Big Data Bootstrap

CS 584: Big Data Analytics

Material adapted from
Michael Jordan's talks (https://simons.berkeley.edu/sites/default/files/docs/509/jordanslides.pdf) &
(https://www.cs.berkeley.edu/~ameet/blb_workshop_slides.pdf)

Big Data Problem?

- Data has not been viewed as a resource, but as a "workload"
- Fundamental issue is data needs to be viewed as a resource and combined with other resources to yield timely, cost-effective, high-quality decisions and inferences
- Just as with time or space, it should be the case that the more of the data resource the better

Leveraging More Data Issues

- Query complexity grows faster than the number of data points
  - More rows in a table —> more columns
  - Number of hypotheses grows exponentially in the number of columns
  - More data —> greater chance that random fluctuations look like signals (e.g., more false positives)
- Sophisticated algorithms will be unlikely to run in an acceptable time frame with more data
  - Back off to cheaper algorithms that may be more error-prone
  - Subsample but requires knowing statistical value of each data point, which we generally don't know a priori

Assessing the Quality of Inference

- Data mining and machine learning are full of algorithms for clustering, classification, regression, etc.
    - Missing is a focus on **uncertainty** in the outputs of such algorithms (“error bars”)
- Driven by the follow application: develop a database that returns answer with error bars to all queries
    - The framework should be used on large-scale problems

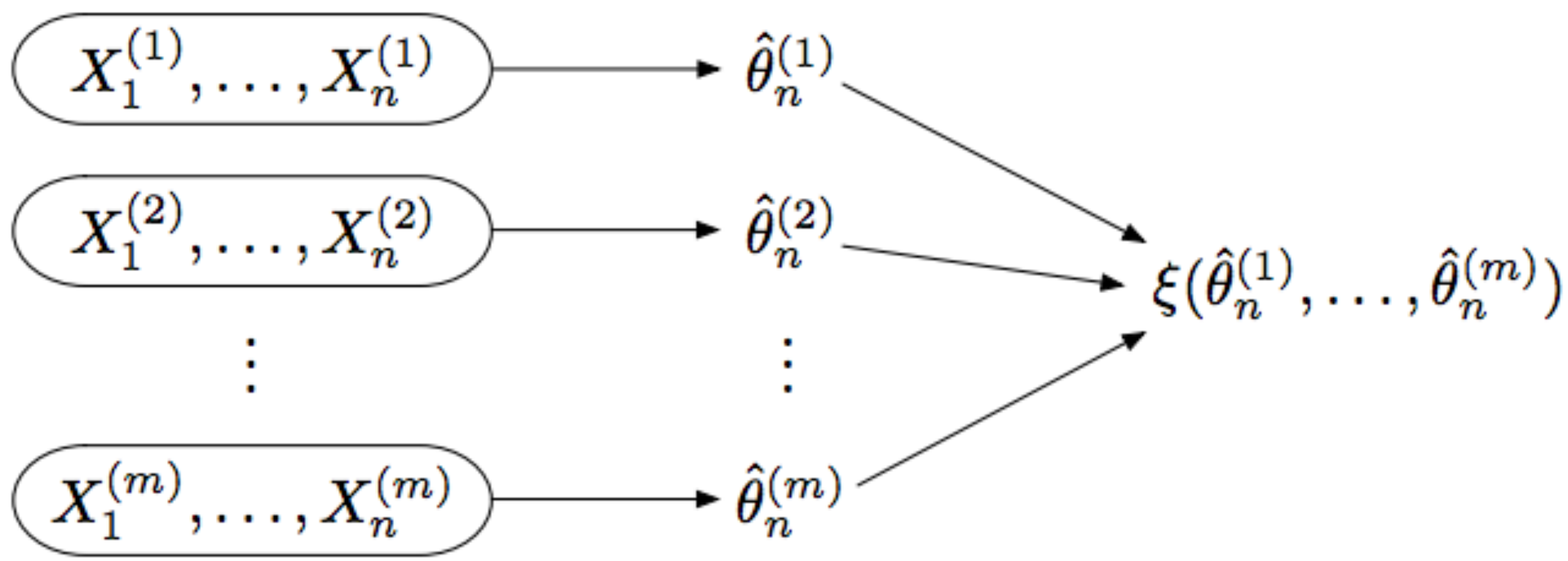
Big Data Bootstrap Setting

- Observe data $X_1, \cdots, X_n$
- Form an estimate $\hat{\theta}_n = \theta(X_1, \cdots, X_n)$
  (e.g., weight parameters in linear regression, a classifier, etc.)
- Compute an assessment ξ of the quality of estimator $\hat{\theta}$
  (e.g., confidence region)

Goal is a procedure for quantifying estimator quality which is accurate, automatic, and scalable

