AALTO UNIVERSITY

RESEARCH PROJECT REPORT

Applying Gaussian Processes to High Frequency Trading

Chengyu HAO, Yaowei LI - Data Science(EIT digital), School of computer science, Aalto University
Research Project

Supervisors:

PAUL CHANG, ARNO SOLIN - School of Computer Science, Aalto University

Contents

1 Introduction

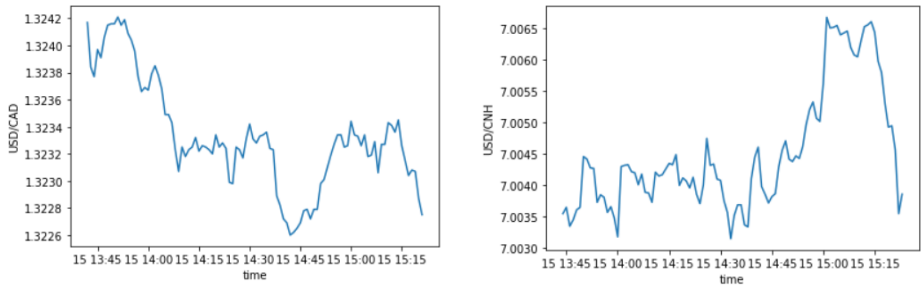
High Frequency trading (HFT) is applied as a platform for matching buyers and sellers in financial asset markets. The transactional data captured can be thought of as an unbounded time series. Recent advances in theory allow for Gaussian Processes to predict online and inference for otherwise computationally infeasible applications. However, the time complexity of fitting the Gaussian process is cubic time complexity, which is impossible to be implement in the high frequency trading. In order to fit the HFT data in linear time, we need to identify a new method that can fit the data more efficiently.

The Kalman filter algorithm applies the concept of state space, based on time domain design, is a highly efficient recursive filter (autoregressive filter) that can be applied from a series of incomplete and noise-containing measurements. It estimates the state of the dynamic system and reduces the algorithm time and space complexity, and is widely used in various signal estimation.

In this paper, we show how temporal Gaussian process regression models in machine learning can be reformulated as linear-Gaussian state space models, which can be solved exactly with classical Kalman filtering theory. The result is an efficient non-parametric learning algorithm, whose computational complexity grows linearly with respect to number of observations. We use this method to online predict the HFT process. Specifically, the method in this report involves extracting information representations of HFT data that optimally represents market dynamics. Further, the data are modelled in a probabilistic framework using ideas around automatic pattern discovery kernels and the representation of Gaussian processes as state space models to allow for linear time complexity.

2 Data Description

Figure 2 shows the relationship between time and exchange rate for different currencies.



(a) USD/CAD (b) USD/CHN

(c) EUR/USD (d) USD/JPY

Figure 2.1: the relationship between time and exchange rate for different currencies with 1-min frequency

3 Kernel Method

In this section, we try different kernels to fit the data. Several kernels are considered below:

- Polynomial kernel. The Polynomial kernel is defined as:

$$k(x,y) = (x^T * y + c)^d$$

- Periodic kernel. The Periodic kernel is defined as:

$$k(x,y) = \theta_1 \exp\left[-\frac{1}{2}\sum_{i=1}^{input\_dim}\left(\frac{\sin(\frac{\pi}{T_i}(x_i - y_i))}{l_i}\right)^2\right]$$

- RBF kerne:

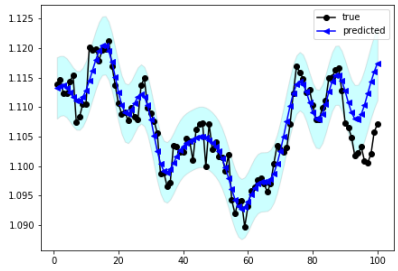
$$k(r) = \sigma^2 \exp\left(-\frac{1}{2}r^2\right)$$

- linear kernel:

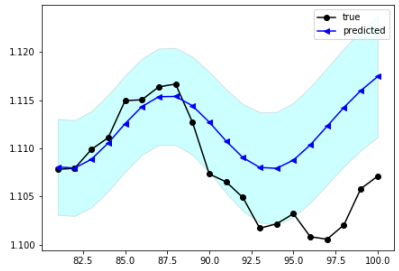
$$k(x,y) = \sum_{i=1}^{input\_dim} \sigma_i^2 x_i y_i$$

3.1 Combine different kernel methods

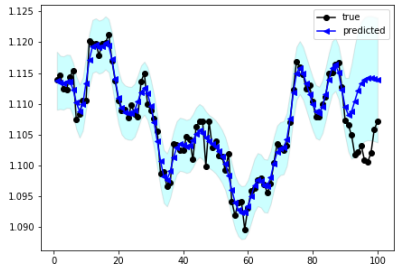
In this section, we combined different kernels with period kernel to further investigate the fitting and prediction effect. We use the likelihood as to reflect the fitting effect and the RMSE to reflect the prediction performance. For the RMSE, assume the time period of training data is t, we evaluate the RMSE of $t+1$ and $t+1$ to $t+10$ to see the prediction performance in short and long time.
Here is the description of our experiment. We fixed the currency as USD-EUR, and choose 4 kinds of time frequency (m1, m5, m30, h4, d1) to evaluate the model performance under different time frequency. For each time frequency we download 500 data points in total. Then we divide the 500 data points into 5 samples, in each sample we take first 90 points as training data and the last 10 points are validation data. Fig. 3.1 shows the fitting and prediction result for different kernels with time frequency as d1.



