Blendenpik: Randomized Least Squares

Mini project - Computational Linear Algebra, EPFL

Moritz Waldleben

June 2022

1 Introduction

Blendenpik [1] is a randomized algorithm for linear least-squares problems. The method constructs a random projection preconditioner for an iterative solver and converges in fewer steps than an unpreconditioned one. Moreover, Blendenpik can beat the standard LAPACK implementation for very large matrices.

In the following report, we will discuss the main concepts of Blendenpik and illustrate some results. Instead of the iterative algorithm LSQR used in the paper [1], we will solve the system with the minimum residual method (MINRES).

2 Motivation

Let us look at a large overdetermined system: $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m, m >> n, \text{and} \operatorname{rank}(A) = n$. The corresponding linear least-squares problem can be written as

$$\min_{x \in \mathbb{R}^n} \|A\mathbf{x} - \mathbf{b}\|_2 \tag{1}$$

One might think that the linear system has redundant information. We could thus try to solve a smaller system and sample $\mathcal{S}$ rows of the matrix to get an approximated solution.

$$x_{\mathcal{S}} = \arg\min_{x} \|A_{\mathcal{S}}x - b_{\mathcal{S}})\|_2$$

This reduction technique is rarely implemented, in practice, because of lacking useful bounds for the forward error (See [1] for a detailed explanation).

An alternative way is to construct a preconditioner for the complete linear system in 1 using randomly selected rows. A reduced QR factorization of the randomly selected $\mathcal{S}$ rows of A ($A_{\mathcal{S}}$) turns out to give us a suitable preconditioner. This aspect is further discussed in section 5. The preconditioner we will use is R and the preconditioned matrix results in AR^{-1}.

To start with we have to look at a concept called coherence which is a property to quantify row sampling techniques.

3 Uniform random sampling and the coherence of a matrix

Coherence tells us how much the solution of the system will depend on a single row of A. Formally coherence can be defined as

$$\mu(A) = \max_{i} \|U(i,:)\|_2^2, \tag{2}$$

where U corresponds to a matrix with orthonormal columns which forms a basis for the column space of A. We have as well that $\frac{n}{m} \leq \mu(A) \leq 1$.

To illustrate this, we will look at the coherence of a randomly generated orthogonal matrices of size $\mathbb{R}^{100\times 50}$.

```
% mean coherence of 1000 random matrices
rng(11);
coherences = [];
for i=1:1000
    U = orth(rand(1000, 50));
    coherence = max(sum(U.^2, 2));
    coherences = [coherences, coherence];
end
disp(mean(coherences))
```

```
>> 0.0720
```

The Matlab script above computes the average coherence number of randomly generated orthogonal matrices. As comparison, the lower bound on the coherence for the matrix U is $\mu_{min}(U) = \frac{50}{1000} = 0.05$ from the definition in 2.

The result shows that the coherence number is nearly optimal for randomly generated matrices. On the other hand, we can expect this, as all elements of the matrix are created using a uniform distribution.

Let's now think of an extreme case where the coherence number is maximal. We could set all elements of a column of A to zero except one entry. When creating an orthonormal basis for A this row will persist and lead to a coherence of $\mu(A) = 1$. More intuitively, this means that the row with the nonzero entry has to be sampled to capture the "essence or direction" of the column with one nonzero entry.

In contrast to a uniform sampled matrix discussed above, general matrices A will have high coherence numbers. On might thus try to do some preprocessing on $A_{\mathcal{S}}$, the matrix from randomly selecting rows of A. The next section deals with the problem of reducing the coherence of the sampled matrix.

4 Blending the rows

A technique to reduce coherence is preprocessing A with randomized row mixing. We recall that we want to solve a system with a preconditione R from the QR decomposition of $A_{\mathcal{S}}$. The convergence rate of the iterative solver depends on the singular values of the preconditioned system AR^{-1}.