The Unachievable Ideal

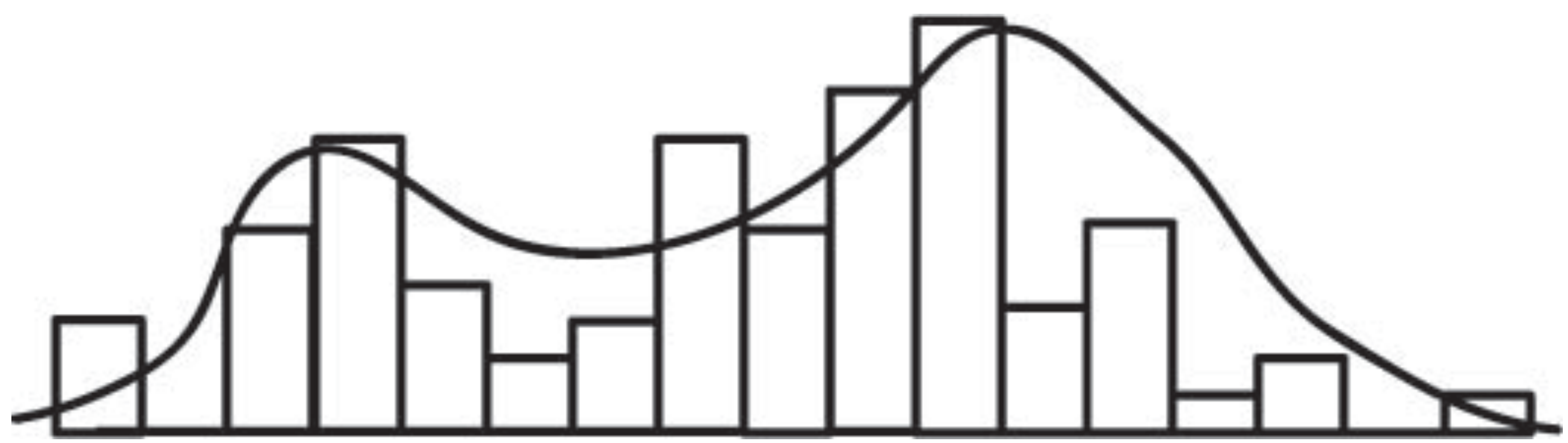
- Observe many independent datasets of size n
- Compute $\hat{\theta}_n$ on each
- Compute ξ based on these multiple realizations of $\hat{\theta}_n$



But we only observe one dataset of size n

Underlying Population

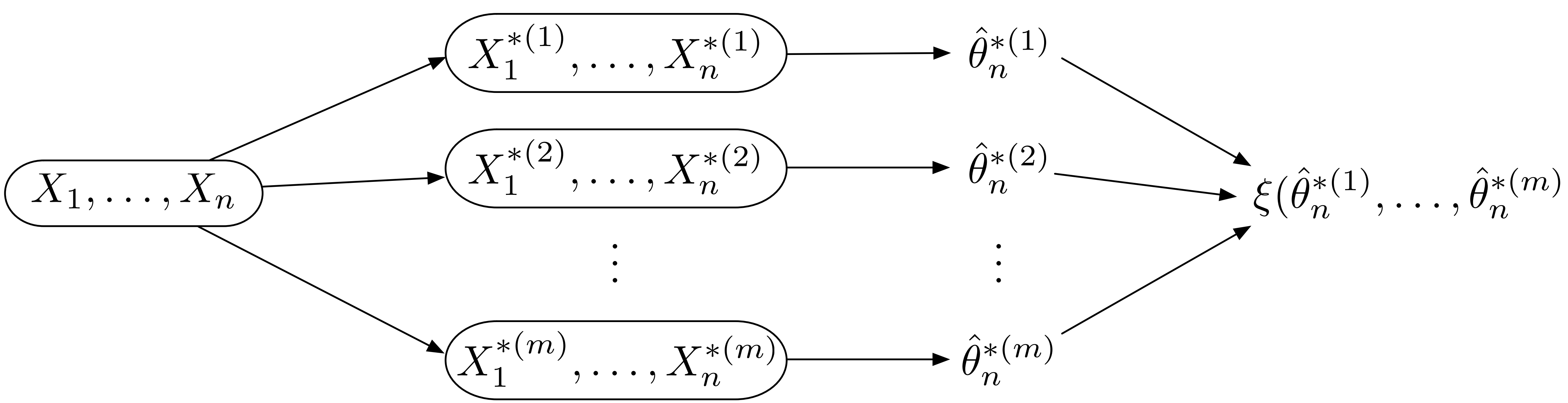
A Sample from the Population



Prior Work: Bootstrap (Efron, 1979)

Use the observed data to simulate multiple datasets of size n

- Repeatedly resample n points with replacement from the original dataset of size n
- Compute $\hat{\theta}_n$ on each resample
- Compute ξ based on these multiple realizations $\hat{\theta}_n$



