Min-wise hashing for large-scale regression

Rajen D. Shah (Statistical Laboratory, University of Cambridge)
joint work with Nicolai Meinshausen (Seminar für Statistik, ETH Zürich)

Theory of Big Data, UCL
8 January 2015
High-dimensional data: large number $p$ of predictors, small number $n$ of observations.

Main challenge is a statistical one: how can we obtain accurate estimates with so few observations?
High-dimensional data: large number $p$ of predictors, small number $n$ of observations.

Main challenge is a statistical one: how can we obtain accurate estimates with so few observations?

Large-scale data: large $p$, large $n$.

Data is not the only relevant resource to consider. The computational budget is also an issue (both memory and computing power).
Text analysis. Given a collection of documents, construct variables which count the number of occurrences of different words. Can add variables giving the frequency of pairs of words (bigrams) or triples of words (trigrams). An example from (Kogan, 2009) contains 4,272,227 predictor variables for $n = 16,087$ documents.

URL reputation scoring (Ma et al., 2009). Information about a URL comprises $> 3$ million variables which include word-stem presence and geographical information, for example. Number of observations, $n > 2$ million.
Text analysis. Given a collection of documents, construct variables which count the number of occurrences of different words. Can add variables giving the frequency of pairs of words (bigrams) or triples of words (trigrams). An example from (Kogan, 2009) contains 4,272,227 predictor variables for \( n = 16,087 \) documents.

URL reputation scoring (Ma et al., 2009). Information about a URL comprises \( > 3 \) million variables which include word-stem presence and geographical information, for example. Number of observations, \( n > 2 \) million.

In both of these examples, the design matrices are sparse.
**Text analysis.** Given a collection of documents, construct variables which count the number of occurrences of different words. Can add variables giving the frequency of pairs of words (bigrams) or triples of words (trigrams). An example from (Kogan, 2009) contains 4,272,227 predictor variables for $n = 16,087$ documents.

**URL reputation scoring** (Ma et al., 2009). Information about a URL comprises $>3$ million variables which include word-stem presence and geographical information, for example. Number of observations, $n > 2$ million.

- In both of these examples, the design matrices are *sparse*.
- This is very different from sparsity in the signal, which is the typical assumption in high-dimensional statistics.
Our computational budget may mean that OLS and Ridge regression are computationally infeasible.
Our computational budget may mean that OLS and Ridge regression are computationally infeasible.

The computer science literature has a variety of algorithms to form a low-dimensional “sketch” of the design matrix i.e. a mapping

\[ X \mapsto S \]

\[ n \times p \quad n \times L, \quad L \ll p. \]
Our computational budget may mean that OLS and Ridge regression are computationally infeasible.

The computer science literature has a variety of algorithms to form a low-dimensional “sketch” of the design matrix i.e. a mapping

$$X \mapsto S$$

$$n \times p \quad n \times L, \quad L \ll p.$$ 

The idea is then to perform the regression on $S$ rather than the larger $X$. 

Linear model with sparse design

Target $Y \in \mathbb{R}^n$

Sparse $X \in \mathbb{R}^{n \times p}$

$\beta^* \in \mathbb{R}^p$

Noise $\epsilon \in \mathbb{R}^n$

Non-zero entries are marked with $\ast$. 
Can we safely reduce our sparse $p$-dimensional problem to a (possibly dense) $L$-dimensional one with $L \ll p$?

The approach we take here is based on *min-wise hashing* (Broder, 1997), and more specifically a variant called *b-bit min-wise hashing* (Li and König, 2011).
Can we safely reduce our sparse $p$-dimensional problem to a (possibly dense) $L$-dimensional one with $L \ll p$?

\[
\text{sparse } \mathbf{X} \in \mathbb{R}^{n \times p}
\approx
\mathbf{S} \in \mathbb{R}^{n \times L}
\]

- The approach we take here is based on \textit{min-wise hashing} (Broder, 1997), and more specifically a variant called \textit{b-bit min-wise hashing} (Li and König, 2011).
- Note PCA may seem like as obvious choice, but may be too computationally expensive to compute.
Min-wise hashing matrix $\mathbf{M}$

$b$-bit min-wise hashing begins with the creation of a min-wise hashing matrix $\mathbf{M}$.

$$
\begin{pmatrix}
1 & 2 & 3 & 4 \\
1 & 1 & 1 \\
1 & 1 \\
1 & 1
\end{pmatrix}
$$
Min-wise hashing matrix $\mathbf{M}$

$b$-bit min-wise hashing begins with the creation of a min-wise hashing matrix $\mathbf{M}$.

