series models. If time permits, we would like to extend the AR family model to a GARCH family model (e.g. E-GARCH) which could yield a feasible if not optimal model specification to capture the volatility pattern of the low dimensional representations.

III. Problem formulation and Application

A. Curves Building and Interest Rate Processes Simulation

From the website of Fed Reserve Bank of New York (https://www.newyorkfed.org/), we get the functional parameters of the Nelson-Siegel-Svensson structural models for Treasury instantaneous forward rates as described in Svensson (1994). Then, using the calibrated parameters as shown in equation (1) and (2), we could construct the zero (forward) rate curve by plugging the maturity index into the zero rate expression.

$$y_i(n) = \beta_0 + \beta_1 \frac{1-\exp(-\frac{n}{\tau_1})}{\frac{n}{\tau_1}} + \beta_2 \left[\frac{1-\exp\left(-\frac{n}{\tau_1}\right)}{\frac{n}{\tau_1}} - \exp\left(-\frac{n}{\tau_1}\right)\right] + \beta_3 \left[\frac{1-\exp\left(-\frac{n}{\tau_2}\right)}{\frac{n}{\tau_2}} - \exp\left(-\frac{n}{\tau_2}\right)\right], \qquad (1)$$

$$f_t(n,0) = \beta_0 + \beta_1 \exp(-\frac{n}{\tau_1}) + \beta_2 \frac{n}{\tau_1} \exp(-\frac{n}{\tau_1}) + \beta_3 \frac{n}{\tau_2} \exp(-\frac{n}{\tau_2}), \qquad (2)$$

The data format should be daily data with continuous maturity spectrum from 1m to 30y (360 dimensions). To make it consistent with the following two swap curves, we re-sample it to get a weekly dataset with discrete maturity spectrum from 3m to 30y (120 dimensions).

For the LIBOR and SOFR curves, we directly download them from Bloomberg terminal by manually changing "As of Date" variable. If time permits, we will replicate the whole curve bootstrapping pipeline to get these curves using market instruments prices (deposits, futures/forwards, swaps).

For the preparation of accuracy and trading strategies backtesing, we take a parametric model, the Ornstein-Uhlenbeck process (equation (3) and (4)), to estimate the parameters of underlying interest rates process for Treasury, LIBOR, and SOFR respectively. With the calibrated process, we simulate a large number of realizations (e.g. 10,000).

$$\mathrm{d}S_t = \lambda(\mu - S_t)dt + \sigma dW_t, \qquad (3)$$

$$S_{(t+\delta)} = S_t e^{-\lambda\delta} + \mu(1 - e^{-\lambda\delta}) + \sigma\sqrt{\frac{1-e^{-\lambda\delta}}{2\lambda}}\,\mathrm{N}(0, 1), \qquad (4)$$

For each simulated scenario, we will obtain two panel data for the swap spread and basis spread respectively. Each of them will be in the shape of N by M, where N is the number of weeks along the calendar dates and M is the number of tenors.

The data preparation could be seen in https://github.com/DarseZ/CurveFrcst-Using-ManifoldLrn/blob/main/CurveBuild.ipynb

We would keep add most recent data points (dates) into the dataset as the project moved forward.

B. Nonlinear Dimension Reduction

Starting from baseline classical methods, PCA, which is a popular technique for explaining curve dynamics, but there are two underlying strong assumptions which may limit the predictive power for datasets with strong nonlinearity: (1) The new orthogonal basis, which is a linear combination of the original basis, usually is not able to capture the most interesting part of nonlinear behavior; (2) Mean and variance are sufficient statistics. Therefore, if the probability distribution is not Gaussian, all other necessary high-order statistics will be lost.

The second approach is Locally Linear Embedding, which is a classical manifold searching method. Given a set of D-dimensional data points $x_1, x_2, \ldots, x_n$, we try to find the embedded d-dimensional feature vectors $y_1, y_2, \ldots, y_n$. The main steps are as following.

(1) Identify nearest neighbors based on some distance metric for each data point x_i, where N_i denote the set of indices of the nearest neighbors for this data point.
(2) Find the optimal local convex combination of the nearest neighbors to represent each data point. That is, we are optimizing function (5) to compute the weights.

$$\mathrm{E}(\mathrm{w}) = \sum_{i=1}^{N} \left\| x_i - \sum_{j \in N_i} W_{ij} x_j \right\|^2 \text{ , s.t. } \sum_{j \in N_i} W_{ij} = 1 \qquad (5)$$

(3) Find the low-dimensional feature vectors y_i, which have the optimal local convex representations with the given W_{ij}. That is, we try to compute y_i by minimizing the following cost function (6).

$$\phi(y) = \sum_{i=1}^{N} \left\| y_i - \sum_{j \in N_i} W_{ij} y_j \right\|^2 \qquad (6)$$

For the third and fourth dimension reduction approach, we would like to implement Multidimensional Scaling (MDS) and t-distributed Stochastic Neighbor Embedding (t-SNE), which belong to the category of semi-classical methods.

