大规模数据 Logistic 回归

李丰[1]

2014 年五月 22 日 • 腾讯

Rev: July 16, 2015

[1]中央财经大学统计与数学学院

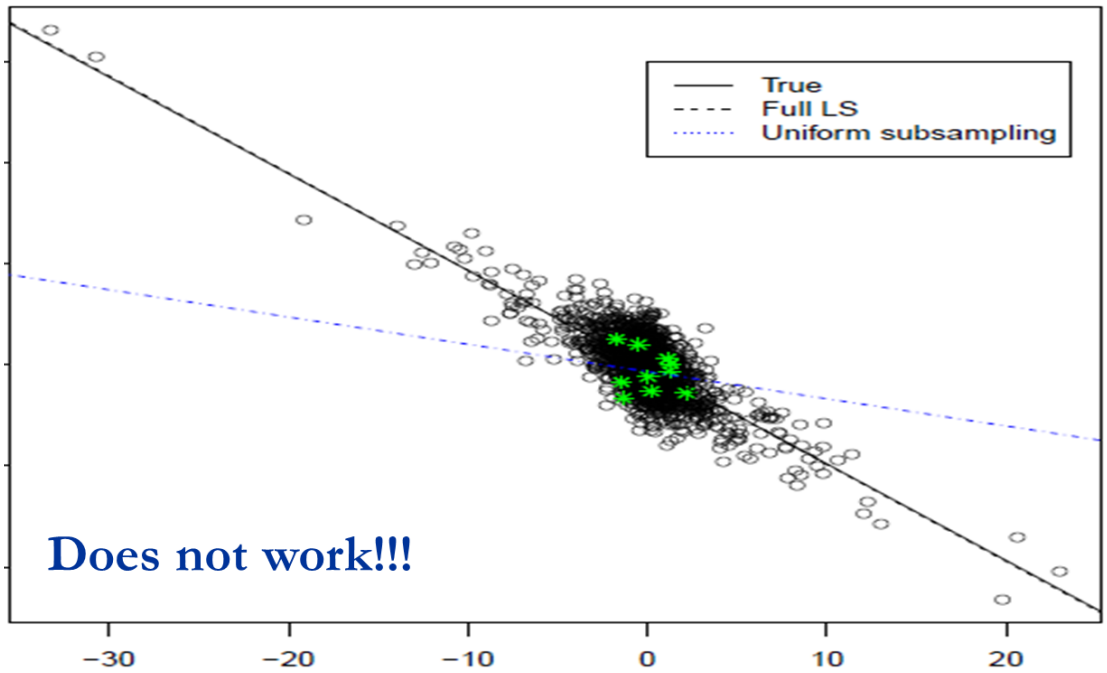
1. 大数据下复杂模型面临挑战
2. 基于 GPU 的分块 Logistic 回归模型
3. 扩展

背景

- 因为大数据：
  - 所以 Hadoop/Mapreduce, Spark,...
  - 所以数据更复杂，要求复杂模型，但是复杂模型在大数据平台实现并不容易。
- 传统的统计学家/数据科学家：
  - 知道许多复杂模型
  - 但是只能在个人电脑（小于 32G 内存）上操作。
- 现有工具
  - 一个中等偏上的计算机基本配置（约 1 万 RMB）：32G 内存，独立显卡，1-2T 硬盘。
  - R/Python 等语言提供了主流统计模型的 in-memory 的基本实现，但是慢！
  - 底层语言 C/C++ 有对数据读写的 memory-effcient 方法，但是不适合直接拿来作复杂模型的统计计算。