\[
\pi_1 \quad 2 \quad 3 \quad 1 \quad 4
\]

\[
\mathbf{X} = \begin{pmatrix}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1
\end{pmatrix} \quad \mapsto \quad \mathbf{M} = \begin{pmatrix}
1 \\
2 \\
2 \\
1 \\
1
\end{pmatrix}
\]

One column of $\mathbf{M}$ generated by the random permutation $\pi_1$ of the variables.
Min-wise hashing matrix $\mathbf{M}$

$$\mathbf{X} = \begin{pmatrix} \pi_2 & 2 & 4 & 1 & 3 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \rightarrow \mathbf{M} = \begin{pmatrix} 1 & 3 \\ 2 & 1 \\ 2 & 1 \\ 1 & 1 \\ 1 & 2 \end{pmatrix}$$
$b$-bit min-wise hashing (Li and König, 2011)

$b$-bit min-wise hashing stores the lowest $b$ bits of each entry of $\mathbf{M}$ when expressed in binary (i.e. the residue mod 2). In the case where $b = 1$, 

$$M_{il}^{(1)} \equiv M_{il} \pmod{2}.$$
$b$-bit min-wise hashing (Li and König, 2011)

$b$-bit min-wise hashing stores the lowest $b$ bits of each entry of $\mathbf{M}$ when expressed in binary (i.e. the residue mod 2). In the case where $b = 1$, $M_{il}^{(1)} \equiv M_{il} \pmod{2}$.

\[
\mathbf{X} = \begin{pmatrix}
1 & 1 \\
1 & 1 \\
1 & 1 \\
1 & 1
\end{pmatrix}
\quad \mapsto \quad
\mathbf{M} = \begin{pmatrix}
1 & 3 \\
2 & 1 \\
2 & 1 \\
1 & 2
\end{pmatrix}
\quad \mapsto \quad
\mathbf{M}^{(1)} = \begin{pmatrix}
1 & 1 \\
0 & 1 \\
0 & 1 \\
1 & 1
\end{pmatrix}
\]

Perform regression using binary $n \times L$ matrix $\mathbf{M}^{(1)}$ rather than $\mathbf{X}$. When $L \ll p$ this gives large computational savings, and empirical studies report good performance (mostly for classification with SVMs).
$b$-bit min-wise hashing (Li and König, 2011)

$b$-bit min-wise hashing stores the lowest $b$ bits of each entry of $\mathbf{M}$ when expressed in binary (i.e. the residue mod 2). In the case where $b = 1$,

$$M_{il}^{(1)} \equiv M_{il} \pmod{2}.$$ 

\[
\mathbf{X} = \begin{pmatrix}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1
\end{pmatrix} \quad \mapsto \quad \mathbf{M} = \begin{pmatrix}
1 & 3 \\
2 & 1 \\
2 & 1 \\
1 & 1
\end{pmatrix} \quad \mapsto \quad \mathbf{M}^{(1)} = \begin{pmatrix}
1 & 1 \\
0 & 1 \\
0 & 1 \\
1 & 1 \\
1 & 0
\end{pmatrix}
\]

- Perform regression using binary $n \times L$ matrix $\mathbf{M}^{(1)}$ rather than $\mathbf{X}$.
- When $L \ll p$ this gives large computational savings, and empirical studies report good performance (mostly for classification with SVMs).
A procedure that is closely related to 1-bit min-wise hashing is what we call *random sign min-wise hashing*.

- Easier to analyse.
- Deals with sparse design matrices with real-valued entries.
- Allows for the construction of a variable importance measure.
Random sign min-wise hashing

1-bit min-wise hashing: keep last bit

\[
X = \begin{pmatrix}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{pmatrix} \quad \mapsto \quad M = \begin{pmatrix}
1 & 3 \\
2 & 1 \\
2 & 1 \\
1 & 1 \\
1 & 2 \\
\end{pmatrix} \quad \mapsto \quad M^{(1)} = \begin{pmatrix}
1 & 1 \\
0 & 1 \\
0 & 1 \\
1 & 1 \\
1 & 0 \\
\end{pmatrix}
\]
Random sign min-wise hashing

1-bit min-wise hashing: keep last bit

\[
X = \begin{pmatrix}
1 & 1 \\
1 & 1 \\
1 & 1 \\
1 & 1
\end{pmatrix} \mapsto M = \begin{pmatrix}
1 & 3 \\
2 & 1 \\
2 & 1 \\
1 & 1 \\
1 & 2
\end{pmatrix} \mapsto M^{(1)} = \begin{pmatrix}
1 & 1 \\
0 & 1 \\
0 & 1 \\
1 & 1 \\
1 & 0
\end{pmatrix}
\]