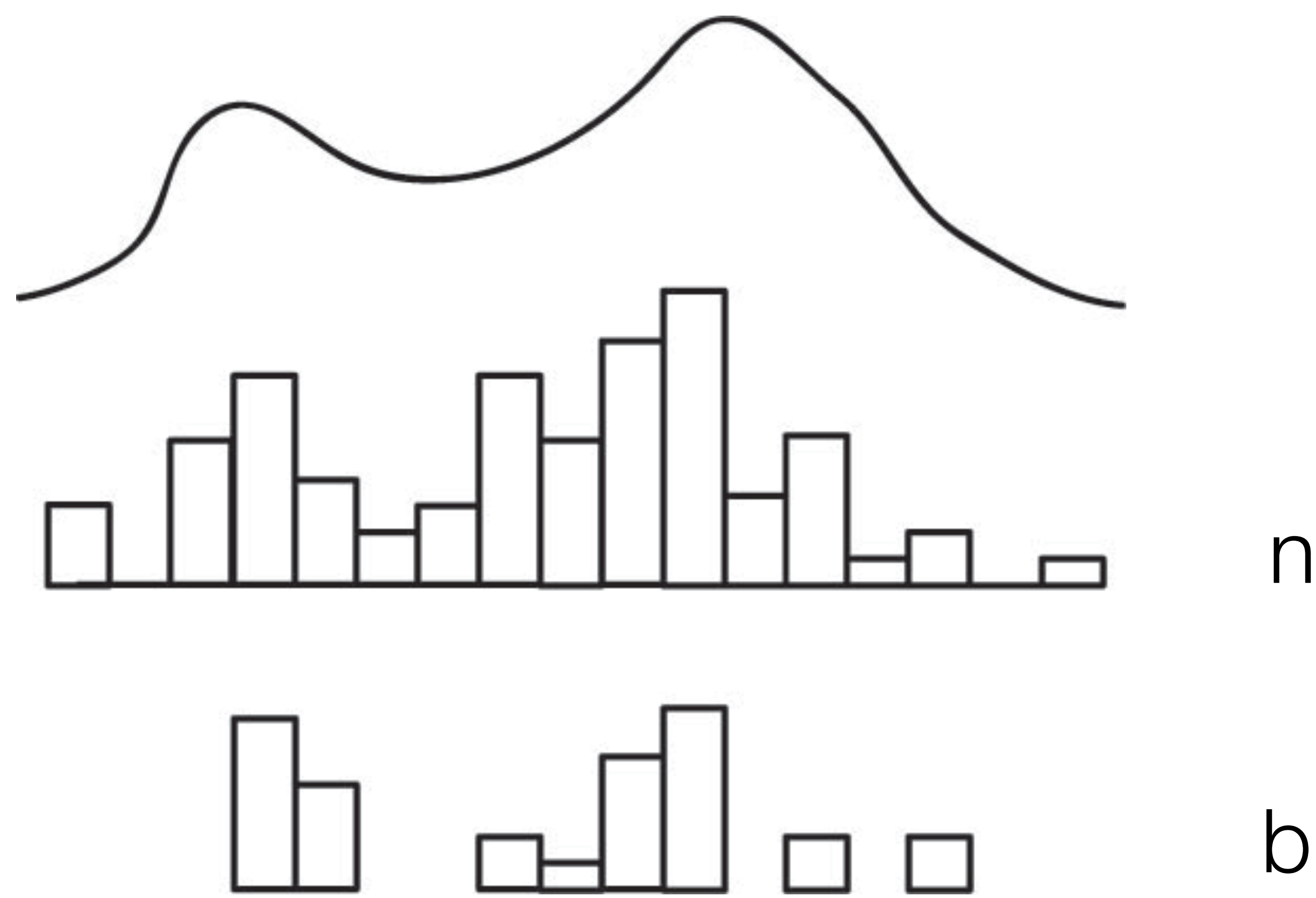
Prior Work: Bootstrap Computational Issues

- Expected number of distinct points in a resample is ~0.632n
- Resources required to compute estimate generally scale in number of distinct data points
  - True of many commonly used learning algorithms (e.g., SVM, logistic regression, linear regression, kernel methods, etc.)
  - Use weighted representation of resampled datasets to avoid physical data replication
- Example: If original dataset has size 1 TB, then each resample is expected to be of size ~632 GB

Prior Work: Bootstrap

- Advantages
  - Accurate for a wide range of θ
  - Automatic - can compute without knowledge of the internals of θ
- Disadvantages
  - Must repeatedly compute θ on ~63% of the data
  - Difficult to parallelize across different computations of θ

Prior Work: Subsampling (Politis, Romano & Wolf 1999)



Prior Work: Subsampling

- Compute estimate on smaller resamples of the data of size b where b < n
- Obtain fluctuations of the estimate and thus error bars
- Key issue: since b < n, the error bars will be on the wrong scale (too large) so need to analytically correct to produce the final estimate