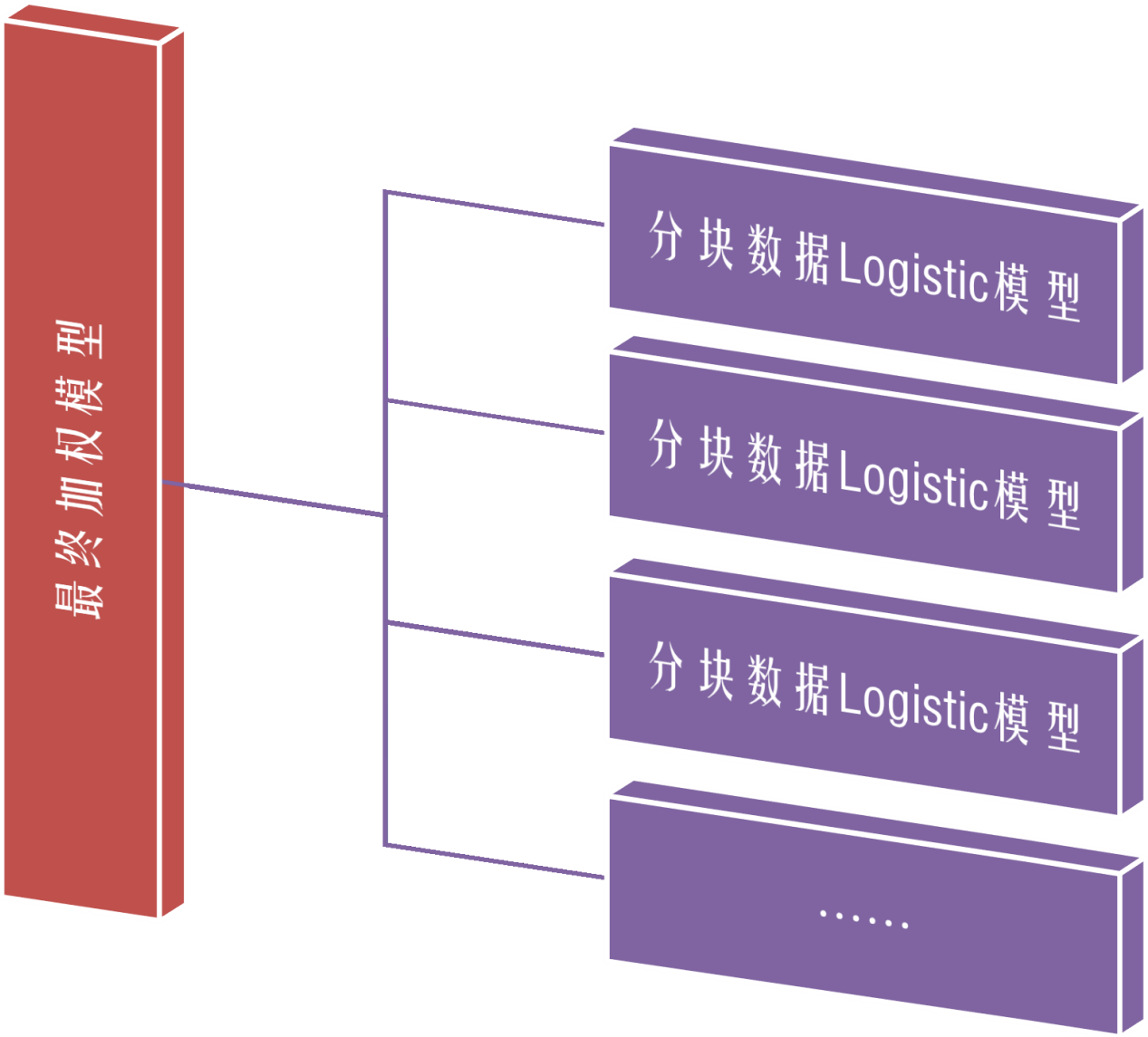
大数据下复杂模型面临挑战

- 复杂模型估计需要通过复杂迭代算法实现。
  - 迭代算法特有的依赖性使得模型估计很难被并行。
  - 全数据的迭代算法不可行。
- 各个击破策略：合理对数据分块然后分别估计，最后汇总。
  - 如何分块影响到模型估计的准确度，以线性回归为例 Ma et al. (2013)

