ICA, Gaussian Process, Fama-French 5 factor model, HMMs and jump models

Kaiwen Zhou

Contents

1 Topic: Random Feature Maps

Given a set of functions of a single variable $\{\phi_i(t)\}_{i=1}^M$ and discrete observations $(t_j)_{j=1}^N$ one can form both the $N \times M$ design matrix $\mathbf{\Phi} = [\Phi_{ji}] := [\phi_i(t_j)]$ and the $N \times N$ kernel $\mathbf{K} = [K(t_j, t_l)] := \mathbf{\Phi}\mathbf{\Phi}^\top$. As we will see in later lectures, often $M \gg N$ and in fact in most cases $M = \infty$ (e.g. Fourier series). In this exercise we will consider random feature map functions.

$$\mathbf{\Phi} := \begin{bmatrix} \phi_1(t_1) & \cdots & \phi_M(t_1) \\ \vdots & \ddots & \vdots \\ \phi_1(t_N) & \cdots & \phi_M(t_N) \end{bmatrix} \in \mathbb{R}^{N\times M}$$

(a) Suppose $\phi_i(t_j)$ are simply white noise vectors, e.g. each entry in the design matrix is an i.i.d. realization of random mean zero Gaussian with variance 1. In this case one can think of each feature map $\phi_{\mathbf{w}_i} := (\phi_i(t_1), \ldots, \phi_i(t_N)) \equiv \mathbf{w}_i$ as a random constant vector function taking a value $\mathbf{w}_i$. Because one is drawing M such random vectors, the feature maps are M random vector-valued constant functions. Construct the kernel matrix and find its eigenvectors and eigenvalues. For fixed $N = 100$, plot the histogram of the eigenvalues of $\mathbf{K}$ as a function of $q = M/N$ for $M = 10, 100, 1000$. Plot the top 3 eigenvectors in each case.

(b) Repeat (a), but this time assume $\phi_i(t_j)$ is a random draw of discretized Brownian motion paths evaluated at $t_j = (j/N)_{j=0}^N \in [0, 1)$. Each feature map is therefore the cumulative sum of Brownian increments, where the Gaussian has zero mean and a variance of $1/N$. Do you see any structure in the top three eigenvectors of the kernel matrix resulting from this Brownian motion design matrix? Empirically, how fast does the spectrum of the n-th eigenvalue decay as a function of n for each of the values of q in (a)?

(c) Investigate the stability of your results in (a) and (b) to resampling of your random feature maps. Which aspects appear stable and which aspects do not? Explain your findings.

Solution:

2 Topic: Brownian Motion and Brownian Bridge Gaussian Processes (GPs)

In class we suggested that both the Brownian motion

$$y_t = \int_0^t dW_t \tag{1}$$

and the Brownian bridge stochastic process

$$y_t = \int_0^t dW_t, \quad \text{where } t \le 1 \text{ and } y(1) = 0 \tag{2}$$

can be thought of as GPs.

(a) Prove that the kernel of Equation (1) is $K_{\mathrm{BM}}(t, t') = \min(t, t')$ and that the kernel of Equation (2) is $K_{\mathrm{BB}}(t, t') = \min(t, t') - tt'$.

Solution: WLOG, suppose $t' \le t$.

The Kernel of a Brownian Motion: By the assumption above, $W(t')$ and $W(t) - W(t')$ are independent, we have

$$\begin{aligned} K_{\mathrm{BM}}(t, t') &= \mathbb{E}[W(t')W(t)] \\ &= \mathbb{E}[W(t')(W(t) - W(t')) + W(t')^2] \\ &= \mathbb{E}[W(t')] \cdot \mathbb{E}[W(t) - W(t')] + \mathbb{E}[W(t')^2] \\ &= 0 + Var[W(t')] \\ &= t' \\ \text{by symmetry} \implies &= \min(t', t). \end{aligned}$$

The Kernel of a Brownian Bridge: A Brownian bridge $X(t)$ from 0 to 0 over $[0, T]$ can be expressed as:

$$X(t) = W(t) - \frac{t}{T} \cdot W(T), \quad 0 \le t \le T.$$

where $W(t)$ is a Brownian motion. For this question in particular, setting $T = 1$, we have $X(t) = W(t) - t \cdot W(1)$, $0 \leq t \leq 1$.