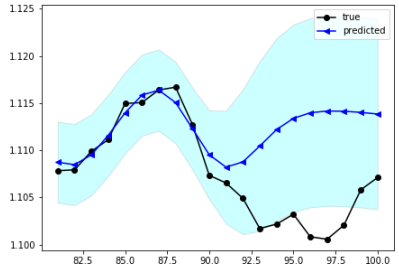
(a) fitting of polynomial and periodic kernel



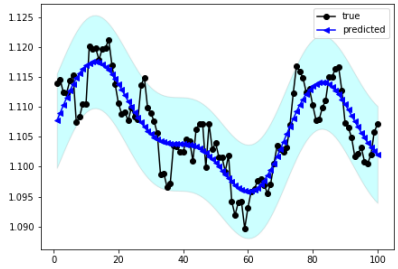
(b) prediction of polynomial and periodic kernel



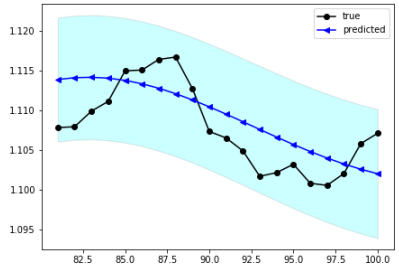
(c) fitting of RBF and periodic kernel



(d) prediction of RBF and periodic kernel



(e) fitting of linear and periodic kernel



(f) prediction of Linear and periodic kernel

Figure 3.1: 3.1(a),3.1(c) and 3.1(e) reflect the fitting function on the trade data with frequency of one day(both training and prediction), the right side plot (3.1(b), 3.1(d) and 3.1(f)) are the prediction of the function

Here we show the normalized RMSE with different kernels and time frequen-

cies. We calculate 4 time frequencies (m1, m5, m30, h4) and 3 kind of kernels. We define k1 as Periodic kernel, k2 as RBF kernel, k3 as Linear kernel, and k4 as Polynomial kernel. Our result shows the kernels of combination of K1+K2, K1+K3, K1+K4.

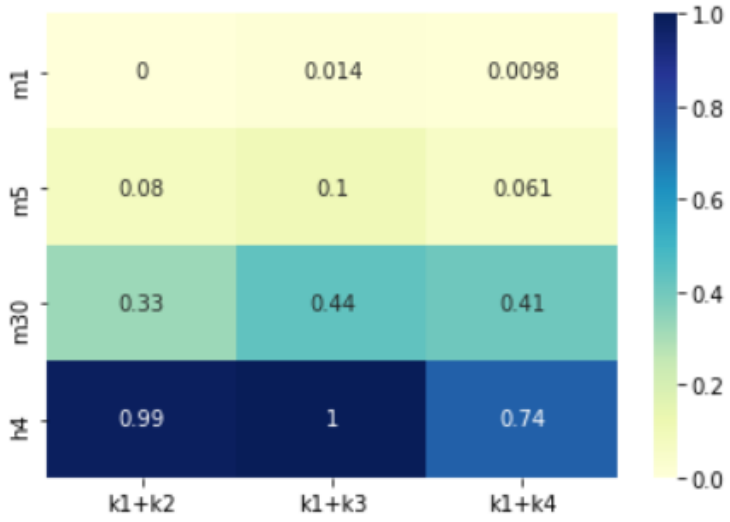


Figure 3.2: Normalized RMSE for time period from t+1 to t+10

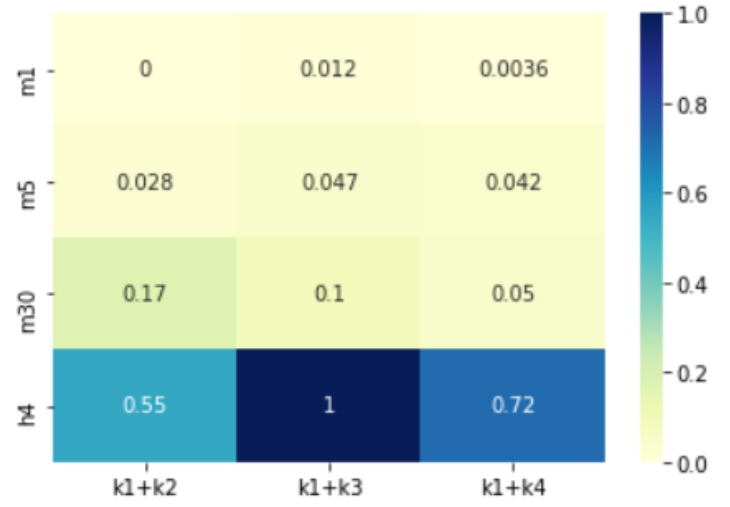


Figure 3.3: Normalized RMSE for time period of t+1

Here we show some specific results in table 1 and table 2.

kernel method	RMSE of T+1	RMSE of T+1-T+10	Log likelihood
Polynomial+Periodic	0.01066	0.02315	359.8
RBF	0.00606	0.01791	337.401
RBF+Periodic	0.00167	0.00955	386.3816
linear kernel and periodic kernel	0.00746	0.01237	354.37796

Table 1: Evaluation on frequency of D1

kernel method	RMSE of T+1	RMSE of T+1-T+10	Log likelihood
Polynomial+Periodic	6.48e-05	4.94e-05	709.818
RBF	6.01e-05	7.77e-05	701.653
RBF+Periodic	6.56e-05	9.42e-05	713.922
linear kernel and periodic kernel	0.00015	0.00015	708.085

Table 2: Evaluation on frequency of m1