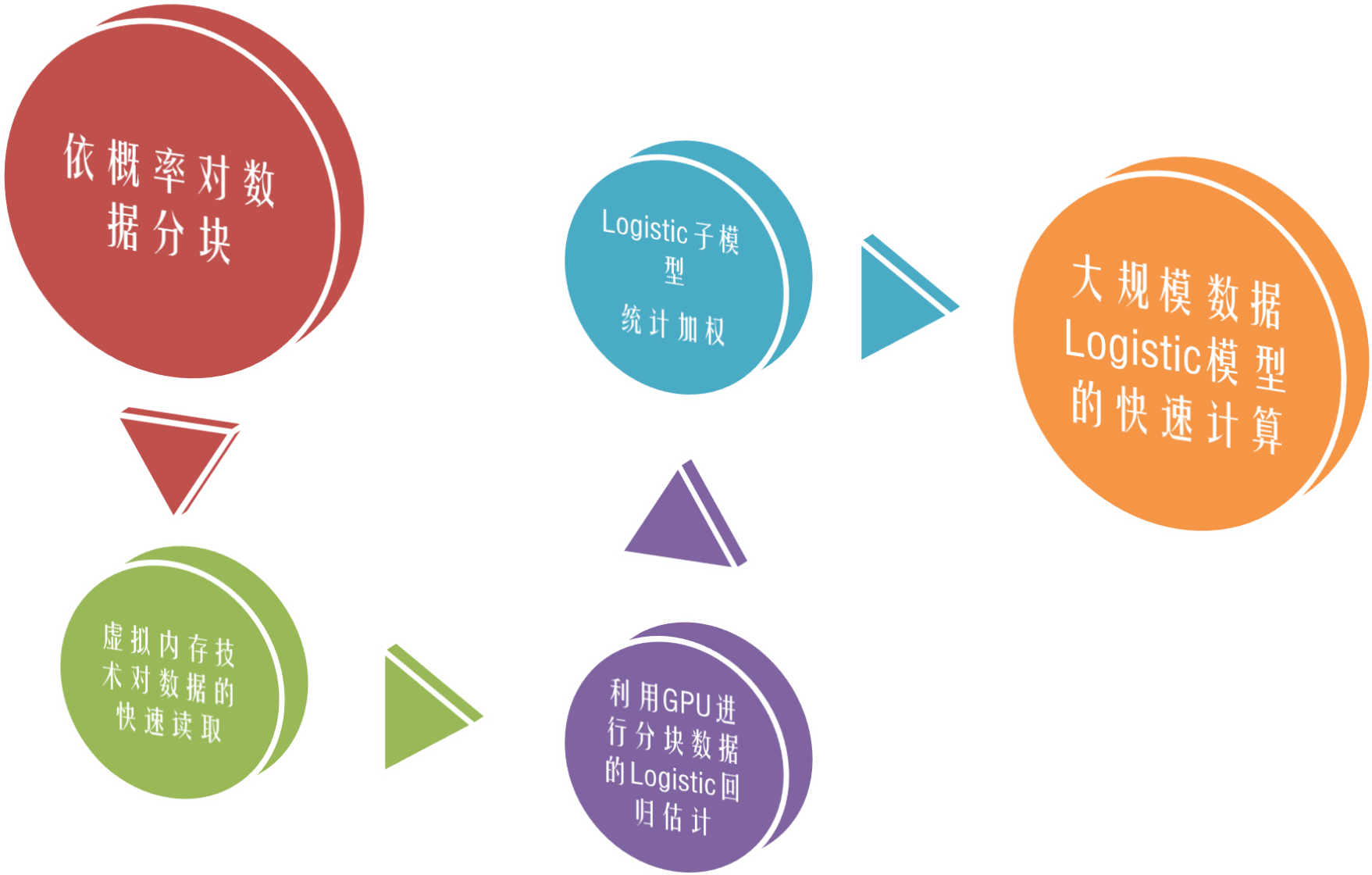


  - 理论上讲分块方法并没有降低计算的复杂度。但是提高单个分块模型的计算性能变得可能。

各个击破: 以 Logistic 回归为例



工作流程


依概率对数据分块
虚拟内存技术对数据的快速读取
利用GPU进行分块数据的Logistic回归估计
Logistic子模型统计加权
大规模数据Logistic模型的快速计算

理论支撑

- Logistic 回归模型

$$p(y_i) = \frac{\exp\{x_i'\beta\}}{1+\exp\{x_i'\beta\}}$$

其中 $x_i = (x_{i1}, ..., x_{ip})'$, $i = 1, ..., n$。利用极大似然估计可以得到 β 的估计值，$\widehat{\beta}$。

- 分块 Logistic 回归模型
  - 把 n 个样本分为 k 块，每块包含 m 个观测值。
  - 对于每一块数据作 Logistic 回归得到

$$\widehat{\beta}_l = \arg \max \sum_{i=1}^{m} \{y_{li}x_{li}'\beta - \log(1+\exp\{x_i'\beta\})\}.$$

  - 全部样本的 Logistic 回归模型的估计量 $\widehat{\beta}$ 可以利用分块 Logistic 回归模型 $\widehat{\beta}_l$ 加权得到

$$\widehat{\beta} = \frac{1}{k}\sum_{l=1}^{k}\widehat{\beta}_l.$$

一些极限性质

当 $m \to \infty$, $n \to \infty$，可以证明

- 相合性：

$$\widehat{\beta} \xrightarrow{p} \beta.$$

- 渐近正态性：

$$\sqrt{n}(\widehat{\beta} - \beta) \sim N(0, I(\beta))$$

其中

$$I(\beta) = \lim_{m \to \infty} \frac{1}{k} \sum_{l=1}^{k} \{ \frac{1}{m} \sum_{i=1}^{m} var(s(\beta_l)) \}^{-1}.$$