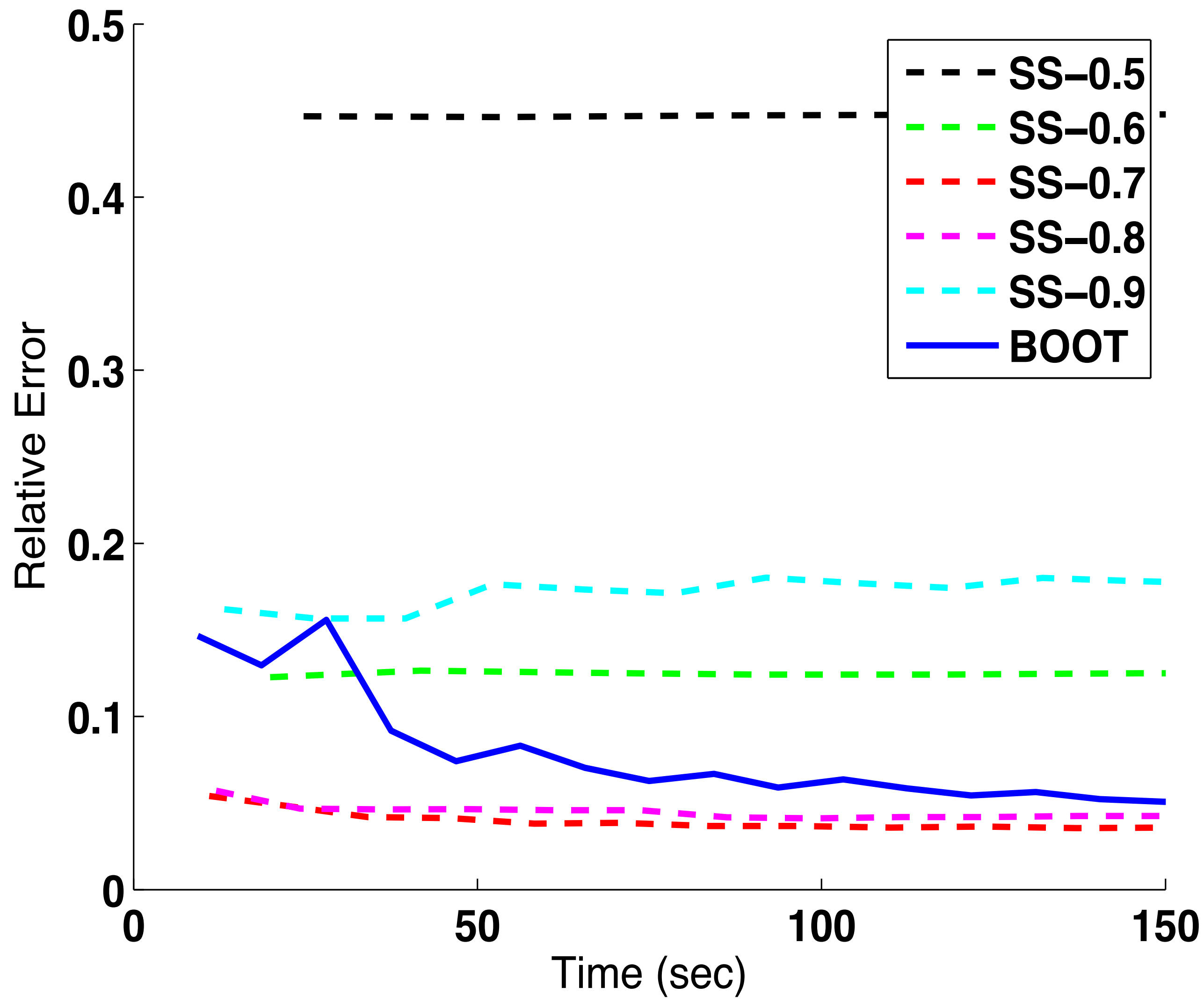
Prior Work: Subsampling

- Advantages
  - Much more favorable computational profile than the bootstrap
- Disadvantages
  - Accuracy sensitive to choice of b
  - Analytical correction introduces some dependency on internals of θ

Empirical Results: Bootstrap and Subsampling

- Multivariate linear regression with d = 100 on n=20,000 on synthetic data
- x values sampled independently from coordinate-wise Gamma distributions
- y = wx + e, where w is a fixed weight vector and e is independent Gamma noise
- Estimate $\hat{\theta}_n = \hat{w} \in \mathbb{R}^d$ via least squares
- Compute a marginal confidence interval for each component of w and assess accuracy via relative mean absolute deviation from true confidence interval size

