Randomized SVD Algorithm[1]

Given an $m \times n$ matrix $\mathbf{A}$, a target number k of singular vectors, and an exponent q (say, $q = 1$ or $q = 2$), this procedure computes an approximate rank-$2k$ factorization $\mathbf{U\Sigma V}^*$, where $\mathbf{U}$ and $\mathbf{V}$ are orthonormal, and $\mathbf{\Sigma}$ is nonnegative and diagonal. **Stage A:**

1. Generate an $n \times 2k$ Gaussian test matrix $\mathbf{\Omega}$.
2. Form $\mathbf{Y}_0 = \mathbf{A\Omega}$ and compute its QR factorization $\mathbf{Y}_0 = \mathbf{Q}_0\mathbf{R}_0$
3. **for** $j = 1, 2, \ldots, q$
   Form $\widetilde{\mathbf{Y}}_j = \mathbf{A}^*\mathbf{Q}_{j-1}$ and compute its QR factorization $\widetilde{\mathbf{Y}}_j = \widetilde{\mathbf{Q}}_j\widetilde{\mathbf{R}}_j$
   Form $\mathbf{Y}_j = \mathbf{A}\widetilde{\mathbf{Q}}_j$ and compute its QR factorization $\mathbf{Y}_j = \mathbf{Q}_j\mathbf{R}_j$.
   end
4. Let $\mathbf{Q} = \mathbf{Q}_q$, so that the columns of $\mathbf{Q}$ form an orthonormal basis for the range of $\mathbf{Y}$.

Stage B:

5. Form $\mathbf{B} = \mathbf{Q}^*\mathbf{A}$
6. Compute an SVD of the small matrix: $\mathbf{B} = \widetilde{\mathbf{U}}\mathbf{\Sigma V}^*$
7. Set $\mathbf{U} = \mathbf{Q}\widetilde{\mathbf{U}}$.

---

[1] Halko N, Martinsson P-G and Tropp J A 2011 Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions *SIAM* Rev. **53** 217–88

Randomized SVD

Serial R

```
1  randSVD <- function(A, k, q=3)
2    {
3      ## Stage A
4      Omega <- matrix(rnorm(n*2*k),
5                      nrow=n, ncol=2*k)
6      Y <- A %*% Omega
7      Q <- qr.Q(qr(Y))
8      At <- t(A)
9      for(i in 1:q)
10       {
11         Y <- At %*% Q
12         Q <- qr.Q(qr(Y))
13         Y <- A %*% Q
14         Q <- qr.Q(qr(Y))
15       }
16
17     ## Stage B
18     B <- t(Q) %*% A
19     U <- La.svd(B)$u
20     U <- Q %*% U
21     U[, 1:k]
22   }
```

Parallel pbdR

```
1  randSVD <- function(A, k, q=3)
2    {
3      ## Stage A
4      Omega <- ddmatrix("rnorm", nrow=n,
5                        ncol=2*k)
6      Y <- A %*% Omega
7      Q <- qr.Q(qr(Y))
8      At <- t(A)
9      for(i in 1:q)
10       {
11         Y <- At %*% Q
12         Q <- qr.Q(qr(Y))
13         Y <- A %*% Q
14         Q <- qr.Q(qr(Y))
15       }
16
17     ## Stage B
18     B <- t(Q) %*% A
19     U <- La.svd(B)$u
20     U <- Q %*% U
21     U[, 1:k]
22   }
```