工作测试环境

- 五个高性能计算节点
- 128G×5 内存，物理硬盘
- 软件环境：Ubuntu Linux (64-bit), Intel Compiler, LAPACK, BLAS, Hadoop。
- 用户终端界面：R

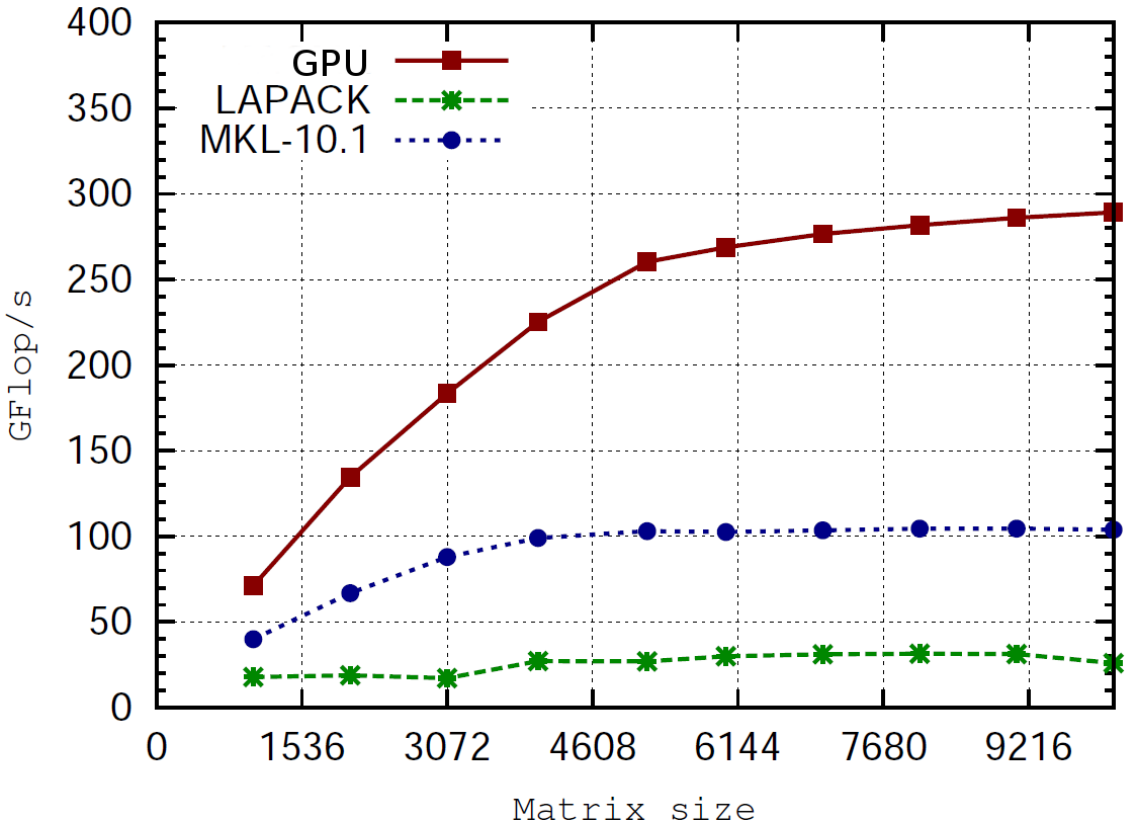
Bottom necks & Speedups

- 分块数据虽然可以读到内存中已经占用较多内存，很难继续进行数据计算。

- Memory-efficient R 内存管理 (C++ 实现 Transparent memory mapping, Kane (2010))

Bottom necks & Speedups

- 将 Logistic 回归分解，80% 时间用来计算对数据矩阵 $X'X$ 的 QR 分解。
- 利用 GPU 可以将 QR 分解提速 290 倍以上 (Tomov et al., 2010)。



扩展性

- 实现机制可以理解为单机版的 MapReduce，但不仅仅是 MapReduce。
- 方法具有一般性，可以拓展到一般的统计分类模型。
- 方法可以方便的部署到并行计算集群，进行超大规模统计模型的快速计算。

参考文献

Kane, M. J. (2010), "Scalable Strategies for Computing with Massive Sets of Data," Ph.D. thesis, Yale University.

Ma, P., Mahoney, M. W., and Yu, B. (2013), "A Statistical Perspective on Algorithmic Leveraging," *arXiv preprint arXiv:1306.5362*.

Tomov, S., Nath, R., Ltaief, H., and Dongarra, J. (2010), "Dense linear algebra solvers for multicore with GPU accelerators," in *Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW), 2010 IEEE International Symposium on*, IEEE, pp. 1–8.

谢谢！