Random sign min-wise hashing: random sign assignments

\{1, \ldots, p\} \rightarrow \{-1, 1\} are chosen independently for all columns

\[l = 1, \ldots, L\] when going from \(M_l\) to \(S_l\).

\[
X = \begin{pmatrix}
1 & 1 \\
1 & 1 \\
1 & 1 \\
1 & 1
\end{pmatrix} \mapsto M = \begin{pmatrix}
1 & 3 \\
2 & 1 \\
2 & 1 \\
1 & 1 \\
1 & 2
\end{pmatrix} \mapsto S = \begin{pmatrix}
1 & -1 \\
-1 & -1 \\
-1 & -1 \\
1 & -1 \\
1 & 1
\end{pmatrix}
\]
The $\mathbf{H}$ and $\mathbf{M}$ matrices

Rather than storing $\mathbf{M}$, we can store the “responsible” variables in $\mathbf{H}$

$$M_{il} = \min_{k \in z_i} \pi_l(k)$$

$$H_{il} = \arg \min_{k \in z_i} \pi_l(k)$$

$$\mathbf{X} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \quad \mapsto \quad \mathbf{M} = \begin{pmatrix} 1 & 3 \\ 2 & 1 \\ 2 & 1 \\ 1 & 1 \end{pmatrix} \quad \mapsto \quad \mathbf{S} = \begin{pmatrix} 1 & -1 \\ -1 & -1 \\ -1 & -1 \\ 1 & -1 \end{pmatrix}$$

$$\mathbf{X} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \quad \mapsto \quad \mathbf{H} = \begin{pmatrix} 2 & 4 \\ 3 & 3 \\ 3 & 3 \\ 2 & 3 \end{pmatrix} \quad \mapsto \quad \mathbf{S} = \begin{pmatrix} 1 & -1 \\ -1 & -1 \\ -1 & -1 \\ 1 & -1 \end{pmatrix}$$
Continuous variables

By using $\mathbf{H}$ rather than $\mathbf{M}$, we can handle continuous variables.

$$
\mathbf{X} = \begin{pmatrix}
1 & 1 & 1 \\
4.2 & 1 \\
1 & 1 \\
7.1 & 1
\end{pmatrix}
\quad \mapsto \quad
\mathbf{H} = \begin{pmatrix}
2 & 4 \\
3 & 3 \\
3 & 3 \\
2 & 3
\end{pmatrix}
\quad \mapsto \quad
\mathbf{S} = \begin{pmatrix}
1 & -1 \\
-4.2 & -4.2 \\
-1 & -1 \\
1 & 7.1
\end{pmatrix}
$$

We get $n \times L$ matrices $\mathbf{H}$, and $\mathbf{S}$ given by

$$
H_{il} = \arg \min_{k \in \mathbb{Z}_i} \pi_l(k)
$$

$$
S_{il} = \psi_{H_{il}} X_{iH_{il}},
$$

where $\psi_{hl}$ is the random sign of the $h^{th}$ variable in the $l^{th}$ permutation.
When $\mathbf{X}$ is binary, $S_{il} = \Psi_{H_{il}}$.

Take $i \neq j$. Write $z_i = \{k : X_{ik} = 1\}$.

$$
\mathbb{E}(S_{il}S_{jl}) = \mathbb{E}(S_{il}S_{jl}|H_{il} = H_{jl})\mathbb{P}(H_{il} = H_{jl})
+ \mathbb{E}(S_{il}S_{jl}|H_{il} \neq H_{jl})\mathbb{P}(H_{il} \neq H_{jl})
= \mathbb{P}(H_{il} = H_{jl})
= \frac{|z_i \cap z_j|}{|z_i \cup z_j|}
$$

The final quantity is known as the Jaccard similarity between $z_i$ and $z_j$. 

When $\mathbf{X}$ is binary, $S_{il} = \Psi_{H_{il}}$.