Empirical Results: Bootstrap and Subsampling

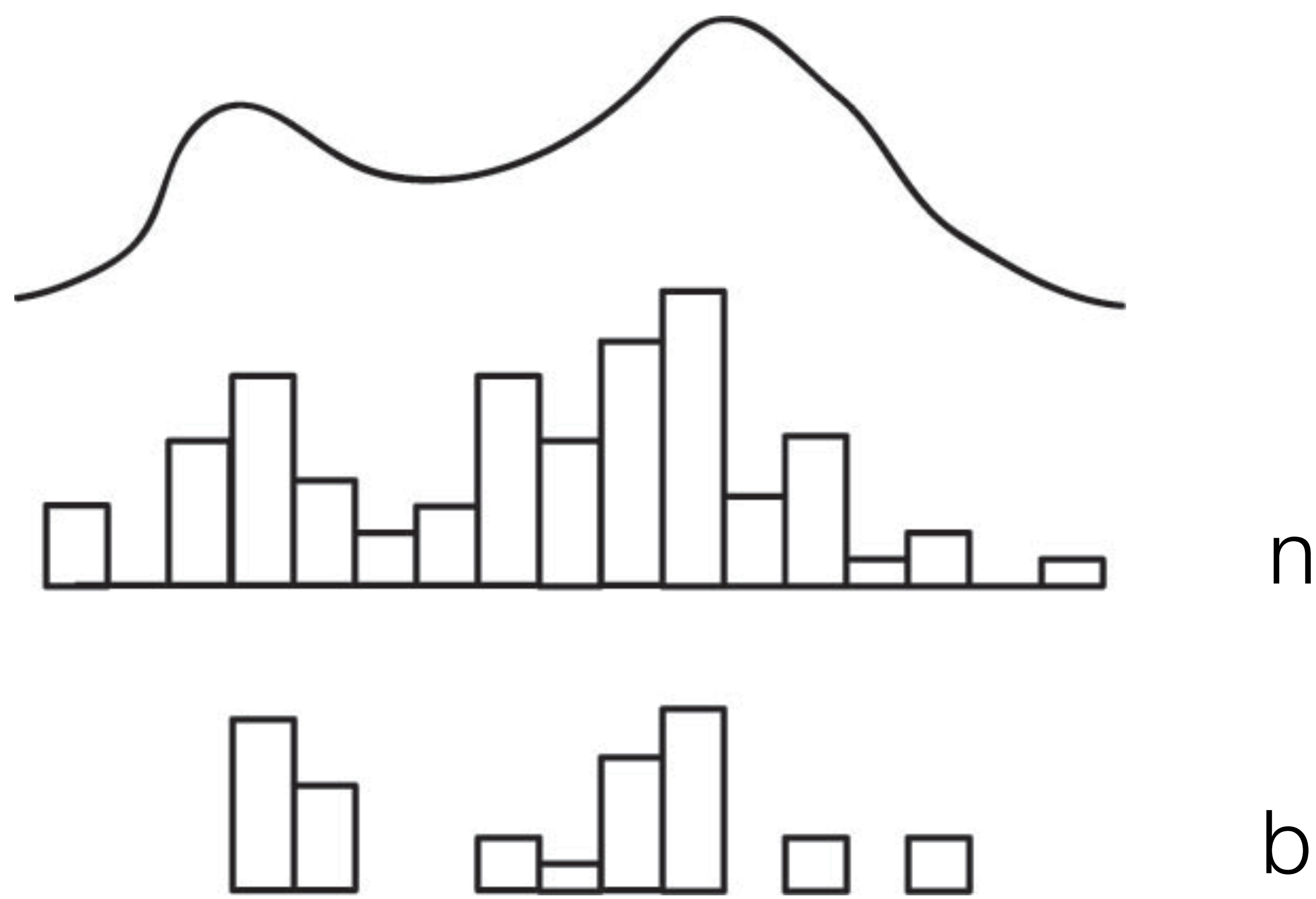


$b = n^{\gamma}$

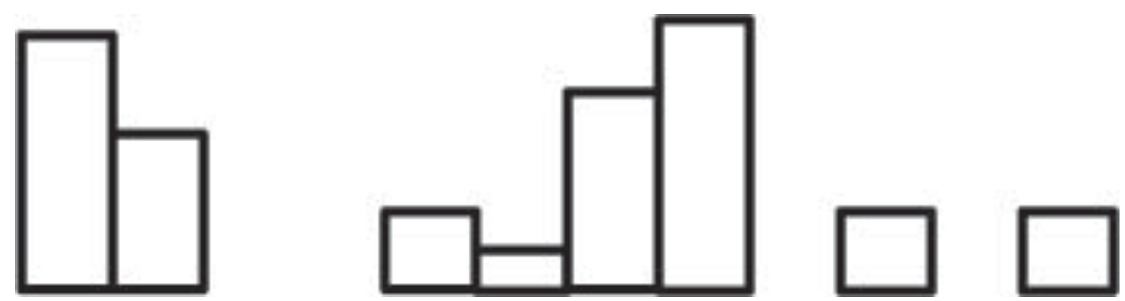
Bag of Little Bootstraps (BLB)

- Combines bootstrap and subsampling and gets the best of best worlds
- Works with small subsets of the data, like subsampling, and thus is appropriate for distributed computing platforms
- But, like bootstrap, doesn’t require analytical rescaling
- And it’s successful in practice

Towards BLB



Pretend Subsample is the Population



- Bootstrap the subsample!
- This means resampling n times with replacement and not b times as in subsampling

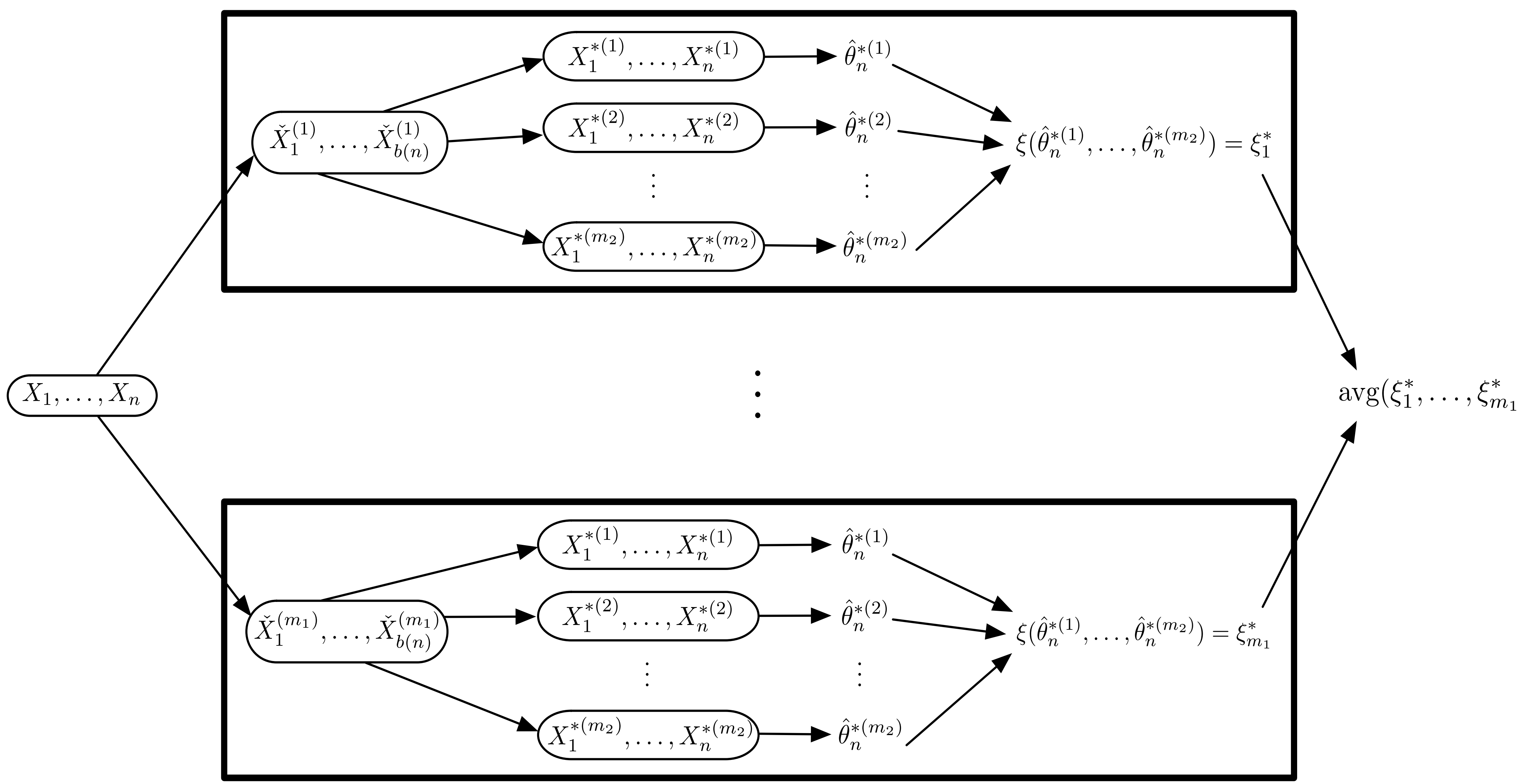
BLB

- Subsample contains only b points so the resulting empirical distribution has its support on b points
- But we can (and should!) resample it with replacement n times, no b times
- Doing this repeatedly for a given subsample gives bootstrap confidence intervals on the right scale — no analytical rescaling is necessary!
- This can be done in parallel for multiple subsamples and combine the results

