Big data challenges for physics in the next decades

David W. Hogg

Center for Cosmology and Particle Physics, New York University

2012 November 09
Huge data sets create new opportunities. They also present new challenges.

Industrial (dot-com) methods are brilliant. They can only solve a very limited set of problems.

We won’t reap the full benefit of larger data sets without new technology. Call to arms. (And get rich too!)
- Large Synoptic Survey Telescope: $10^{10}$ galaxies in $10^{15}$ pixels
  - get the cosmic gravitational-lensing shear map
  - and then the cosmological parameters
  - $10^9$ stars too, all moving and varying
- Gaia: $10^9$ stars in $10^{12}$ pixels
  - infer the dynamics of the Milky Way
  - but also—necessarily—the distribution function of stars in that potential
  - precision requirements are outrageous (milli-pixel positions)
- Large Hadron Collider instruments: taking data $10^7$ times faster than it can be moved to disk
  - they found the Higgs! (probably)
  - have to make hard cuts and throw away data
how do you store big data sets?

- Distribute the data.
  - < 1 Tb of data per CPU, thousands of CPUs
  - all modern databases can do this, even open-source ones
- There is a CPU near every data point.
  - and there are many, many CPUs
- Hardware is cheap; management is expensive.
- Massive CPU redundancy creates opportunities...
map–reduce or die

▶ “We won’t even consider algorithms that can’t be written in the map–reduce framework.”
map–reduce

- **map:**
  - at each “data point” (on the distributed system), do an operation on that datum, produce output
  - if "kittens" is in document:
    return (URL, PageRank)
  - **distributed data** is the key: Every datum is near a CPU.

- **reduce:**
  - between each pair of outputs, do an operation and return one new output, recurse up the tree
  - if PageRank[0] > PageRank[1]:
    return (URL[0], PageRank[0])
  else:
    return (URL[1], PageRank[1])
  - **tree structure** of the data center is the key: There are only $\log_2 N$ branches to any datum.
map–reduce

"kittens?"

reduce reduce reduce reduce

reduce reduce reduce reduce reduce

reduce reduce reduce reduce reduce

reduce reduce reduce reduce

reduce reduce reduce reduce

reduce reduce reduce

reduce reduce reduce

reduce reduce reduce

reduce reduce reduce

reduce reduce reduce

reduce reduce reduce

reduce reduce reduce
Brilliant. And a huge opportunity.
  - the computational complexity is $N \log N$
  - but the **time it takes** scales as just $\log N$

Map–reduce made internet search **possible**.
  - (goes by many names, some trade-marked!)
maximum-likelihood and map–reduce

- full-data likelihood: \( p(D \mid \theta) = \prod_p p(d_n \mid \theta) \)
- Find a **local maximum with respect to** \( \theta \) of this likelihood.
- **map:**
  - compute \( \frac{d \ln p(d_n \mid \theta)}{d \theta} \)
- **reduce:**
  - pairwise sum
- Go uphill. Repeat as necessary; each iteration only takes \( \log N \) time.
  - (use L-BFGS or conjugate-gradient or whatever you like)
  - complexity is \( N \log N \), **compute time** is \( \log N \)
- Local optimization typically takes \( M^2 \log N \) time, where \( M \) is the **number of parameters**.
  - (maybe \( M^3 \) if you want a full error analysis)
all that matters is the number of parameters

▶ So we are done, right?
all that matters is the number of parameters

- So we are done, right?
- Nope.
all that matters is the number of parameters

- So we are done, right?
- Nope.
- In all **real** problems, the number of parameters $M$ **scales with the data volume** $N$.
- If you want to go beyond maximum-likelihood, things get **hard**.
physics problems are hierarchical

\[ n = 1, \ldots, N \]

\[ \sum \sigma_n \]

\[ \Omega \]

\[ \gamma \]

\[ \epsilon_{n}^{\text{obs}} \]

\[ \epsilon_{n}^{\text{true}} \]

\[ x_n \]

\[ \alpha \]
Nuisance parameters

- Nuisance parameters tend to increase in number with the data size.
- Relevant backgrounds get more subtle and must be modeled more carefully.
- Details of sample selection and observed data distribution functions become more important.
- As precision expectations rise (and they rise with $N$), noise models get more realistic.
  - all these effects bring new parameters
calibration parameters