Take $i \neq j$. Write $z_i = \{k : X_{ik} = 1\}$.

$$E(S_{il}S_{jl}) = E(S_{il}S_{jl} | H_{il} = H_{jl})P(H_{il} = H_{jl})$$
$$+ E(S_{il}S_{jl} | H_{il} \neq H_{jl})P(H_{il} \neq H_{jl})$$
$$= P(H_{il} = H_{jl})$$
$$= \frac{|z_i \cap z_j|}{|z_i \cup z_j|}$$

The final quantity is known as the Jaccard similarity between $z_i$ and $z_j$. 
Random sign min-wise hashing and Jaccard similarity

- When $\mathbf{X}$ is binary, $S_{il} = \Psi_{H_{il}}$.
- Take $i \neq j$. Write $z_i = \{k : X_{ik} = 1\}$.

\[
\mathbb{E}(S_{il}S_{jl}) = \mathbb{E}(S_{il}S_{jl}|H_{il} = H_{jl})\mathbb{P}(H_{il} = H_{jl})
+ \mathbb{E}(S_{il}S_{jl}|H_{il} \neq H_{jl})\mathbb{P}(H_{il} \neq H_{jl})
= \mathbb{P}(H_{il} = H_{jl})
= \frac{|z_i \cap z_j|}{|z_i \cup z_j|}
\]

- The final quantity is known as the Jaccard similarity between $z_i$ and $z_j$. 
When $\mathbf{X}$ is binary, $S_{il} = \Psi_{H_{il}}$.

Take $i \neq j$. Write $z_i = \{k : X_{ik} = 1\}$.

\[
\mathbb{E}(S_{il}S_{jl}) = \mathbb{E}(S_{il}S_{jl} | H_{il} = H_{jl}) \mathbb{P}(H_{il} = H_{jl}) \\
+ \mathbb{E}(S_{il}S_{jl} | H_{il} \neq H_{jl}) \mathbb{P}(H_{il} \neq H_{jl}) \\
= \mathbb{P}(H_{il} = H_{jl}) \\
= \frac{|z_i \cap z_j|}{|z_i \cup z_j|}
\]

The final quantity is known as the Jaccard similarity between $z_i$ and $z_j$. 

Rajen Shah (Cambridge) Large-scale regression 8 January 2015 15 / 36
Consider performing ridge regression using $S$ with regularisation parameter $L\lambda$. The fitted values are

$$S(L\lambda I + SS^T)^{-1}S^TY = SS^T(L\lambda I + SS^T)^{-1}Y$$

$$= \frac{1}{L}SS^T(\lambda I + \frac{1}{L}SS^T)^{-1}Y.$$
Consider performing ridge regression using $S$ with regularisation parameter $L\lambda$. The fitted values are

$$S(L\lambda I + SS^T)^{-1}S^TY = SS^T(L\lambda I + SS^T)^{-1}Y$$
$$= \frac{1}{L}SS^T(\lambda I + \frac{1}{L}SS^T)^{-1}Y.$$ 

As $L \to \infty$, the fitted values tend to

$$J(\lambda I + J)^{-1}Y.$$
Consider performing ridge regression using $S$ with regularisation parameter $L\lambda$. The fitted values are

$$S(L\lambda I + S^T S)^{-1}S^T Y = SS^T (L\lambda I + SS^T)^{-1}Y = \frac{1}{L} SS^T (\lambda I + \frac{1}{L} SS^T)^{-1}Y.$$

As $L \to \infty$, the fitted values tend to

$$J(\lambda I + J)^{-1}Y.$$

The fitted values from ridge regression on the original $X$ are

$$XX^T (\lambda I + XX^T)^{-1}Y.$$
Consider performing ridge regression using $S$ with regularisation parameter $L\lambda$. The fitted values are

$$S(L\lambda I + S^T S)^{-1}S^T Y = SS^T(L\lambda I + SS^T)^{-1}Y$$

$$= \frac{1}{L}SS^T(\lambda I + \frac{1}{L}SS^T)^{-1}Y.$$ 

As $L \to \infty$, the fitted values tend to

$$J(\lambda I + J)^{-1}Y.$$ 

The fitted values from ridge regression on the original $X$ are

$$XX^T(\lambda I + XX^T)^{-1}Y.$$ 

Ridge regression using $S$ approximates a kernel ridge regression using the Jaccard similarity kernel.
Consider performing ridge regression using $S$ with regularisation parameter $L\lambda$. The fitted values are

$$S(L\lambda I + SS^T)^{-1}S^TY = SS^T(L\lambda I + SS^T)^{-1}Y = \frac{1}{L}SS^T(\lambda I + \frac{1}{L}SS^T)^{-1}Y.$$ 

As $L \to \infty$, the fitted values tend to

$$J(\lambda I + J)^{-1}Y.$$ 

The fitted values from ridge regression on the original $X$ are

$$XX^T(\lambda I + XX^T)^{-1}Y.$$ 

Ridge regression using $S$ approximates a kernel ridge regression using the Jaccard similarity kernel.
Can we construct a $\mathbf{b}^* \in \mathbb{R}^L$ such that $\mathbf{X}\mathbf{\beta}^*$ is close to $\mathbf{S}\mathbf{b}^*$ on average?

