Point Process Homework2

Description

Equipments maintenance is a one of the most important application of point process model. This homework aims to improve the understanding of the point process model by using hawkes process to solve practical problems,which derived from malfunction prediction of automated teller machines (ATMs) .

Background

Predictive Maintenance Problem

This project is derived from Predictive Maintenance Problem. It involves equipment risk prediction to allow for proactive scheduling of corrective maintenance. Such an early identification of potential concerns helps deploy limited resources more efficiently and cost effectively, reduce operations costs and maximize equipment uptime. Predictive maintenance is adopted in a wide variety of applications such as fire inspection, data center and electrical grid management. For its practical importance in different scenarios and relative rich event data for modeling, this homework is modeling a real-world dataset of more than 1,000 automated teller machines (ATMs) from a global bank headquartered in North America.

Dataset

In maintenance support services, when a device fails, the equipment owner raises a maintenance service ticket and technician will be assigned to repair the failure. The studied dataset is comprised of the event logs involving error reporting and failure tickets, which is originally collected from 1,554 ATMs. The event log of error records includes device identity, timestamp, message content, priority, code, and action. A ticket (TIKT) means that maintenance will be conducted. A ticket (TIKT) means that maintenance will be conducted. Statistics of the data is presented in Table-1. The error type indicates which component encounters an error:

- printer (PRT)
- cash dispenser module (CNG)
- Internet data center (IDC)
- communication part (COMM)

- printer monitor (LMTP)
- miscellaneous e.g., hip card module, usb (MISC)

The training data consists of 1085 ATMs and testing data has 469 ATMs, in total Wincor ATMs that cover 5 ATM machine models: ProCash 2100 RL (980, 430), 1500 RL (19, 5), 2100 FL (53, 21), 1500 FL (26, 10), and 2250XE RL (7, 3). The numbers in the bracket indicate the number of machines for training and testing.

In dataset, event types have been numerically, and the specific correspondence is shown in the Table-2.

Table 1 ATM data set different types of event statistics table

Type	total
TIKT	3516
PRT	99252
CNG	80921
IDC	15074
COMM	165567
LMTP	49094
MISC	138791

Table 2 ATM event type correspondence relation

Type	Number
TIKT	6
PRT	0
CNG	1
IDC	2
COMM	3
LMTP	4
MISC	5

Dataset contains two data files：atm_train.csv and atm_test.csv

In data files, the following data is provided:

- id: ATM code
- time: timestamp in day
- event: event type

Logs of each ATM can be regraded as one event sequence.

Multi-dimensional Hawkes Process

Multi-dimensional hawkes process is a extention of one-dimensional hawkes process. In addition to considering the event incentives in each dimension, the multi-dimensional Hawkes process also considers the impact of incentives between events in different dimensions. Specifically, we have Z Hawkes processes that are coupled with each other: each of the Hawkes processes corresponds to an individual and the influence between individuals are explicitly modeled. Formally, the conditional intensity for the d-th dimension expressed as follows:

$$\lambda_d(t) = \mu_d + \sum_{i:t_i<t} a_{d_i d} \exp\big(-w(t - t_i)\big)$$

The coefficient $a_{d_i d} \geq 0$ captures the mutually-exciting property from the d_i-th to d-th dimension. We collect the parameters into matrix-vector forms $\boldsymbol{\mu} = (\mu_d)$ for the base intensity, and $\boldsymbol{A} = (a_{d_i d})$ for the mutuallyexciting coefficients, called infectivity matrix.

ADMM-MM Algorithm

ADMM-MM algorithm is a special algorithm for multi-dimensional hawkes model learning with sparse and low-rank regularization. This algorithm combines techniques of alternating direction method of multipliers and majorization minimization.

Algorithm 1 ADMM-MM (ADM4) for estimating $\mathbf{A}$ and $\boldsymbol{\mu}$

Input: Observed samples $\{c_1, \ldots, c_m\}$.
Output: $\mathbf{A}$ and $\boldsymbol{\mu}$.
Initialize $\boldsymbol{\mu}$ and $\mathbf{A}$ randomly; Set $\mathbf{U}_1 = 0$, $\mathbf{U}_2 = 0$.
while $k = 1, 2, \ldots,$ **do**
Update $\mathbf{A}^{k+1}$ and $\boldsymbol{\mu}^{k+1}$ by optimizing Q defined in (8) as follows:
while not converge **do**
Update $\mathbf{A}$, $\boldsymbol{\mu}$ using (10) and (9) respectively.
end while
Update $\mathbf{Z}_1^{k+1}$ using (6); Update $\mathbf{Z}_2^{k+1}$ using (7).
Update $\mathbf{U}_1^{k+1} = \mathbf{U}_1^k + (\mathbf{A}^{k+1} - \mathbf{Z}_1^{k+1})$ and $\mathbf{U}_2^{k+1} = \mathbf{U}_2^k + (\mathbf{A}^{k+1} - \mathbf{Z}_2^{k+1})$.
end while
return $\mathbf{A}$ and $\boldsymbol{\mu}$.

Task

You should implement hawkes model with with sparse and low-rank regularization, and fitting ATM dataset with ADMM-MM algorithm.

In report, you should describe the software language, key libraries, and other things necessary to run the program. Then you should explain what kind of problems are encountered and how to solve them. As for the results of model, you should you should display the actual results of the program and analyze them.

Metrics

- Event Type Prediction：Precision, Recall, macro-F1 Score over 7 event types under. Note all these metrics are computed for each type, and then averaged over all types.
- Event Time Prediction: Mean Absolute Error (MAE) which measures the absolute difference between the predicted time point and the actual one.

Submission

You should submit a .zip file with all source code and the report.

Grade

Source code(50%): model results, code norm, program running

Report(50%): well-organized and analytical

Note: new creative exploration can be regarded as a bonus.

Reference

1. Zhou K, Zha H, Song L. Learning social infectivity in sparse low-rank networks using multi-dimensional hawkes processes[C]//Artificial Intelligence and Statistics. 2013: 641-649.

Academic misconduct and handling

We will review all source code and analyze the plagiarism of source code and reports, therefore, don't try to test its reliability. For self-completed student, there will be reward points. For plagiarized student, it will be awarded zero points. If you are convicted of plagiarism, you can ask TAs to make a defense.