Minwise hashing for large-scale regression and classification with sparse data

Nicolai Meinshausen (Seminar für Statistik, ETH Zürich)
joint work with Rajen Shah (Statslab, University of Cambridge)

Simons Institute
18 September 2013
Large-scale sparse regression

Prediction problems with large-scale sparse predictors:

1. **Medical risk prediction/drug surveillance** (OMOP project). $n \approx 100,000$ patients with $p \approx 30,000$ indicator variables about medication history and symptoms. With interactions of second order, $p \approx 450$ million. With third order $p \approx 4.5$ trillion.

2. **Text data regression or classification**. Binary word indicator variables for approximately $p \approx 20,000$ words. Bi-grams and N-grams of higher order lead to hundreds of millions of variables.

3. **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.
Sparse linear model

Ignoring interactions (for now), can write regression model as:

$$\text{target } Y \in \mathbb{R}^n$$

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

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

$$\text{noise } \epsilon \in \mathbb{R}^n$$

Non-zero entries are marked with $\ast$.

Classification model (logistic regression) analogous.
Can we safely reduce sparse $p$-dimensional problem to a dense $L$-dimensional one with $L \ll p$?

Here: dimensionality reduction with *b-bit minwise hashing* (Li and Koenig, 2011) and a closely related idea.
Suppose we have sets $z_1, \ldots, z_n \subseteq \{1, \ldots, p\}$. Min-wise hashing gives estimates of the Jaccard index of every pair of sets $z_i, z_j$, given by

$$J(z_i, z_j) = \frac{|z_i \cap z_j|}{|z_i \cup z_j|}.$$
Min-wise hashing (Broder, 1997; Broder et al., 1998)

Suppose we have sets $z_1, \ldots, z_n \subseteq \{1, \ldots, p\}$. Min-wise hashing gives estimates of the Jaccard index of every pair of sets $z_i, z_j$, given by

$$J(z_i, z_j) = \frac{|z_i \cap z_j|}{|z_i \cup z_j|}.$$ 

- Let $\pi_1, \ldots, \pi_L$ be random permutations of $\{1, \ldots, p\}$ (in practice all random functions implemented by hash functions).
- Let the $n \times L$ matrix $M$ be given by $M_{il} = \min \pi_l(z_i)$.

Then for each $i, j, l$, $\mathbb{P}(M_{il} = M_{jl}) = J(z_i, z_j)$. 

Minwise hashing for sparse data
Min-wise hashing matrix $\mathbf{M}$

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

Can repeat $L$ times to build $\mathbf{M}$ with repeated (pseudo-) random permutations $\pi$.

\[
\pi = \begin{pmatrix}
2 & 4 & 1 & 3
\end{pmatrix}
\]

\[
\begin{pmatrix}
* & * & * & * \\
* & * & * & * \\
* & * & * & *
\end{pmatrix}
\Rightarrow
\begin{pmatrix}
1 & 3 \\
2 & 1 \\
1 & 1 \\
1 & 2
\end{pmatrix}
\]

Work with $\mathbf{M}$ instead of sparse $\mathbf{X}$. Encode all levels in a column as dummy variables?
b-bit min-wise hashing (Li and König, 2011)

*b-bit min-wise hashing* stores only the lowest *b* bits of each entry of *M* when expressed in binary (i.e. the residue mod 2), so for *b* = 1,

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

Perform regression using binary *n* × *L* matrix *M*\(^{(1)}\) rather than *X*.

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

When *L* \(\ll p\) this gives large computational savings, and empirical studies report good performance (mostly for classification with SVM’s).
Will study a variant of 1-bit min-wise hashing we call MRS-mapping (min-wise hash random sign)

- Easier to analyse and avoids choice of number of bits $b$ to keep.
- Deals with sparse design matrices with real-valued entries.
- Allows for the construction of a variable importance measure.