\begin{align*}
\text{sparse } \mathbf{X} & \in \mathbb{R}^{n \times p} \\
& \begin{pmatrix}
* & * & * \\
* & * & * \\
* & * & * \\
* & * & *
\end{pmatrix} \\
\beta^* & \in \mathbb{R}^p \\
& \begin{pmatrix}
* \\
* \\
* \\
*
\end{pmatrix} \\
\text{dense } \mathbf{S} & \in \mathbb{R}^{n \times L} \\
& \begin{pmatrix}
* & * & * & * \\
* & * & * & * \\
* & * & * & * \\
* & * & * & *
\end{pmatrix} \\
\mathbf{b}^* & \in \mathbb{R}^L \\
& \begin{pmatrix}
* \\
* \\
* \\
*
\end{pmatrix}
\end{align*}
Approximation error

Is there a $b^*$ such that the expected value is unbiased (if averaged over the random permutations and sign assignments)?

\[
\begin{align*}
\text{sparse } X &\in \mathbb{R}^{n \times p} \\
\beta^* &\in \mathbb{R}^p \\
S &\in \mathbb{R}^{n \times 1} \\
b^* &\in \mathbb{R}^1
\end{align*}
\]

\[
\mathbb{E}_{\pi,\psi}
\]
Approximation error

- Assume for now that there are $q \leq p$ non-zero entries in each row of $X$. Unequal row sparsity can also be dealt with.
Assume for now that there are $q \leq p$ non-zero entries in each row of $X$. Unequal row sparsity can also be dealt with.

Consider one permutation with min-hash value $H_i$ for $i = 1, \ldots, n$ and random signs $\psi_k$, $k = 1, \ldots, p$.

$$
\begin{bmatrix}
S \in \mathbb{R}^{n \times 1} \\
\begin{pmatrix}
\psi_{H_1} X_{1H_1} \\
\psi_{H_2} X_{2H_2} \\
\vdots \\
\psi_{H_p} X_{pH_p} \\
\vdots \\
\end{pmatrix}
\end{bmatrix} =: b^{\ast} \in \mathbb{R}^1
$$

$$
= q \sum_{k=1}^{p} \beta_k^{\ast} \psi_k
$$
Approximation error

Can we find a $b^* \in \mathbb{R}^L$ such that $X\beta^*$ is close to $Sb^*$ on average?

\[
\mathbb{E}_{\pi, \psi} \left[ \begin{pmatrix} \psi_{H_1} X_{1H_1} \\ \psi_{H_2} X_{2H_2} \\ \vdots \\ \vdots \\ \vdots \\ S \end{pmatrix} \right] = \begin{pmatrix} \sum_{k=1}^{p} X_{1k} \beta^*_k q \mathbb{P}(H_1 = k) \\ \sum_{k=1}^{p} X_{2k} \beta^*_k q \mathbb{P}(H_2 = k) \\ \vdots \\ \vdots \end{pmatrix} =: b^*
\]
Can we find a $b^* \in \mathbb{R}^L$ such that $X\beta^*$ is close to $Sb^*$ on average?

\[
\mathbb{E}_{\pi,\psi} \left[ \begin{pmatrix} \psi_{H_1} X_{1H_1} \\ \psi_{H_2} X_{2H_2} \\ \vdots \\ \vdots \\ \vdots \\ S \end{pmatrix} \left( q \sum_{k=1}^{p} \beta_k^* \psi_k \right) \right] = \begin{pmatrix} \sum_{k=1}^{p} X_{1k} \beta_k^* q \mathbb{P}(H_1 = k) \\ \sum_{k=1}^{p} X_{2k} \beta_k^* q \mathbb{P}(H_2 = k) \\ \vdots \\ \vdots \\ \vdots \\ \sum_{k=1}^{p} X_{pk} \beta_k^* q \mathbb{P}(H_p = k) \end{pmatrix} = X\beta^* \text{ (unbiased)}. \]
Theorem

Let $\mathbf{b}^* \in \mathbb{R}^L$ be defined by

$$b^*_l = \frac{q}{L} \sum_{k=1}^{p} \beta^*_k \psi_{kl} w_{\pi_l(k)},$$

where $w$ is a vector of weights. Then there is a choice of $w$, such that:

(i) The approximation is unbiased: $E_{\pi, \psi}(S\mathbf{b}^*) = \mathbf{X}\beta^*$.

