Coresets for Bayesian Logistic Regression

Tamara Broderick
ITT Career Development Assistant Professor, MIT

With: Jonathan H. Huggins, Trevor Campbell
Bayesian inference
Bayesian inference

- Complex, modular
Bayesian inference

- Complex, modular; coherent uncertainties
Bayesian inference

- Complex, modular; coherent uncertainties; prior info
Bayesian inference

- Complex, modular; coherent uncertainties; prior info

\[ p(\theta) \]
Bayesian inference

- Complex, modular; coherent uncertainties; prior info

\[ p(y|\theta)p(\theta) \]
Bayesian inference

- Complex, modular; coherent uncertainties; prior info

\[ p(\theta|y) \propto p(y|\theta)p(\theta) \]
Bayesian inference

• Complex, modular; coherent uncertainties; prior info
  \[ p(\theta|y) \propto_{\theta} p(y|\theta)p(\theta) \]

• MCMC
Bayesian inference

- Complex, modular; coherent uncertainties; prior info
  \[ p(\theta|y) \propto_{\theta} p(y|\theta)p(\theta) \]

- MCMC: Accurate but can be slow  [Bardenet, Doucet, Holmes 2015]
Bayesian inference

- Complex, modular; coherent uncertainties; prior info
  \[ p(\theta|y) \propto p(y|\theta)p(\theta) \]
- MCMC: Accurate but can be slow [Bardenet, Doucet, Holmes 2015]
- (Mean-field) variational Bayes: (MF)VB
Bayesian inference

- Complex, modular; coherent uncertainties; prior info
  \[ p(\theta|y) \propto \theta p(y|\theta)p(\theta) \]
- MCMC: Accurate but can be slow [Bardenet, Doucet, Holmes 2015]
- (Mean-field) variational Bayes: (MF)VB
  - Fast
Bayesian inference

- Complex, modular; coherent uncertainties; prior info
  \[ p(\theta|y) \propto \theta \cdot p(y|\theta)p(\theta) \]
- MCMC: Accurate but can be slow \cite{Bardenet, Doucet, Holmes 2015}
- (Mean-field) variational Bayes: (MF)VB
  - Fast, streaming, distributed \cite{Broderick, Boyd, Wibisono, Wilson, Jordan 2013}
Bayesian inference

- Complex, modular; coherent uncertainties; prior info
  \[ p(\theta|y) \propto p(y|\theta)p(\theta) \]

- MCMC: Accurate but can be slow  [Bardenet, Doucet, Holmes 2015]

- (Mean-field) variational Bayes: (MF)VB

- Fast, streaming, distributed  [Broderick, Boyd, Wibisono, Wilson, Jordan 2013]

![Graphs showing run time vs. number of threads for Wikipedia (3.6M) and Nature (350K) datasets.](image)
Bayesian inference

- Complex, modular; coherent uncertainties; prior info
  \[ p(\theta|y) \propto p(y|\theta)p(\theta) \]

- MCMC: Accurate but can be slow  [Bardenet, Doucet, Holmes 2015]

- (Mean-field) variational Bayes: (MF)VB
  - Fast, streaming, distributed  [Broderick, Boyd, Wibisono, Wilson, Jordan 2013]

- Misestimation & lack of quality guarantees  
  [MacKay 2003; Bishop 2006; Wang, Titterington 2004; Turner, Sahani 2011]
Bayesian inference

• Complex, modular; coherent uncertainties; prior info
  \[ p(\theta|y) \propto p(y|\theta)p(\theta) \]

• MCMC: Accurate but can be slow [Bardenet, Doucet, Holmes 2015]

• (Mean-field) variational Bayes: (MF)VB
  • Fast, streaming, distributed [Broderick, Boyd, Wibisono, Wilson, Jordan 2013]

- [MacKay 2003; Bishop 2006; Wang, Titterington 2004; Turner, Sahani 2011; Fosdick 2013; Dunson 2014; Bardenet, Doucet, Holmes 2015]

- Wikipedia (3.6M)
- Nature (350K)

• Misestimation & lack of quality guarantees

Bayesian inference

- Complex, modular; coherent uncertainties; prior info
  \[ p(\theta|y) \propto \theta \cdot p(y|\theta) \cdot p(\theta) \]

- MCMC: Accurate but can be slow  [Bardenet, Doucet, Holmes 2015]

- (Mean-field) variational Bayes: (MF)VB
  - Fast, streaming, distributed  [Broderick, Boyd, Wibisono, Wilson, Jordan 2013]

- Misestimation & lack of quality guarantees
  [MacKay 2003; Bishop 2006; Wang, Titterington 2004; Turner, Sahani 2011; Fosdick 2013; Dunson 2014; Bardenet, Doucet, Holmes 2015; Opper, Winther 2003; Giordano, Broderick, Jordan 2015]