Downside: slightly less efficient to implement.
MRS-mapping

1-bit min-wise hashing: keep last bit

\[
X = \begin{pmatrix}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{pmatrix} \Rightarrow M = \begin{pmatrix}
1 & 3 \\
2 & 1 \\
2 & 1 \\
1 & 1 \\
1 & 2 \\
\end{pmatrix} \Rightarrow M^{(1)} = \begin{pmatrix}
1 & 1 \\
0 & 1 \\
0 & 1 \\
1 & 1 \\
1 & 0 \\
\end{pmatrix}
\]
MRS-mapping

1-bit min-wise hashing: keep last bit

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

MRS-map: random sign assignments \(\{1, \ldots, p\} \mapsto \{-1, 1\}\) are chosen independently for all columns \(l = 1, \ldots, L\) when going from \(M_l\) to \(S_l\).

\[
X = \begin{pmatrix}
1 & 1 \\
1 & 1 \\
1 & 1 \\
1 & 1 \\
\end{pmatrix}
\Rightarrow
M = \begin{pmatrix}
1 & 3 \\
2 & 1 \\
2 & 1 \\
1 & 1 \\
\end{pmatrix}
\Rightarrow
S = \begin{pmatrix}
1 & 1 \\
-1 & -1 \\
-1 & -1 \\
1 & -1 \\
\end{pmatrix}
\]
Equivalent to storing $\mathbf{M}$, we can store the “responsible” variables in $\mathbf{H}$

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

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

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

Can handle continuous variables

\[
X = \begin{pmatrix}
1 & 4.2 & 1 \\
1 & 1 & \\
7.1 & 1 & \\
\end{pmatrix}
\Rightarrow
H = \begin{pmatrix}
2 & 4 \\
3 & 3 \\
3 & 3 \\
2 & 3 \\
2 & 1 \\
\end{pmatrix}
\Rightarrow
S = \begin{pmatrix}
1 & 1 \\
-4.2 & -4.2 \\
-1 & -1 \\
1 & -1 \\
1 & 7.1 \\
\end{pmatrix}
\]

We get \( n \times L \) matrices \( H \), and \( S \) given by

\[
H_{il} = \text{argmin}_{k \in z_i} \pi_l(k)
\]

\[
S_{il} = \Psi_{H_{il}} X_{iH_{il}}
\]

where \( \Psi_{hl} \) is the random sign of the \( h \)-th variable in the \( l \)-th permutation.
Approximation error

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

- Assume that there are $q \leq p$ non-zero entries in each row of $X$.
- If not, can be dealt with.

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

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

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

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

$S \in \mathbb{R}^{n \times 1}$

$b^* \in \mathbb{R}^{1}$

$E_{\pi, \psi}$
Approximation error

Example: binary $\mathbf{X}$ with one permutation with min-hash value $H_i$ for $i = 1, \ldots, n$ and random signs $\psi_k$, $k = 1, \ldots, p$.

$$\mathbb{E}_{\pi, \psi} \left[ \begin{array}{c} \mathbf{S} \in \mathbb{R}^{n \times 1} \\ (\psi_{H_1} \psi_{H_2} \ldots \ldots \ldots) \end{array} \right] = : \mathbf{b}^* \in \mathbb{R}^1$$

$$= \begin{pmatrix} q \sum_{k=1}^{p} \beta_k^* \psi_k \end{pmatrix}$$
Approximation error

Can we find a $b^* \in \mathbb{R}^L$ such that $X\beta^*$ is close to $Sb^*$ on average?
Example: binary $X$ with one permutation with min-hash value $H_i$ for $i = 1, \ldots, n$ and random signs $\psi_k$, $k = 1, \ldots, p$.

\[
\mathbb{E}_{\pi, \psi} \left[ \begin{pmatrix} \psi_{H_1} \\ \psi_{H_2} \\ \vdots \\ \vdots \\ \vdots \\ S \end{pmatrix} \right] = \left[ \begin{pmatrix} \sum_{k=1}^p \beta_k^* q \mathbb{P}(H_1 = k) \\ \sum_{k=1}^p \beta_k^* q \mathbb{P}(H_2 = k) \end{pmatrix} \right] = b^*
\]
Can we find a $b^* \in \mathbb{R}^L$ such that $X\beta^*$ is close to $Sb^*$ on average? Example: binary $X$ with one permutation with min-hash value $H_i$ for $i = 1, \ldots, n$ and random signs $\psi_k$, $k = 1, \ldots, p$.