BLB Algorithm

- Repeatedly subsample b points without replacement from the original dataset of size n
- For each subsample do:
  - Repeatedly resample n points with replacement from the subsample
  - Compute estimate on each resample
  - Compute estimate of quality based on these multiple resampled realizations
- One estimate of quality per sample. Output their average as our final estimate

BLB


$X_1, \ldots, X_n$
$\check{X}_1^{(1)}, \ldots, \check{X}_{b(n)}^{(1)}$
$X_1^{*(1)}, \ldots, X_n^{*(1)}$
$X_1^{*(2)}, \ldots, X_n^{*(2)}$
$X_1^{*(m_2)}, \ldots, X_n^{*(m_2)}$
$\hat{\theta}_n^{*(1)}$
$\hat{\theta}_n^{*(2)}$
$\hat{\theta}_n^{*(m_2)}$
$\xi(\hat{\theta}_n^{*(1)}, \ldots, \hat{\theta}_n^{*(m_2)}) = \xi_1^*$
$\check{X}_1^{(m_1)}, \ldots, \check{X}_{b(n)}^{(m_1)}$
$\xi(\hat{\theta}_n^{*(1)}, \ldots, \hat{\theta}_n^{*(m_2)}) = \xi_{m_1}^*$
$\mathrm{avg}(\xi_1^*, \ldots, \xi_{m_1}^*)$

BLB Computational Considerations

- Use weighted representation of resampled datasets to avoid physical data replication
- Many commonly used estimation algorithms scale in number of distinct data points
- Example: If $n = 1{,}000{,}000$ with each data point 1 MB. If $b = n^{0.6}$, then
  - Full dataset has size 1 TB
  - Subsampled datasets ~ 4 GB
  - In contrast, bootstrap resamples ~632 GB

BLB Properties

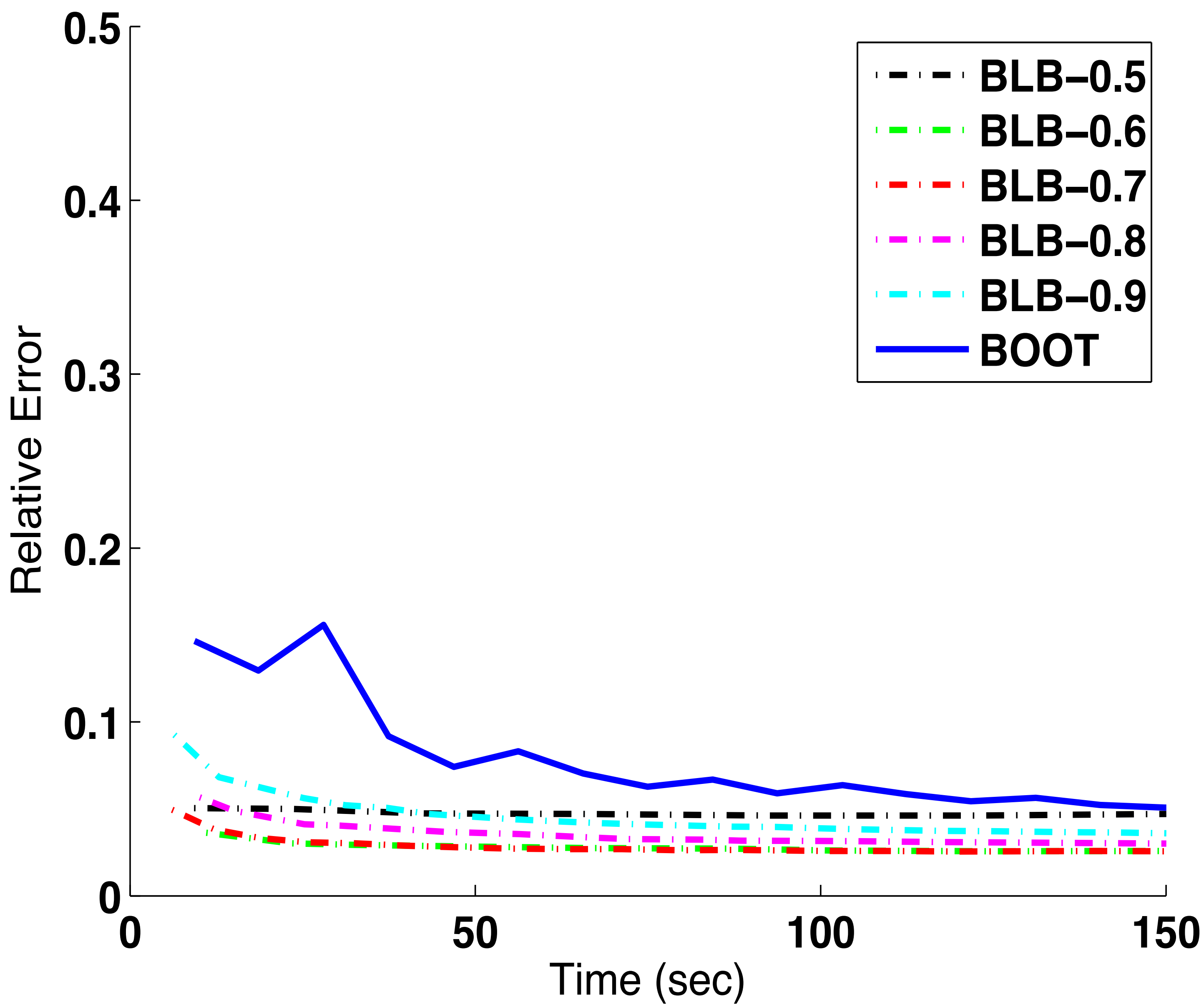
- Like Bootstrap
  - Accurate for wide range of θ. Shares the bootstrap's consistency and higher-order correctness.
  - Automatic - can compute without knowledge of the internals of θ
- Beyond Bootstrap and Subsampling
  - Can explicitly control b
  - More robust to choice of b, which can be much smaller than n
  - Generally faster than bootstrap and requires less total computation
  - Easy to parallelize across different computations of θ