(c) Wikipedia (3.6M) (d) Nature (350K)
Bayesian inference

- Complex, modular; coherent uncertainties; prior info
  \[ p(\theta|y) \propto p(y|\theta)p(\theta) \]

- MCMC: Accurate but can be slow  [Bardenet, Doucet, Holmes 2015]

- (Mean-field) variational Bayes: (MF)VB
  - Fast, streaming, distributed  [Broderick, Boyd, Wibisono, Wilson, Jordan 2013]

- Misestimation & lack of quality guarantees
  [MacKay 2003; Bishop 2006; Wang, Titterington 2004; Turner, Sahani 2011; Fosdick 2013; Dunson 2014; Bardenet, Doucet, Holmes 2015; Opper, Winther 2003; Giordano, Broderick, Jordan 2015]

- Our proposal: use data summarization for fast, streaming, distributed algs. with theoretical guarantees
Data summarization
Data summarization

- Exponential family likelihood

\[
p(y_{1:N} | x_{1:N}, \theta) = \prod_{n=1}^{N} \exp \left[ T(y_n, x_n) \cdot \eta(\theta) \right]
\]
Data summarization

- Exponential family likelihood

\[
p(y_{1:N} | x_{1:N}, \theta) = \prod_{n=1}^{N} \exp \left[ T(y_n, x_n) \cdot \eta(\theta) \right]
\]
Data summarization

- Exponential family likelihood

\[ p(y_{1:N} \mid x_{1:N}, \theta) = \prod_{n=1}^{N} \exp \left[ T(y_n, x_n) \cdot \eta(\theta) \right] = \exp \left[ \sum_{n=1}^{N} T(y_n, x_n) \right] \cdot \eta(\theta) \]
Data summarization

- Exponential family likelihood

\[
p(y_{1:N} | x_{1:N}, \theta) = \prod_{n=1}^{N} \exp \left[ T(y_n, x_n) \cdot \eta(\theta) \right]
\]

= \exp \left[ \sum_{n=1}^{N} T(y_n, x_n) \right] \cdot \eta(\theta)

- Scalable, single-pass, streaming, distributed, complementary to MCMC
Data summarization

- Exponential family likelihood

\[
p(y_{1:N} | x_{1:N}, \theta) = \prod_{n=1}^{N} \exp \left[ T(y_n, x_n) \cdot \eta(\theta) \right]
\]

\[
= \exp \left[ \sum_{n=1}^{N} T(y_n, x_n) \right] \cdot \eta(\theta)
\]

- Scalable, single-pass, streaming, distributed, complementary to MCMC

- **But:** Often no simple sufficient statistics
Data summarization

- Exponential family likelihood

\[ p(y_{1:N} | x_{1:N}, \theta) = \prod_{n=1}^{N} \exp \left[ T(y_n, x_n) \cdot \eta(\theta) \right] \]

- Sufficient statistics

\[ = \exp \left[ \sum_{n=1}^{N} T(y_n, x_n) \right] \cdot \eta(\theta) \]

- Scalable, single-pass, streaming, distributed, complementary to MCMC

- *But*: Often no simple sufficient statistics

  - E.g. Bayesian logistic regression; GLMs; “deeper” models
Data summarization

- Exponential family likelihood

\[ p(y_{1:N} | x_{1:N}, \theta) = \prod_{n=1}^{N} \exp \left[ T(y_n, x_n) \cdot \eta(\theta) \right] \]

\[ = \exp \left[ \sum_{n=1}^{N} T(y_n, x_n) \right] \cdot \eta(\theta) \]

- Sufficient statistics

- Scalable, single-pass, streaming, distributed, complementary to MCMC

- But: Often no simple sufficient statistics

- E.g. Bayesian logistic regression; GLMs; “deeper” models

- Likelihood

\[ p(y_{1:N} | x_{1:N}, \theta) = \prod_{n=1}^{N} \frac{1}{1 + \exp(-y_n x_n \cdot \theta)} \]
Data summarization

- Exponential family likelihood

\[
p(y_{1:N} | x_{1:N}, \theta) = \prod_{n=1}^{N} \exp \left[ T(y_n, x_n) \cdot \eta(\theta) \right]
\]

- Sufficient statistics

\[
= \exp \left[ \sum_{n=1}^{N} T(y_n, x_n) \right] \cdot \eta(\theta)
\]

- Scalable, single-pass, streaming, distributed, complementary to MCMC

- **But**: Often no simple sufficient statistics
  - E.g. Bayesian logistic regression; GLMs; “deeper” models
    - Likelihood
      \[
p(y_{1:N} | x_{1:N}, \theta) = \prod_{n=1}^{N} \frac{1}{1 + \exp(-y_n x_n \cdot \theta)}
\]
  - Our proposal: *approximate* sufficient statistics
Baseball