(ii) $E_{\pi, \psi}(\|\mathbf{b}^*\|_2^2) \leq 2q\|\beta^*\|_2^2 / L$.

(iii) If $\|\mathbf{X}\|_\infty \leq 1$, then $\frac{1}{n} E_{\pi, \psi}(\|S\mathbf{b}^* - \mathbf{X}\beta^*\|_2^2) \leq 2q\|\beta^*\|_2^2 / L$. 
Assume model

\[ Y = \alpha^* 1 + X\beta^* + \varepsilon. \]

Random noise \( \varepsilon \in \mathbb{R}^n \) satisfies \( \mathbb{E}(\varepsilon_i) = 0, \mathbb{E}(\varepsilon_i^2) = \sigma^2 \) and \( \text{Cov}(\varepsilon_i, \varepsilon_j) = 0 \) for \( i \neq j \).
Linear model

- Assume model
  \[ Y = \alpha^* \mathbf{1} + X\beta^* + \varepsilon. \]

- Random noise \( \varepsilon \in \mathbb{R}^n \) satisfies \( \mathbb{E}(\varepsilon_i) = 0, \mathbb{E}(\varepsilon_i^2) = \sigma^2 \) and \( \text{Cov}(\varepsilon_i, \varepsilon_j) = 0 \) for \( i \neq j \).

- We give bounds on a mean-squared prediction error (MSPE) of the form
  \[
  \text{MSPE}((\hat{\alpha}, \hat{b})) := \mathbb{E}_{\varepsilon, \pi, \psi}\left( \|\alpha^* \mathbf{1} + X\beta^* - \hat{\alpha} \mathbf{1} - S\hat{b}\|_2^2 \right) / n.
  \]
Ordinary least squares

**Theorem**

Let \((\hat{\alpha}, \hat{b})\) be the least squares estimator and let \(L^* = \sqrt{2qn\|\beta^*\|_2}/\sigma\). We have

\[
\text{MSPE}((\hat{\alpha}, \hat{b})) \leq 2 \max \left\{ \frac{L}{L^*}, \frac{L^*}{L} \right\} \sigma \sqrt{\frac{2q}{n}} \|\beta^*\|_2 + \frac{\sigma^2}{n}.
\]
Ordinary least squares

**Theorem**

Let \((\hat{\alpha}, \hat{b})\) be the least squares estimator and let \(L^* = \sqrt{2qn}\|\beta^*\|_2/\sigma\). We have

\[
\text{MSPE}((\hat{\alpha}, \hat{b})) \leq 2 \max \left\{ \frac{L}{L^*}, \frac{L^*}{L} \right\} \sigma \sqrt{\frac{2q}{n}} \|\beta^*\|_2 + \frac{\sigma^2}{n}.
\]

- Imagine a situation where more predictors are added to the design matrix but their associated coefficients are all 0, so \(\|\beta^*\|_2 = O(1)\). The MSPE only increases like \(\sqrt{q}\) compared to the factor of \(p\) we would see if OLS were used.

- Additionally assume \(n = O(q)\) (ensures the MSPE is bounded asymptotically). Then \(L^* = O(q)\). This could be a substantial reduction over \(p\).
Ridge regression

Define

\[(\hat{\alpha}, \hat{b}^\eta) := \arg \min_{(\alpha, b) \in \mathbb{R} \times \mathbb{R}^p} \| Y - \alpha 1 - Sb \|^2_2 \text{ such that } \|b\|^2 \leq (1 + \eta) \frac{2q\|\beta^*\|^2}{L}. \]

**Theorem**

Let

\[\rho := \exp \left( - \frac{L\eta^2}{36q(36 + \eta)} \right).\]

Then

\[\text{MSPE}((\hat{\alpha}, \hat{b}^\eta)) \leq \sqrt{2q}\|\beta^*\|_2 \left( \frac{2\sigma \sqrt{1 + \eta + (L^*/L)}}{\sqrt{n}} \right) + \frac{\sigma^2}{n} + \rho \frac{\|X\beta^*\|^2}{n}. \]

- Similar result for logistic regression available.
Interaction models

Let \( f^* \in \mathbb{R}^n \) be given by

\[
f_i^* = \sum_{k=1}^{p} X_{ik} \theta_k^{*,(1)} + \sum_{k,k_1=1}^{p} X_{ik} \mathbb{1}_{\{X_{ik_1} = 0\}} \Theta_{k,k_1}^{*,(2)}, \quad i = 1, \ldots, n.
\]
Interaction models

