Model Fitting with Distributed Data

Stanford Data Science Initiative Retreat

November 2, 2017
Goal

- Make it possible to build statistical models on distributed data.
  - Let participant sites maintain access control
  - Use distributed computation, i.e., aggregate computation results on distributed data
- Create an extensible framework so that others can contribute methods
- Create a deployment framework using a well-established service.
Pooled Data

<table>
<thead>
<tr>
<th></th>
<th>$Y_1$</th>
<th>$X_{1,1}$</th>
<th>$X_{1,2}$</th>
<th>...</th>
<th>$X_{1,p}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$Y_1$</td>
<td>$X_{1,1}$</td>
<td>$X_{1,2}$</td>
<td>...</td>
<td>$X_{1,p}$</td>
</tr>
<tr>
<td>2</td>
<td>$Y_2$</td>
<td>$X_{2,1}$</td>
<td>$X_{2,2}$</td>
<td>...</td>
<td>$X_{2,p}$</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>$n$</td>
<td>$Y_n$</td>
<td>$X_{n,1}$</td>
<td>$X_{n,2}$</td>
<td>...</td>
<td>$X_{n,p}$</td>
</tr>
</tbody>
</table>

$Y$ is the *outcome* (response) and the $X$s are *covariates* (features or predictors). The numbers 1, 2, $\ldots$, $n$ are unique observation (row) numbers.
Row-partitioned data

Figure: Left: Pooled data. Right: Distributed Data. Source: Wolfson, et. al (2010)
Communication Topology

A commonly used topology is the star network.

In a star network, a single node runs a master process that communicates with other nodes in the network that are clients or workers. The final results are available at the master and can then be shared back to clients.
Common Models

- Standard Linear Regression: response $Y$ on predictors $X$
- Poisson, Logistic, Multinomial
- Cox Regression
- Penalized versions of the above

In cancer studies, response is often *time to a particular event* (TTE), (death, relapse etc.)
A commonly used model for TTE is stratified Cox Regression.
Maximization via Newton-Raphson

\(K\) sites, \(l_k(\beta), S_k(\beta), I_k(\beta)\) are site-specific likelihood, score and information matrix.

0. Set \(i = 0, \beta_0 = 0\), a tolerance \(\epsilon\) and a maximum number of iterations \(B\).
1. Transmit \(\beta_i\) to each site
2. Each site \(k\) sends back \(l_k(\beta_i), S_k(\beta_i)\) and \(l(\beta_i)\)
3. Compute \(l(\beta_i) = \sum_{k=1}^{K} l(\beta_i), S(\beta_i) = \sum_{k=1}^{K} S_k(\beta_i)\), \(I(\beta_i) = \sum_{k=1}^{K} I_k(\beta_i)\),
4. Set

\[
\beta_{i+1} = \beta_i + I^{-1}(\beta_i)S(\beta_i)
\]

5. Stop if converged or iteration count exceeded. Else increment \(i\) and repeat step 1.

For the Cox Model, the convergence is very fast.
Site 1
Data: $X_1$
Summaries:
\[ l_1(\beta) = l(X_1, \beta), \]
\[ S_1(\beta) = l_1'(X_1, \beta), \]
\[ I_1(\beta) = -S_1'(X_1, \beta) \]

Site 2
Data: $X_2$
Summaries:
\[ l_2(\beta) = l(X_2, \beta), \]
\[ S_2(\beta) = l_2'(X_2, \beta), \]
\[ I_2(\beta) = -S_2'(X_2, \beta) \]

Site 3
Data: $X_3$
Summaries:
\[ l_3(\beta) = l(X_3, \beta), \]
\[ S_3(\beta) = l_3'(X_3, \beta), \]
\[ I_3(\beta) = -S_3'(X_3, \beta) \]

Site 4
Data: $X_4$
Summaries:
\[ l_4(\beta) = l(X_4, \beta), \]
\[ S_4(\beta) = l_4'(X_4, \beta), \]
\[ I_4(\beta) = -S_4'(X_4, \beta) \]

Master Site
\[ \beta = 0 \]
Iterate to convergence:
\[ S = \sum S_i(\beta), I = \sum I_i(\beta) \]
\[ \beta_{i+1} = \beta_i + I^{-1}S \]
Workflow and Implementation

The prototype implementation uses the R language and OpenCPU that exposes R as a RESTful (http) service. The main steps are the following.

1. Define the Computation
2. Set up a Worker Process for the Computation
3. Set up a Master Process for the Computation
4. Run the Computation

We address each in turn.
1. Defining the Computation

Requirements: \texttt{R, distcomp} package

Computation Type \rightarrow Stratified Cox Model
\rightarrow Rank k SVD

Computation-specific Parameters \rightarrow Shiny App
\rightarrow \texttt{defineNewComputation()}

Exemplar Data \rightarrow Definition file (\texttt{defn.rds})
2. Setting up Worker

- OpenCPU URL
- Definition file (`defn.rds`)
- Site Data
- Shiny App
- setupWorker()
- Site ready
3. Setting up Master

- Full Path of Workspace
- Full Path of `defn.rds`
- Worker Site URLS

Shiny App

```
setupMaster()
```

R script
Example: Treatment data on reducing drug abuse.