Curling

[Agarwal et al 2005, Feldman & Langberg 2011]
Baseball

Curling
Coresets

[Agarwal et al 2005, Feldman & Langberg 2011]
Coresets

- Pre-process data to get a smaller, weighted data set

[Agarwal et al 2005, Feldman & Langberg 2011]
Coresets

- Pre-process data to get a smaller, weighted data set
- Theoretical guarantees on quality

[Agarwal et al 2005, Feldman & Langberg 2011]
Coresets

• Pre-process data to get a smaller, weighted data set
• Theoretical guarantees on quality
• Fast algorithms; error bounds for streaming, distributed

[Agarwal et al 2005, Feldman & Langberg 2011]
Coresets

- Pre-process data to get a smaller, weighted data set
- Theoretical guarantees on quality
- Fast algorithms; error bounds for streaming, distributed
- Cf. data squashing, big data GP ideas

Coresets

- Pre-process data to get a smaller, weighted data set
- Theoretical guarantees on quality
- Fast algorithms; error bounds for streaming, distributed
- Cf. data squashing, big data GP ideas, subsampling

Coresets

• Pre-process data to get a smaller, weighted data set
• Theoretical guarantees on quality
• Fast algorithms; error bounds for streaming, distributed
• Cf. data squashing, big data GP ideas, subsampling

Coresets

- Pre-process data to get a smaller, weighted data set
- Theoretical guarantees on quality
- Fast algorithms; error bounds for streaming, distributed
- Cf. data squashing, big data GP ideas, subsampling

Coresets

- Pre-process data to get a smaller, weighted data set
- Theoretical guarantees on quality
- Fast algorithms; error bounds for streaming, distributed
- Cf. data squashing, big data GP ideas, subsampling
- We develop: coresets for Bayes

Coresets

• Pre-process data to get a smaller, weighted data set
• Theoretical guarantees on quality
• Fast algorithms; error bounds for streaming, distributed
• Cf. data squashing, big data GP ideas, subsampling
• We develop: coresets for Bayes
  • Focus on: Logistic regression

Coresets

- Pre-process data to get a smaller, weighted data set
- Theoretical guarantees on quality
- Fast algorithms; error bounds for streaming, distributed
- Cf. data squashing, big data GP ideas, subsampling
- We develop: coresets for Bayes
  - Focus on: Logistic regression

Coresets

- Pre-process data to get a smaller, weighted data set
- Theoretical guarantees on quality
- Fast algorithms; error bounds for streaming, distributed
- Cf. data squashing, big data GP ideas, subsampling
- We develop: coresets for Bayes
- Focus on: Logistic regression

Coresets

- Pre-process data to get a smaller, weighted data set
- Theoretical guarantees on quality
- Fast algorithms; error bounds for streaming, distributed
- Cf. data squashing, big data GP ideas, subsampling
- We develop: coresets for Bayes
  - Focus on: Logistic regression

Coresets

- Pre-process data to get a smaller, weighted data set
- Theoretical guarantees on quality
- Fast algorithms; error bounds for streaming, distributed
- Cf. data squashing, big data GP ideas, subsampling
- We develop: coresets for Bayes
  - Focus on: Logistic regression

Step 1: calculate sensitivities of each datapoint
Step 1: calculate sensitivities of each datapoint
Step 2: sample points proportionally to sensitivity
Step 2: sample points proportionally to sensitivity
Step 3: weight points by inverse of their sensitivity

[Huggins, Campbell, Broderick 2016]
Step 3: weight points by inverse of their sensitivity
Step 4: input weighted points to existing approximate posterior algorithm
Step 4: input weighted points to existing approximate posterior algorithm
Step 4: input weighted points to existing approximate posterior algorithm

webspam
350K points
127 features

[Full MCMC]
Step 4: input weighted points to existing approximate posterior algorithm

Full \rightarrow MCMC \rightarrow > 2 \text{ days}

webspam
350K points
127 features

[Huggins, Campbell, Broderick 2016]
Step 4: input weighted points to existing approximate posterior algorithm

Full $\rightarrow$ MCMC $\rightarrow$ > 2 days

webspam 350K points 127 features

Coreset $\rightarrow$ MCMC $\rightarrow$ [Huggins, Campbell, Broderick 2016]
Step 4: input weighted points to existing approximate posterior algorithm

- Full MCMC: > 2 days
- Webspam: 350K points, 127 features: < 2 hours
- Coreset MCMC: Fast!