The idea now is to apply a unitary transformation $\mathcal{F}$ which will still have the singular values of the original one. For this uniform transformation $\mathcal{F}$, several strategies are possible. One of them is to first multiply each row with $+1$ or -1 with equal probability and then apply a normalized discrete cosine transform (DCT). Theorem 3.3 in the Blendenpik paper [1] ensures that $\mathcal{F}AR^{-1}$ has low coherence with a high probability.

In summary, first, we uniformly sample rows using a matrix S to get $SA = A_{\mathcal{S}}$ and then "blend" them (therefore the name Blendenpik) to obtain a low coherence for $A_{\mathcal{S}}$ without changing the condition number of AR^{-1} of the preconditioned system.

It is important to note that there is no obvious connection between the condition number and coherence of a matrix. The following part will now establish this relation.

5 Bound for the condition number

The missing part is to link the coherence of the Matrix A with its condition number. The obtained preconditioner R from the discussions in the previous chapter can be useful if we bound the condition number for the preconditioned iterative solver. This ensures convergences in a few steps. In the case of the MINRES version of Blendenpik, we would use A^TA instead of A and the preconditioned system reads then

$$\begin{aligned} (AR^{-1})^T(AR^{-1})\mathbf{y} &= (AR^{-1})^T\mathbf{b} \\ \mathbf{y} &= R\mathbf{x}. \end{aligned} \tag{3}$$

The following theorem gives us the desired bound and establishes a link between the condition and coherence number.

Theorem 1 *Let A be an $m \times n$ full rank matrix, and let $\mathcal{S}$ be a random sampling operator that samples $r \geq n$ rows from A uniformly. Let $\tau = C\sqrt{m\mu(A)\log(r)/r}$, where C is some constant defined in the proof. Assume that $\delta^{-1}\tau < 1$. With probability of at least $1-\delta$, the sampled matrix $\mathcal{S}A$ is full rank, and if $\mathcal{S}A = QR$ is a reduced QR factorization of $\mathcal{S}A$, we have*

$$\kappa\left(AR^{-1}\right) \leq \sqrt{\frac{1+\delta^{-1}\tau}{1-\delta^{-1}\tau}} \tag{4}$$

To prove the above we will state a theorem that corresponds to theorem 7 in [3] (less general and rephrased for our problem).

Theorem 2 *Suppose $A \in \mathbb{R}^{m\times n}$, and $r \leq n$. Construct C with Algorithm 6, using the EXPECTED (r) algorithm. If the sampling probabilities $\{p_i\}_{i=1}^n$ used by the algorithm is of form (5), then*

$$\mathbf{E}\left[\left\|AA^T - CC^T\right\|_2\right] \leq O(1)\sqrt{\frac{\log r}{\beta r}}\|A\|_F\|A\|_2.$$

To relate this theorem 2 with everything introduced so far some clarifications are needed.

First of all, the mentioned EXPECTED (r) algorithm corresponds to a column sampling algorithm. Every column of a matrix A is sampled with probability rp_i and it holds that $\sum_{i=1}^n p_i = 1$. Hence in expectation r columns are sampled. After that, a rescaling of all entries is done using a diagonal matrix with values of $\frac{1}{\sqrt{rp_i}}$. The Algorithm 6 returns us the matrix $C = A\tilde{S}D$ where we get the matrix $\tilde{S}$ from sampling and D from rescaling in the EXPECTED (r) algorithm. Note that in the theorem the sampling matrix is applied to from the left SA. This means $\tilde{S} = S^T$ of theorem 1.

Secondly, we have to clarify how the parameter $\beta \in (0, 1]$ is established. Theorem 2 holds among others for probabilities defined by

$$p_i \geq \beta\frac{\left\|A^{(i)}\right\|_2^2}{\|A\|_F^2} \tag{5}$$

We will now go on to prove the theorem 1. We start by substituting the general matrix A with the transpose U^T of an $U \in \mathbb{R}^{m\times n}$ matrix with orthonormal columns which span the column space of the matrix A which we are interested in. Now for the Blendenpik algorithm, we will sample uniformly r rows of a matrix U as the theorem is applied to U^T. The probabilities are thus set to $p_i = \frac{1}{m}$ and entries of the scaling matrix D are equal to $\sqrt{\frac{r}{m}}$. Further, we know that for a matrix with orthonormal columns it holds that $U^TU = I$. In total, we can restate the inequality as follows

$$\mathbf{E}\left[\left\|I_{n\times n} - \frac{m}{r}U^TS^TSU\right\|_2\right] \leq O(1)\sqrt{\frac{\log r}{\beta r}}\|U^T\|_F\|U^T\|_2$$