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

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

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

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

(i) The approximation is unbiased: $\mathbb{E}_{\pi, \psi}(\mathbf{Sb}^*) = \mathbf{X}\beta^*$. 
Approximation error

**Theorem**

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

$$b^*_l = \frac{q}{L} \sum_{k=1}^{p} \beta^*_k \psi_{kl} w_{\pi_1}(k),$$

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

(i) The approximation is unbiased: $\mathbb{E}_{\pi, \psi}(Sb^*) = X\beta^*$.

(ii) If $\|X\|_\infty \leq 1$, then $\frac{1}{n} \mathbb{E}_{\pi, \psi}(\|Sb^* - X\beta^*\|_2^2) \leq 2q\|\beta^*\|_2^2 / L$. 
Linear model

Assume model

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

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 will give bounds on a mean-squared prediction error (MSPE) of the form

\[ \text{MSPE}(\hat{b}) := \mathbb{E}_{\varepsilon, \pi, \Psi} \left( \|X\beta^* - S\hat{b}\|_2^2 \right) / n. \]
Ordinary least squares

**Theorem**

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

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

- If the size of the signal is fixed and columns of \( X \) are independent with roughly equal sparsity, then \( \sqrt{q}\|\beta^*\|_2 \leq \text{const}\sqrt{p} \) and we have \( \text{MSPE}(\hat{b}) \to 0 \) if \( p/n \to 0 \).
- If the signal \( X\beta^* \) is partially replicated in \( B \) groups of variables then we only need \( (p/B)/n \to 0 \).
Ridge regression

Can also estimate with ridge regression. Very similar results to OLS.

- The dimension $L$ of the projection can be chosen arbitrarily large (from a statistical point of view).
- Ridge penalty parameter is then the relevant tuning parameter.
Can also estimate with ridge regression. Very similar results to OLS.

- The dimension $L$ of the projection can be chosen arbitrarily large (from a statistical point of view).
- Ridge penalty parameter is then the relevant tuning parameter.

Similar results for logistic regression available.
Interactions

Linear model:

\[
\text{target } Y \in \mathbb{R}^n \approx \begin{bmatrix}
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
\end{bmatrix} \times \begin{bmatrix}
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
\end{bmatrix} + \begin{bmatrix}
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
\end{bmatrix}
\]

\[\beta^* \in \mathbb{R}^p \approx \begin{bmatrix}
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
\end{bmatrix} + \begin{bmatrix}
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
* \\
\end{bmatrix}
\]

Can we also fit pair-wise interactions if \( p \geq 10^6 \)?

Min-wise hashing does it (almost) for free.
Interactions

Linear model:

target \( Y \in \mathbb{R}^n \)

\[
\begin{bmatrix}
* \\
* \\
* \\
* \\
* \\
* \\
* \\
*
\end{bmatrix} \approx \begin{bmatrix}
\beta^* \in \mathbb{R}^p \\
\end{bmatrix} + \begin{bmatrix}
noise \, \epsilon \in \mathbb{R}^n \\
\end{bmatrix}
\]

\[\begin{bmatrix}
sparse \, X \in \mathbb{R}^{n \times p} \\
\end{bmatrix}
\]

Can we also fit pair-wise interactions if \( p \geq 10^6 \) ?