Compared to the direct eigen-analysis of the N data points themselves in PCA, MDS selects influential dimensions by the eigen-analysis of the N^2 data points of a pairwise distance matrix. The goal is to preserve the pairwise distances as best as possible after mapping to the low-dimensional space.

t-SNE converts similarities between data points to joint probabilities and tries to minimize the K-L divergence between the joint probabilities of the low-dimensional embedding and the high-dimensional data.

The dimension reduction part could be seen in https://github.com/DarseZ/CurveFrcst-Using-ManifoldLrn/blob/main/DmnsRdct_StateFrcst.ipynb

As we try to model non-linear relationship among different tenors, the low-dim representations may not be that interpretable as the parametric framework or the baseline PCA approach. This is the cost of improving prediction power.

C. Time Series Forecasting and Inverse Transform

Our objective is to predict the most likely curve transformation given its observed shape at a particular moment in time. Our method converts original spread curves into several main drivers in low dimensional space by manifold learning. After conversion, we employ both the AR family models and GARCH family models with added Kalman filter to attain more adaptability.

If time permits, we will try a perturbation test (add a specific tenor into the input of the neutral net), the visualize how the entire curve will be impacted by the exogenous impulse after time interval Δt. For the counterparty in dimension reduction based approaches (e.g. PCA, MDS, LLE, t-SNE), the procedure is as follows: (1) estimate reduced representations and loadings for each tenor using original curve, (2) re-estimate the reduced representations using perturbed curve, (3) compute the inverse transform using original loadings and new reduced representations.

Because the (inverse) manifold learning algorithm is not an injective function, we may cannot come up with a general approach. Three potential solutions: (1) nonparametric regression, by Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimension reduction via tangent space alignment,"; (2) inverse manifold learning (encoding and decoding) by the scholars in Xihu University (2020). (3) simple LLE reconstruction, by Jin Chen, Shijie Deng, and Xiaoming Huo (2004).

By taking the third option, suppose low-dimensional feature vectors $y_1, y_2, \dots, y_n$ have been obtained through LLE in the previous subsection and we have a new prediction y_{n+1}, we could reconstruct x_{n+1} using the following steps.

(1) Identify nearest neighbors based on some distance metric for each data point y_{n+1+}, where N_{n+1} denote the set of indices of the nearest neighbors for this data point.
(2) The weights of the local optimal convex combination w_j are obtained by optimizing function (7).

$$E(w) = \left\| y_{n+1} - \sum_{j \in N_{n+1}} W_j x_j \right\|^2 \text{ , s.t. } \sum_{j \in N_{n+1}} W_j = 1 \qquad (7)$$

(3) The data point in high-dimensional space should be $\widehat{x_{n+1}} = \sum_{j \in N_{n+1}} W_j X_j$.

D. Performance Evaluation and Systematic Trading Strategies

We define three widely used statistical measure and an economic measure to evaluate the forecasting performance. For the statistical measure, regarding the relative difference of spread values between two adjacent time steps could be classified into non-negative (positive) and negative categories, we calculate the precision rate ($\frac{true\ positive}{true\ positive+false\ positive}$), recall rate ($\frac{true\ positive}{true\ positive+false\ negative}$), and F1 score ($\frac{2*precision*recall}{precision+recall}$).

For the economic measure, we backtest three systematic curve trading strategies based on the model forecasting.

The level trading signal will be $0.33 * 2\text{yr} + 0.33 * 5\text{yr} + 0.33 * 10\text{yr}$:

If we predict the signal is going to increase at next period (average level increases), then float leg payment will increase, we will build a float leg receiver position.

The fly trading signal will be $-0.5 * 2\text{yr} + 0.5 * 10\text{yr}$:

If we predict the signal is going to increase at next period (curve slope increases), then float leg payment of 10yr will increase relative to the 2yr, we will build a float leg receiver position of 10yr and a float leg payer position of 2yr.

The butterfly trading signal will be $0.25 * 2\text{yr} - 0.5 * 5\text{yt} + 0.25 * 10\text{yr}$:

If we predict the signal is going to increase at next period (curve curvature increases), then float leg payment of 10yr and 2yr will increase relative to the 5yr, we will build a float leg receiver position of 10yr and 2yr and a float leg payer position of 5yr.

By holding the corresponding swap portfolio for one-period (a week) suggested by the trading signal, we will get three cumulative PnL plot for each given model specification. This will help us identify the difference between different algorithms in predicting specific patterns of curve dynamics.

IV. Results

V. Conclusion

References

[1] Chen J, Deng S J, Huo X. Electricity price curve modeling and forecasting by manifold learning[J]. IEEE Transactions on Power Systems, 2008, 23(3): 877-888.

[2] Huo X, Ni X S, Smith A K. A survey of manifold-based learning methods[J]. Recent advances in data mining of enterprise data, 2007: 691-745.

[3] Blaskowitz O J. A forecast evaluation of PCA-based adaptive forecasting schemes for the EURIBOR swap term structure[D]. Christian-Albrechts Universität Kiel, 2009.

[4] Sack B P, Wright J, Gürkaynak R. The US Treasury yield curve: 1961 to the present[R]. Board of Governors of the Federal Reserve System (US), 2006.

[5] Kondratyev A. Learning curve dynamics with artificial neural networks[J]. Available at SSRN 3041232, 2018.