By the definition of a kernel, we have

$$\begin{aligned} K_{\mathrm{BB}}(t, t') &= \mathbb{E}[(W(t) - t \cdot W(1))(W(t') - t' \cdot W(1))] \\ &= \mathbb{E}[W(t)W(t') - tW(1)W(t') - t'W(1)W(t) + tt'W(1)W(1)] \\ &= \min(t', t) - t\min(t', 1) - t'\min(1, t) + tt' \cdot \min(1, 1) \\ 0 \leq t' \leq t \leq 1 \implies &= \min(t', t) - tt' \end{aligned}$$

(b) Implement the kernels Equation (1) and Equation (2) in Scikit-Learn (e.g. see the kernels they currently have implemented and inherit from the kernel class) and show that paths generated using Equation (1) have properties of Brownian motion paths, while those generated using Equation (2) have properties of a Brownian bridge. Plot 100 paths generated from the priors of each model.

Solution: See the attached Jupyter notebook.

(c) Brownian bridge can be thought of as Brownian motion conditioned to pass through $y(1) = 0$. Within the GP framework, one should therefore be able to simulate Brownian bridge paths using the posterior density corresponding to the kernel (1), conditioned on a single training point $(t, y) = (1, 0)$. Using the GP formula for the posterior density as a function of the kernel in Equation (1), show that for a single training point $(t, y) = (1, 0)$, it corresponds to sampling from the distribution $\mathcal{N}(0, K_{\mathrm{BB}}(t, t'))$. Plot 100 paths of the posterior density of Brownian Motion trained on $(t, y) = (1, 0)$.

Solution: See the attached Jupyter notebook.

3 Topic: Applying HMMs and Jump Models to Equity Factors

The deliverables for this homework are:

1. one Python notebook (that will read in all the data needed and then does all necessary calculations),
2. all data files you use. Make sure to comment and annotate your steps in the notebook clearly.

Download the daily Fama-French 5-factor model (FF5M). A description of the factors is available here. You will be using Mkt-RF, SMB, HML, RMW, and CMA for the time period 2010/01/04 through 2023/09/29.

Create one notebook with all your code, analysis, and answers to the following questions:

(a) Briefly describe how the 5 time series from FF5M have been constructed.

(b) Produce a table of average return, volatility, and correlations of the FF5M time series. Returns and volatilities should be annualized.

(c) Compute and plot cumulative daily returns and volatilities for the 5 time series as in the top half of Exhibit 10 in "Greedy Online Classification of Persistent Market States Using Realized Intraday Volatility Features" by Nystrup, Kolm, and Lindström covered in one of the lectures. The result for Mkt-RF will be similar to the S&P500 used in the exhibit.

(d) For each time series, compare the hidden states inferred from hmmlearn (fits an HMM) and the jump model. For the purposes of this analysis, you can assume there are two hidden states. While the results from hmmlearn only depend on the initialization, the results for the jump model depend on your initialization and the penalty parameter, λ. Therefore, for the jump model, you will need to experiment with different values for λ. To address the issue that the results depend on the initialization, for each model do the inference with 5 different initializations and pick the initialization that gives the best result.

(e) For each of the 5 factors of the FF5M, write out your findings. In particular:
   - For each factor, how do the state sequences from the HMM and jump model compare?
   - How do the state sequences compare across factors? (Hint: Develop a way to plot/visualize the state sequences over time for easy comparison.)

(f) (Extra credit) Based on your results above, propose and test a trading strategy for each of the factors. Daily, each strategy should decide whether to go long one unit of the factor or not to hold it (i.e., hold zero units of the factor).

Important:

- Make sure each strategy is deployed out-of-sample. In other words, you can only use information available to you before the day you make each trade decision.
- For each factor, use the long-only strategy as a benchmark. Compare the performance of your trading strategy in each of the factors with the long-only strategy in the factor. Does your strategy outperform? Compute its annualized average return, annualized volatility, annualized Sharpe ratio, and annualized α and β relative to the benchmark strategy.