Now we have to work on the upper bound. We can further develop inequality (5). As we want to bound by the coherence of A we get

$$\beta \leq p_i\frac{\|U^T\|_F^2}{\left\|U^{T(i)}\right\|_2^2}.$$

This inequality still holds for $\beta = p_i\frac{\|U^T\|_F^2}{\mu(A)} = \frac{\|U^T\|_F^2}{m\mu(A)}$. This is true as the column norm of U is smaller than the coherence of our original matrix A as coherence is defined using U. The right hand side becomes

$$\begin{aligned}\mathbf{E}\left[\left\| I_{n\times n} - \frac{m}{r}U^T S S^T U \right\|_2\right] &\leq O(1)\sqrt{\frac{m\mu(A)\log r}{r}}\|U^T\|_2 \\ &\leq O(1)\sqrt{\frac{m\mu(A)\log r}{r}} \\ &= \tau\end{aligned}$$

where the third step follows from the fact that the second matrix norm of U^T is one as

$$\|U^T\|_2 = \sqrt{\lambda_{max}((U^T U)(U^T U)^T)} = \sqrt{\lambda_{max}(I)} = 1.$$

The proof has to be completed with showing that the expectation in the inequality can be changed to involve $\kappa(AR^{-1})$. We will now state a theorem which corresponds to theorem 1 in [2].

Theorem 3 *Suppose that $l, m,$ and n are positive integers such that $m \geq l \geq n$. Suppose further that A is a full-rank $m \times n$ matrix, and that the SVD of A is*

$$A_{m\times n} = U_{m\times n}\Sigma_{n\times n}V^*_{n\times n}.$$

Suppose in addition that T is an $l \times m$ matrix such that the $l \times n$ matrix TU has full rank. Then, there exists an $n \times n$ matrix P, and an $l \times n$ matrix Q whose columns are orthonormal, such that

$$T_{l\times m}A_{m\times n} = Q_{l\times n}P_{n\times n}. \tag{6}$$

Furthermore, if P is any $n \times n$ matrix, and Q is any $l \times n$ matrix whose columns are orthonormal, such that P and Q satisfy 6, then the condition numbers of AP^{-1} and TU are equal.

This means that $\kappa(AR^{-1}) = \kappa(U_S)$. To go further we apply the Markov's inequality (i.e. $\mathbb{P}(X \geq a) \leq \frac{\mathbb{E}(X)}{a}$) on the left side of the inequality with $a = \delta^{-1}\tau$. This yields

$$\mathbb{P}(\left[\left\| I_{n\times n} - \frac{m}{r}U_S^T U_S \right\|_2\right] \geq \tau^{-1}\delta) \leq \delta.$$

With probability $1 - \delta$ we have that

$$\left\| I_{n\times n} - \frac{m}{r}U_S^T U_S \right\|_2 \geq \tau^{-1}\delta) \leq \delta < 1.$$

From the assumptions in theorem 1, we get the bound of 1 and additionally it was stated that A_S is full rank with probability $1 - \delta$.

No we will use a Rayleigh quotient argument to establish $\kappa(U_{\mathcal{S}})$. The matrix $M = \frac{m}{r} U_{\mathcal{S}}^T U_{\mathcal{S}}$ is symmetric and thus each eigenvalue of M will be equal to a Rayleigh quotient of $R(M, x)$ for x nonzero. We get that

$$R(M,x) = \frac{x^T M x}{x^T x} \quad = \frac{x^T(I - I + M)x}{x^T x} = \frac{x^T x - x^T(I - M)x}{x^T x} \quad = 1 - R(I - M, x).$$

$I - M$ is as well symmetric and the spectral norm is smaller than $\delta^{-1}\tau$ with probability $1 - \delta$. This means $|R(I - M, x)| \leq \delta^{-1}\tau$ with the same probability. From the equation above it follows that $R(M, x)$ can be found in between $1 - \delta^{-1}\tau$ and $1 + \delta^{-1}\tau$.