Let $f^* \in \mathbb{R}^n$ be given by

$$f_i^* = \sum_{k=1}^{p} X_{ik} \theta_i^{*,(1)} + \sum_{k,k_1=1}^{p} X_{ik} \mathbb{1}\{X_{ik_1} = 0\} \Theta_{k,k_1}^{*,(2)}, \quad i = 1, \ldots, n.$$ 

Assume $\|X\|_{\infty} \leq 1$. Previous results hold if $\|\beta^*\|_2$ is replaced by

$$\ell(\Theta^*) := \|\theta_i^{*,(1)}\|_2 + 2 \left( q \sum_{k,k_1,k_2} \left| \Theta_{kk_1}^{*,(2)} \Theta_{kk_2}^{*,(2)} \right| \right)^{1/2}.$$ 

**Theorem**

*There exists $b^* \in \mathbb{R}^L$ such that*

(i) $E_{\pi,\psi}(Sb^*) = f^*$;

(ii) $E_{\pi,\psi}(\|Sb^* - f^*\|_2^2)/n \leq 2q\ell^2(\Theta^*)/L.$
Interaction models

Let $f^* \in \mathbb{R}^n$ be given by

$$f_i^* = \sum_{k=1}^{p} X_{ik} \theta_k^{*,(1)} + \sum_{k,k_1=1}^{p} X_{ik} \mathbb{1}\{X_{ik_1}=0\} \Theta_{k,k_1}^{*,(2)}, \quad i = 1, \ldots, n.$$ 

Assume $\|X\|_\infty \leq 1$. Previous results hold if $\|\beta^*\|_2$ is replaced by

$$\ell(\Theta^*) := \|\theta_k^{*,(1)}\|_2 + 2 \left(q \sum_{k,k_1, k_2} \left| \Theta_{k,k_1}^{*,(2)} \Theta_{k,k_2}^{*,(2)} \right| \right)^{1/2}.$$ 

Theorem

There exists $b^* \in \mathbb{R}^L$ such that

(i) $\mathbb{E}_{\pi, \psi}(Sb^*) = f^*$;

(ii) $\mathbb{E}_{\pi, \psi}(\|Sb^* - f^*\|_2^2)/n \leq 2q\ell^2(\Theta^*)/L.$

If there are a finite number of non-zero interaction terms with finite value, the approximation error becomes very small if $L \gg q^2$. 

Rajen Shah (Cambridge)

Large-scale regression

8 January 2015 26 / 36
Assume the linear model from before, but with $X\beta^*$ replaced by $f^*$.

**Theorem**

Let $\hat{b}$ be the least squares estimator and let $L^* = \sqrt{2qn} \ell(\Theta^*)/\sigma$. We have

\[
\text{MSPE}(\hat{b}) \leq 2 \max\left\{ \frac{L}{L^*}, \frac{L^*}{L} \right\} \sigma \sqrt{\frac{2q}{n}} \ell(\Theta^*) + \frac{\sigma^2}{n}.
\]

Consider a situation where there are a fixed number of interaction and main effects of fixed size, so $\ell(\Theta^*) = O(\sqrt{q})$. 
Assume the linear model from before, but with $\mathbf{X}\beta^*$ replaced by $\mathbf{f}^*$.

**Theorem**

Let $\hat{\mathbf{b}}$ be the least squares estimator and let $L^* = \sqrt{2qn} \ell(\Theta^*)/\sigma$. We have

$$\text{MSPE}(\hat{\mathbf{b}}) \leq 2 \max \left\{ \frac{L}{L^*}, \frac{L^*}{L} \right\} \sigma \frac{2q}{\sqrt{n}} \ell(\Theta^*) + \frac{\sigma^2}{n}.$$

- Consider a situation where there are a fixed number of interaction and main effects of fixed size, so $\ell(\Theta^*) = O(\sqrt{q})$.
- If $n, q$ and $p$ increase by collecting new data and adding uninformative variables, then in order for the MSPE to vanish asymptotically, we require $q^2/n \to 0$. 
Variable importance

Predicted values are

\[ \hat{f} = S\hat{b} \]

Let \( \hat{f}^{-(k)} \) be the predictions obtained when setting \( X_k = 0 \). If the underlying model is linear and contains only main effects,

\[ \hat{f} - \hat{f}^{-(k)} \approx X_k \beta_k^*. \]

Construct \( \tilde{S} \) in exactly the same way as \( S \) but using a matrix \( \tilde{H} \) rather than \( H \), with \( \tilde{H} \) defined by

\[ \tilde{H}_{il} := \arg \min_{k \in z_i \setminus H_{il}} \pi_l(k). \]
Variable importance