BLB Hyperparameter Selection

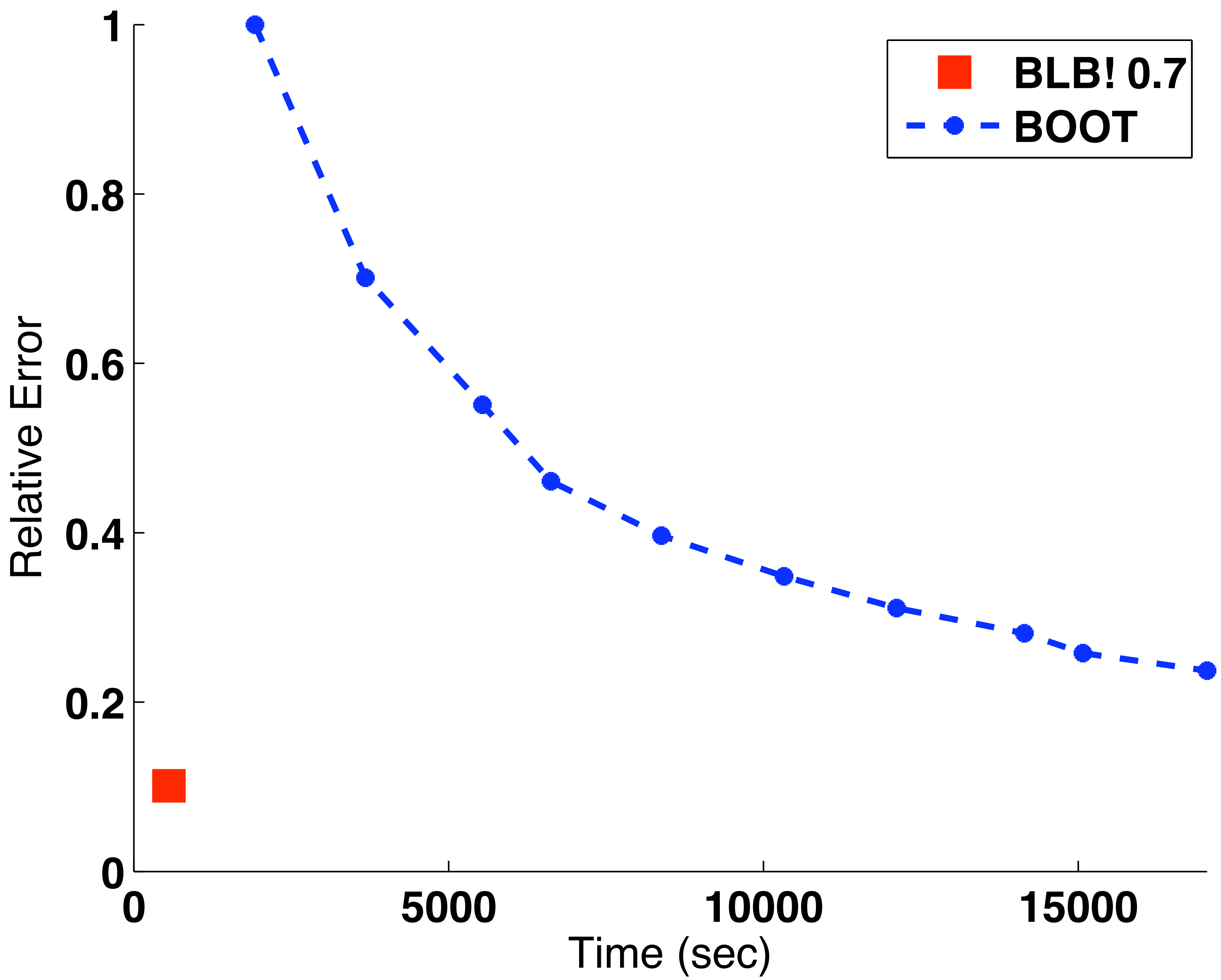
- b = the number of unique samples for each bag
- s = the number of size b samples w/o replacement
- r = the number of inner bootstrap samples

- b: the larger the better although $b = n^{0.7}$ works well
- s, r: adaptively increase this until a convergence condition is reached (median doesn't change)