Pooled Data fit:

|         | coef   | exp(coef) | se(coef) | z     | Pr(>|z|) |
|---------|--------|-----------|----------|-------|----------|
| age     | -0.028076 | 0.972315  | 0.008131 | -3.453 | 0.000554 *** |
| becktota| 0.009146  | 1.009187  | 0.004991 | 1.832  | 0.066914 . |
| ndrugfp1| -0.521973 | 0.593349  | 0.124424 | -4.195 | 2.73e-05 *** |
| ndrugfp2| -0.194178 | 0.823512  | 0.048252 | -4.024 | 5.72e-05 *** |
| ivhx3TRUE| 0.263634 | 1.301652  | 0.108243 | 2.436  | 0.014868 *  |
| race    | -0.240021 | 0.786611  | 0.115632 | -2.076 | 0.037920 *  |
| treat   | -0.212616 | 0.808466  | 0.093747 | -2.268 | 0.023331 *  |

Distributed Data fit:

<table>
<thead>
<tr>
<th></th>
<th>coef</th>
<th>exp(coef)</th>
<th>se(coef)</th>
<th>z</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>-0.0280495</td>
<td>0.97234</td>
<td>0.0081301</td>
<td>-3.4501</td>
<td>5.6041e-04</td>
</tr>
<tr>
<td>becktota</td>
<td>0.0091441</td>
<td>1.00919</td>
<td>0.0049918</td>
<td>1.8318</td>
<td>6.6979e-02</td>
</tr>
<tr>
<td>ndrugfp1</td>
<td>-0.5219296</td>
<td>0.59337</td>
<td>0.1244240</td>
<td>-4.1948</td>
<td>2.7315e-05</td>
</tr>
<tr>
<td>ndrugfp2</td>
<td>-0.1941709</td>
<td>0.82352</td>
<td>0.0482507</td>
<td>-4.0242</td>
<td>5.7168e-05</td>
</tr>
<tr>
<td>ivhx3TRUE</td>
<td>0.2636376</td>
<td>1.30166</td>
<td>0.1082448</td>
<td>2.4356</td>
<td>1.4868e-02</td>
</tr>
<tr>
<td>race</td>
<td>-0.240021</td>
<td>0.78658</td>
<td>0.1156319</td>
<td>-2.0761</td>
<td>3.7887e-02</td>
</tr>
<tr>
<td>treat</td>
<td>-0.2125720</td>
<td>0.80850</td>
<td>0.0937466</td>
<td>-2.2675</td>
<td>2.3359e-02</td>
</tr>
</tbody>
</table>
A distributed Stratified Cox Model fit for Stage 4 Breast Cancer data was fit using time to death (overall survival) among

- Stanford University
- Vanderbilt University
- Mt. Sinai Hospital
- Palo Alto Medical Foundation
- Group Health Seattle (now part of Kaiser).

Variables in the model, included age, race, whether they were on chemo, or hormonal or targeted therapies. The model found significant coefficients for targeted and hormonal therapies. When Mt. Sinai was included, race was also significant.
Sites may be wary of sharing even summaries. A *homomorphic encryption* scheme $E$ is an encryption scheme such that:

\[ E(x) + E(y) = E(x + y). \]

So, one can recover the value of $x + y$ by adding encrypted values of $x$ and $y$ and decrypting the sum!
One Implementation

Homomorphic Encryption ensures that

$$E(r) + E(l_1) + E(l_2) + E(l_3) = E(r + l_1 + l_2 + l_3) = e_l (\text{say}).$$

So we can decrypt $D(e_l) = e_l - r$ will yield $l_1 + l_2 + l_3$!

The R package homomorpheR implements a prototype using Pallier encryption scheme.
Simulated Example

Three sites each with data sizes 1000, 500 and 1500 with

- sex (0, 1) for male/female
- age between 40 and 70
- a biomarker bm
- a time to some event of interest
- an indicator event which is 1 if an event was observed and 0 otherwise.

Model is

\[ S(t, \text{age}, \text{sex}, \text{bm}) = [S_0(t)]^{\exp(\beta_1 \text{age} + \beta_2 \text{sex} + \beta_3 \text{bm})} \]
### Results

**Pooled Data fit:**

<table>
<thead>
<tr>
<th></th>
<th>coef</th>
<th>exp(coef)</th>
<th>se(coef)</th>
<th>z</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>sex</td>
<td>-0.17959</td>
<td>0.83562</td>
<td>0.05069</td>
<td>-3.54</td>
<td>0.0004</td>
</tr>
<tr>
<td>age</td>
<td>0.02009</td>
<td>1.02029</td>
<td>0.00286</td>
<td>7.02</td>
<td>2.1e-12</td>
</tr>
<tr>
<td>bm</td>
<td>0.00682</td>
<td>1.00684</td>
<td>0.02501</td>
<td>0.27</td>
<td>0.7852</td>
</tr>
</tbody>
</table>

**Distributed Data Homomorphic fit:**

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>Std. Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>-0.179585179</td>
<td>0.050694606</td>
</tr>
<tr>
<td>sex</td>
<td>0.020087782</td>
<td>0.002859509</td>
</tr>
<tr>
<td>bm</td>
<td>0.006815326</td>
<td>0.025006028</td>
</tr>
</tbody>
</table>
Challenges

- Data quality varies from site to site, especially follow-up
- Tools needed for data harmonization and social networking aspects of collaboration
- Deployment times at sites varied from 3 to 6 months
- IT folks don’t know R!
- Security concerns, audit trails, fault tolerance