⇒ Min-wise hashing does it (almost) for free.
Can view minwise hashing operation as a tree-type operation.
Interaction models

Let $\|X\|_{\infty} \leq 1$ and let $\mathbf{f}^* \in \mathbb{R}^n$ be given by

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

**Theorem**

Define

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

Then there exists $\mathbf{b}^* \in \mathbb{R}^L$ such that

(i) $\mathbb{E}_{\pi,\Psi}(S\mathbf{b}^*) = \mathbf{f}^*$;

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

If there are a finite number of non-zero interaction terms with finite value, the approximation error becomes very small if $L \gg q^2$. 
Assume the linear model from before, but with $X\beta^*$ replaced by $f^*$.

Previous results hold if $\|\beta^*\|_2$ is replaced by $\ell(\Theta^*)$.

For example:

**Theorem**

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

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

Using MRS-maps for interaction fitting

- requires only fit of a linear model
Advantages

Using MRS-maps for interaction fitting
- requires only fit of a linear model
- does not require interactions to be created explicitly
Advantages

Using MRS-maps for interaction fitting
- requires only fit of a linear model
- does not require interactions to be created explicitly
- has a complexity saving factor of $(q/p)^2$ over the brute force approach.

Does require a larger number $L$ of minwise hashing operations than fitting main effect models.
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 contains only main effects, \( \hat{f} - \hat{f}^{-(k)} \approx X_k \beta_k^* \).
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 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 use second-smallest instead of smallest active variable in the random permutation.
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 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 use second-smallest instead of smallest active variable in the random permutation.
Store \( n \times L \) matrices \( S, \tilde{S} \) and \( H \). Then

\[
\hat{f}^{-(k)} = (S \circ 1_{H \neq k} + \tilde{S} \circ 1_{H = k})\hat{b}.
\]
Some observations from numerical simulations:

- Scheme becomes more competitive when repeating many times and aggregating.
Numerical results

Some observations from numerical simulations:

- Scheme becomes more competitive when repeating many times and aggregating.
- Predictive accuracy can decrease if we make $L$ too large.
- In the absence of interactions: similar performance to ridge/random projections
- With interactions: performance between linear model (with ridge penalty or random projections) and Random Forest (Breiman, 01).
Volatility prediction

Forecast financial volatility of stocks based on 10-K report filings (Kogan, 2009).

Have $p = 4,272,227$ predictor variables for $n = 16,087$ observations.

Use various targets (volatility after release; a linear model; a non-linear model) and compare prediction accuracy with regression on random projections.
Volatility prediction

Correlation between prediction and response (volatility in year after release of text). Added additional noise with variance $\sigma^2$ to the response.

Red: MRS-mapping. Blue: random projections (as functions of $L$ up to 500)
Volatility prediction

Response: linear model in original variables

\[ \begin{align*}
\sigma = 0 & \quad \sigma = 0.5 & \quad \sigma = 2 & \quad \sigma = 5 \\
0.0 & \quad 0.2 & \quad 0.4 & \quad 0.6 & \quad 0.8 & \quad 1.0
\end{align*} \]
Volatility prediction

Response: interaction model in original variables

![Graph showing different values of σ (0, 0.5, 2, 5) with their corresponding response curves.](image)
Classification of malicious URLs with $n \approx 2$ million and $p \approx 3$ 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 MRS-mapping 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 $n \approx 20,000$, $p \approx 100,000$. 
Lasso with and without MRS-mapping has similar performance here.
Ridge regression following MRS-mapping performs better than ridge regression applied to the original data.
Discussion

*B-bit minwise hashing* and closely related *MRS-maps* interesting technique for dimensionality reduction for large-scale sparse design matrices.

- Prediction error can be bounded with a slow rate (in the absence of assumptions on the design except sparsity).
- Behaves similar to random projections (or ridge regression) if only linear effects are present.
- Linear model in the compressed, dense, low-dimensional matrix can fit interactions among the large number of original sparse variables.