Empirical Results: BLB w/ n=20,000

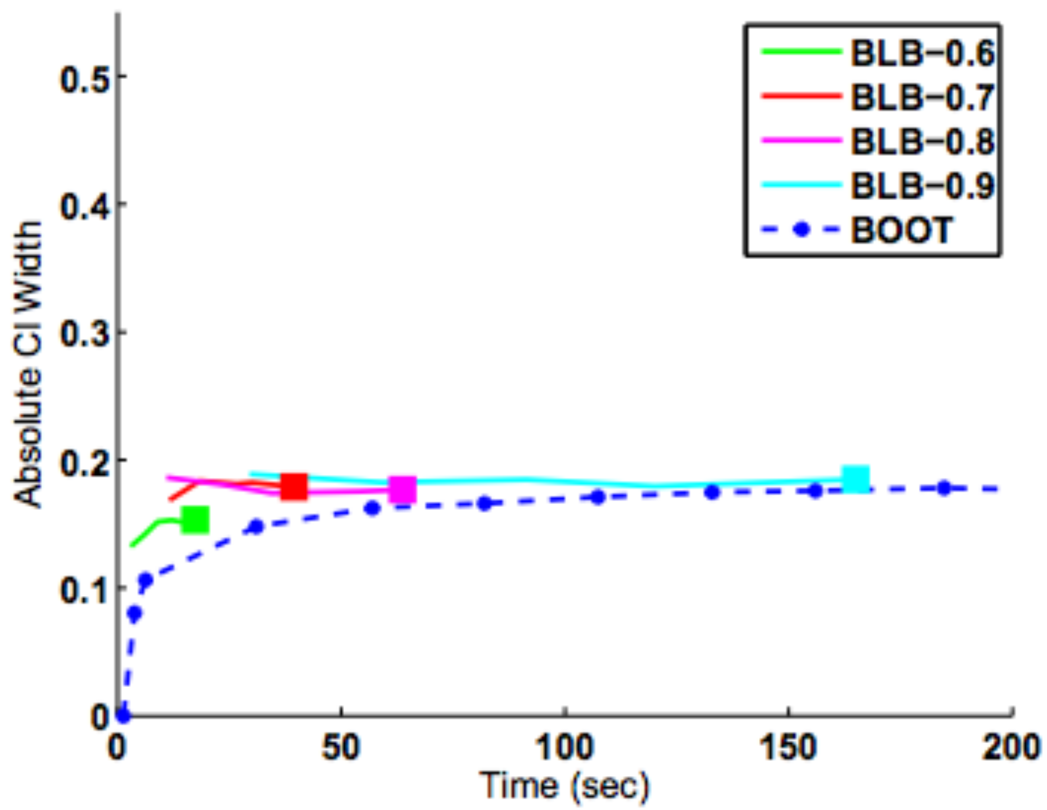


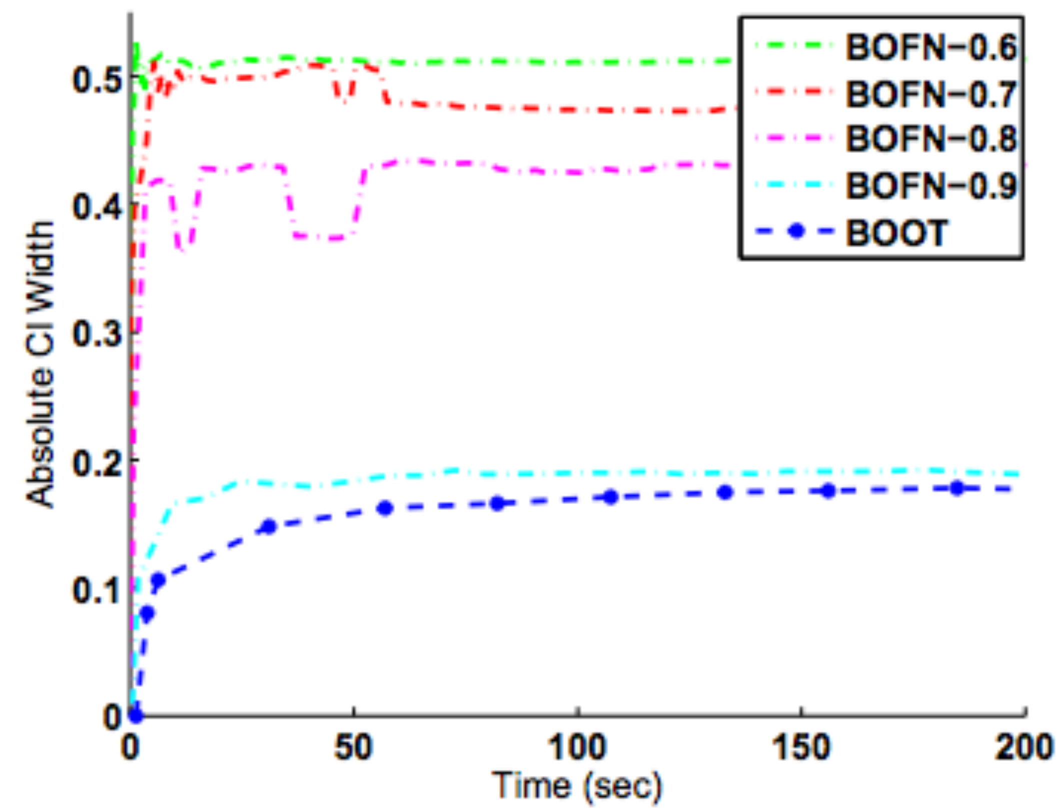
Empirical Results: BLB on 150 GB data



Empirical Results: UCI connect 4 dataset

Logistic regression, d=42, n=67,557





BLB Summary

- Shares the bootstrap's favorable statistical properties (consistency & higher-order correctness)
- Permits computation on multiple subsamples and resamples simultaneously in parallel
- Well-suited to large-scale data and modern, parallel, and distributed computing architectures