[Huggins, Campbell, Broderick 2016]
Theory
Theory

- Finite-data theoretical guarantee

Thm sketch (HCB). Choose $\varepsilon > 0$, $\delta \in (0, 1)$. Our algorithm runs in $O(N)$ time and creates a coreset-size $\sim \text{const} \cdot \varepsilon^{-2} + \log(1/\delta)$.

W.p. $1 - \delta$, it constructs a coreset with $|\ln \mathcal{E} - \ln \tilde{\mathcal{E}}| \leq \varepsilon |\ln \mathcal{E}|$. [Huggins, Campbell, Broderick 2016]
Theory

- Finite-data theoretical guarantee
  - On the log evidence (vs. posterior mean, uncertainty, etc)

Thm sketch (HCB). Choose $\varepsilon > 0$, $\delta \in (0,1)$. Our algorithm runs in $O(N)$ time and creates coreset-size $\sim \text{const} \cdot \varepsilon^{-2} + \log(1/\delta)$

W.p. $1 - \delta$, it constructs a coreset with $|\ln \mathcal{E} - \ln \tilde{\mathcal{E}}| \leq \varepsilon |\ln \mathcal{E}|$

[Huggins, Campbell, Broderick 2016]
Theory

- Finite-data theoretical guarantee
- On the log evidence (vs. posterior mean, uncertainty, etc)

Thm sketch (HCB). Choose $\varepsilon > 0$, $\delta \in (0,1)$. Our algorithm runs in $O(N)$ time and creates $\text{coreset-size} \sim \text{const} \cdot \varepsilon^{-2} + \log(1/\delta)$

W.p. $1 - \delta$, it constructs a coreset with $\left| \ln \mathcal{E} - \ln \tilde{\mathcal{E}} \right| \leq \varepsilon |\ln \mathcal{E}|$

- Can quantify the propagation of error in streaming and parallel settings

1. If $D_i'$ is an $\varepsilon$-coreset for $D_i$, then $D_1' \cup D_2'$ is an $\varepsilon$-coreset for $D_1 \cup D_2$.

2. If $D'$ is an $\varepsilon$-coreset for $D$ and $D''$ is an $\varepsilon'$-coreset for $D'$, then $D''$ is an $\varepsilon''$-coreset for $D$, where $\varepsilon'' = (1 + \varepsilon)(1 + \varepsilon') - 1$.

[1] [Huggins, Campbell, Broderick 2016]
Criteo Releases Industry’s Largest-Ever Dataset for Machine Learning to Academic Community

Over one terabyte of data released to help researchers benchmark distributed learning algorithms in critical research
• Subset yields 6M data points, 1K features
Polynomial approximate sufficient statistics

- Subset yields 6M data points, 1K features

Criteo Releases Industry’s Largest-Ever Dataset for Machine Learning to Academic Community

Over one terabyte of data released to help researchers benchmark distributed learning algorithms in critical research
Polynomial approximate sufficient statistics

- Subset yields 6M data points, 1K features

Criteo Releases Industry’s Largest-Ever Dataset for Machine Learning to Academic Community

Over one terabyte of data released to help researchers benchmark distributed learning algorithms in critical research

[Figure: Graph showing time (sec) on the y-axis against cores on the x-axis, with a decreasing curve indicating lower time with more cores.]

[Huggins, Adams, Broderick, submitted]
Polynomial approximate sufficient statistics

• Subset yields 6M data points, 1K features
• Streaming, distributed; minimal communication
Polynomial approximate sufficient statistics

- Subset yields 6M data points, 1K features
- Streaming, distributed; minimal communication
- 24 cores, <20 sec

[Huggins, Adams, Broderick, submitted]
Polynomial approximate sufficient statistics

• Subset yields 6M data points, 1K features
• Streaming, distributed; minimal communication
• 24 cores, <20 sec
• Bounds on Wasserstein

Criteo Releases Industry’s Largest-Ever Dataset for Machine Learning to Academic Community

Over one terabyte of data released to help researchers benchmark distributed learning algorithms in critical research

[Huggins, Adams, Broderick, submitted]
Conclusions

• **Reliable** Bayesian inference at scale via data summarization
  • Coresets, polynomial approximate sufficient statistics
  • Streaming, distributed

• Challenges and opportunities:
  • Beyond logistic regression
  • Generalized linear models; deep models; high-dimensional models

[Huggins, Campbell, Broderick 2016; Huggins, Adams, Broderick, submitted; Bardenet, Maillard 2015; Geppert, Ickstadt, Munteanu, Quedenfeld, Sohler 2017; Ahfock, Astle, Richardson 2017]
References


**JH Huggins, T Campbell, and T Broderick. Coresets for scalable Bayesian logistic regression. *NIPS* 2016.**
References


• D Dunson. Robust and scalable approach to Bayesian inference. Talk at ISBA 2014.


