My Data Science Challenging Problems # 53: Streaming Data Analytics for M- and Z- estimators

Drafted by Haoda Fu
XXX
XXXX
xxx

November 28, 2018

Abstract

If you happen to have a solution of this question, please contact Haoda Fu at fu_haoda@lilly.com or fuhaoda@gmail.com or commit your solution to `git@github.com:fuhaoda/DCSP53_StreamingDataAnalytics.git`

Key words and Phrases: *** ***.
Short title: ***

1 Introduction

As an example, if we are interested in to calculate the mean value of $X_1, \cdots, X_{100}$. The traditional approach is to save all the 100 numbers then take the average. For streaming data analytics, we only need to save the $\bar{X}_n$ and n, then when the X_{n+1} comes in, we can update the results as $\bar{X}_{n+1} = \frac{n\bar{X}_n + X_{n+1}}{n+1}$.

Streaming data analytic is an important area for digital health. Many of the analytic problems are end up as M- or Z- estimators. How can we build up streaming data analytics for M- and Z- estimators?

2 Definition

Suppose that we are interested in a parameter (or functional) θ attached to the distribution of observations $X_1, \cdots, X_n$. A popular method for finding an estimator $\widehat{\theta}_n =$

$\widehat{\theta}_n(X_1, \cdots, X_n)$ is to maximize a criterion function of the type,

$$\theta \mapsto M_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} m_\theta(X_i).$$

Here $m_\theta : \mathcal{X} \mapsto \mathbb{R}$ are known functions. An estimator maximizing $M_n(\theta)$ over Θ is called an *M-estimator*.

As related, some of problems are defined to find solution of,

$$Z_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} Z_\theta(X_i) = 0,$$

where Z_θ are known vector-valued maps. Such estimator is called *Z-estimator* (for zero).

Suppose the data $X_1, X_2, X_3, \cdots$ are coming in sequence. How can we continue to update our estimator of θ. So a few related questions,

1. We cannot save all the X_i, what information do we need to save so that we can have asymptotic distribution of θ correctly.

2. What are the efficient updating algorithms for the mean and variance estimation?

3. How can we design `C++` codes so that it can be broadly used.

The solution could be related stochastic gradient decent approach. We also need some efficient solution, such as modified limited memory BFGS algorithms.

3 Extension

We can also add a weigh to each of X_i or its corresponding functions, such as

$$\theta \mapsto M_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} w_i m_\theta(X_i).$$

and

$$Z_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} w_i Z_\theta(X_i) = 0,$$

with $\sum_{i=1}^{n} w_i = 1$ and $w_i \geq 0, \forall i$. Through weighting, we can put more weight on the recent X_i and less weight on the X_i far way from current.

Appendix: Proof of Equation ()

.

References