Finally, the condition number of a matrix can be defined as $\kappa(A) = \sqrt{\frac{\lambda_{max}(A^T A)}{\lambda_{min}(A^T A)}}$. We get the desired bound applying this and canceling out the term $\frac{m}{r}$

$$\kappa(U_S) = \frac{\lambda_{max}(U^T U)}{\lambda_{min}(U^T U)} \leq \sqrt{\frac{1 + \delta^{-1}\tau}{1 - \delta^{-1}\tau}}.$$

This gives us the result as we recall that $\kappa(AR^{-1}) = \kappa(U_S)$.

6 Algorithm

Based on the discussion of the previous sections below the pseudo-code for the Blendenpik algorithm with MINRES as an iterative solver.

Algorithm 1 Blendenpik with MINRES and DCT transformation

7 Numerical experiments

Now it is interesting to see how many rows we would have to sample to get a good preconditioner R. In addition for theorem 1, we would like to see the influence of coherence on the number of iterations until convergence.

To conduct our experiments, let us define two ill-conditioned matrices one is a coherent (large $\mu(A)$) A_1 and one an incoherent matrix A_2.

```
rng(11);
U = orth(rand(20000,400));
S = diag(linspace(1,1e5,400));
A1 = U*S*V’;

rng(11);
A2 = [diag(linspace(1,e5,400)); zeros(19600,400)];
A2 = A2 + 1e-8*ones(20000,400);
```

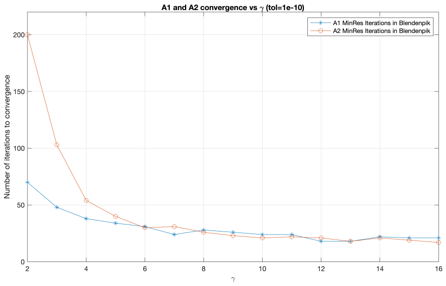


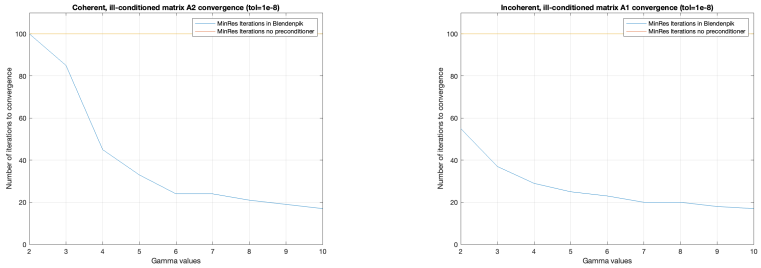
Figure 1: test

If we look at a sampling probability of $\gamma n/\tilde{m}$ we can plot

...

The plot is not surprising for small sample sizes (ex. $\gamma = 2$) the preconditioner is too bad.

As a next step, we can try to plot the convergence of the residual of our precondition system at each iteration step for the matrices. We will as well use as a comparison the LSQR solver applied to the normal equation. We fix γ to 5.



(a) label 1

(b) label 2

Figure 2: 2 Figures side by side

....

References

[1] Haim Avron and Petar Maymounkov. Blendenpik: Supercharging lapack's least-squares solver. *SIAM J. Scientific Computing*, 32:1217–1236, 01 2010.

[2] Petros Drineas, Michael Mahoney, and Senthilmurugan Muthukrishnan. Relative-error *cur* matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, 30:844–881, 05 2008.

[3] Vladimir Rokhlin and Mark Tygert. A fast randomized algorithm for overdetermined linear least-squares regression. *Proceedings of the National Academy of Sciences of the United States of America*, 105:13212–7, 09 2008.