- As you go from $N = 10^3$ to $N = 10^9$, you expect to do more than $10^3$ times better in accuracy.
- Calibration of the device must get correspondingly better.
  - time-dependence, temperature, Solar Cycle, hysteresis
  - all these effects bring new parameters
  - you usually can’t measure these parameters well enough in your “calibration program”; when $N$ is large, you end up self-calibrating
Bayesian inference isn’t map–reduce

\[ p(\theta \mid D) = \frac{1}{Z} p(\theta) \prod_n p(d_n \mid \theta) \]

- map:
  - compute functions \( p(d_n \mid \theta) \)
- reduce:
  - product functions together (starting with the prior)
- but think about how you pass forward those functions
  - \( \theta \) has \( 10^6 \) or more parameters
  - functions are multi-modal
  - support is broader than Gaussian
  - when the data get large, the \textbf{resolution required} becomes unsustainable
even the frequentists are doomed

▶ All the “$M$ scaling with $N$” arguments apply to frequentists and Bayesians alike.
▶ Computing the full-data likelihood function is just as hard as computing the full-data posterior PDF.
  ▶ (local optimization of the likelihood is easy, full description of the function is hard)
marginalization is hard—and unavoidable

\[ \sum_{n=1}^{N} \sigma_n \rightarrow \epsilon_{n}^{\text{obs}} \rightarrow \epsilon_{n}^{\text{true}} \rightarrow \alpha \rightarrow \Omega \rightarrow \gamma \]

\[ n = 1, \ldots, N \]
Bayesian state-of-the-art

- There **are** huge non-parametric Bayesian inferences with massive marginalizations out there.

- How were they done?
  - carefully chosen priors that make the inferences and marginalizations analytic or tractable
  - **we can’t do this**
  - why not? Because for us the priors **actually are** our prior beliefs. Our **real** prior beliefs are not conjugate to anything!

- “Bayesian” is becoming a bad word.
my approach

- Brute Force (tm).
- Plus some help from applied math and computer vision.
- approximate Bayesian inference
- very clever Markov-Chain Monte Carlo methods
- exploiting problem sparsity
my approach

- Brute Force (tm).
- Plus some help from applied math and computer vision.
- approximate Bayesian inference
- very clever Markov-Chain Monte Carlo methods
- exploiting problem sparsity
my approach

- Brute Force (tm).
- Plus some help from applied math and computer vision.
  - approximate Bayesian inference
  - very clever Markov-Chain Monte Carlo methods
  - exploiting problem sparsity
my day job

- Lang & Hogg (forthcoming): a $10^9$-parameter model of the $10^{13}$ Sloan Digital Sky Survey pixels (The Tractor)
- Brewer et al. (forthcoming): Bayesian non-parametrics but with priors that represent our actual prior knowledge
- Foreman-Mackey et al. (arXiv:1202.3665): *emcee*, the MCMC Hammer: flexible, parallelized, adaptive sampler
- Bovy et al. (arXiv:1105.3975): a 60,000-parameter model of 700,000 flux measurements, followed by predictions for 160,000,000 point sources
it’s actually worse than you think

- “More data” means “more new discoveries”.
- The **number of hypotheses to test** also grows with the data size $N$.
- There will be far more hypotheses than physicists.
  - (this is already true, of course)
  - citizen science?
  - robot science?
Huge data sets create new opportunities.
  - they also present new challenges

Industrial (dot-com) methods are brilliant.
  - they can only solve a very limited set of problems

We won’t reap the full benefit of larger data sets without new technology.
  - call to arms
  - (and get rich too!)