Predicted values are

$$\hat{f} = S\hat{b}$$

Let $\hat{f}^{-(k)}$ be the predictions obtained when setting $X_k = 0$. If the underlying model is linear and contains only main effects,

$$\hat{f} - \hat{f}^{-(k)} \approx X_k \beta_k^*.$$ 

Construct $\tilde{S}$ in exactly the same way as $S$ but using a matrix $\tilde{H}$ rather than $H$, with $\tilde{H}$ defined by

$$\tilde{H}_{il} := \arg \min_{k \in z_i \setminus H_{il}} \pi_l(k).$$

Store $n \times L$ matrices $S, \tilde{S}$ and $H$. Then

$$\hat{f}_i - \hat{f}_i^{-(k)} = \sum_{l=1}^{L} (S_{il} - \tilde{S}_{il}) 1\{H_{il}=k\} \hat{b}_l.$$
Variable importance

Predicted values are
\[ \hat{f} = S \hat{b} \]

Let \( \hat{f}^{-k} \) be the predictions obtained when setting \( X_k = 0 \). If the underlying model is linear and contains only main effects,
\[ \hat{f} - \hat{f}^{-k} \approx X_k \beta_k^*. \]

Construct \( \tilde{S} \) in exactly the same way as \( S \) but using a matrix \( \tilde{H} \) rather than \( H \), with \( \tilde{H} \) defined by
\[ \tilde{H}_{il} := \arg \min_{k \in z \setminus H_{il}} \pi_l(k). \]

Store \( n \times L \) matrices \( S, \tilde{S} \) and \( H \). Then
\[ \hat{f}_i - \hat{f}_i^{-k} = \sum_{l=1}^{L} (S_{il} - \tilde{S}_{il}) 1\{H_{il}=k\} \hat{b}_l. \]
The compressed design matrix $\mathbf{S}$ is generated in a random fashion.

We can repeat the construction $B > 1$ times to obtain $B$ different $\mathbf{S}$ matrices.
The compressed design matrix $S$ is generated in a random fashion. We can repeat the construction $B > 1$ times to obtain $B$ different $S$ matrices. In the spirit of bagging (Breiman, 1996) we can then aggregate the predictions obtained from the different random mappings by averaging them.
Aggregation

- The compressed design matrix $S$ is generated in a random fashion.
- We can repeat the construction $B > 1$ times to obtain $B$ different $S$ matrices.
- In the spirit of bagging (Breiman, 1996) we can then aggregate the predictions obtained from the different random mappings by averaging them.
- Using $B > 1$ often gives large improvements and our experience has been that $L$ can be chosen much lower than for $B = 1$ to achieve the same predictive accuracy.
Have $p = 4,272,227$ predictor variables for $n = 16,087$ observations.
• Have $p = 4,272,227$ predictor variables for $n = 16,087$ observations.
• Created artificial responses from i) and linear model ii) a model with interactions.
- Have $p = 4,272,227$ predictor variables for $n = 16,087$ observations.
- Created artificial responses from i) and linear model ii) a model with interactions.
- Compared prediction accuracy with random projections: this method takes a matrix $A \in \mathbb{R}^{p \times L}$ with i.i.d. $N(0,1)$ entries as forms a sketch $XA \in \mathbb{R}^{n \times L}$. 
Response: linear model

Correlation between prediction and response.

Response: interaction model

Correlation between prediction and response.

Large-scale classification of malicious URLs with \( n \approx 2 \text{ million} \) and \( p \approx 3 \text{ million} \).
Data are ordered into consecutive days.

Response \( Y \in \{0, 1\}^n \) is a binary vector where 1 corresponds to a malicious URL.

In order to compare min-wise hashing with the Lasso- and ridge-penalised logistic regression, we split the data into the separate days, training on the first half of each day and testing on the second. This gives on average \( n \approx 20,000 \), \( p \approx 100,000 \).
Ridge regression following min-wise hashing performs here better than ridge regression applied to the original data.
Lasso with and without min-wise hashing have similar performance here.
**Discussion**

*b-bit min-wise hashing* and closely related *random sign min-wise hashing* are interesting dimensionality reduction techniques for large-scale sparse design matrices.

- Prediction error following compression can be bounded with a slow rate (in the absence of assumptions on the design).
- Behaves similarly to random projections if only main effects are present.
- Regression using only main effects from the compressed, dense, low-dimensional matrix can capture interactions among the large number of original sparse variables.