1 The power method

We give a simple algorithm for computing the Singular Value Decomposition of a matrix $A \in \mathbb{R}^{m \times n}$. We start by computing the first singular value $\sigma_1$ and left and right singular vectors $u_1$ and $v_1$ of $A$, for which $\min_{i<j} \log(\sigma_i/\sigma_j) \geq \lambda$:

1. Generate $x_0$ such that $x_0(i) \sim \mathcal{N}(0, 1)$.
2. $s \leftarrow \log(4 \log(2n/\delta)/\varepsilon\delta)/2\lambda$
3. for $i$ in $[1, \ldots, s]$: 
4. $x_i \leftarrow A^T Ax_{i-1}$
5. $v_1 \leftarrow x_i/\|x_i\|$ 
6. $\sigma_1 \leftarrow \|Av_1\|$
7. $u_1 \leftarrow Av_1/\sigma_1$
8. return $(\sigma_1, u_1, v_1)$

Let us prove the correctness of this algorithm. First, write each vector $x_i$ as a linear combination of the right singular values of $A$ i.e. $x_i = \sum_j \alpha^i_j v_j$. From the fact that $v_j$ are the eigenvectors of $A^TA$ corresponding to eigenvalues $\sigma^2_j$ we get that $\alpha^i_j = \alpha^i_1^{-1} \sigma^2_j$. Thus, $\alpha^i_1 = \alpha^0_1 \sigma^2$. Looking at the ratio between the coefficients of $v_1$ and $v_i$ for $x_s$ we get that:

$$\frac{|<x_s, v_1>|}{|<x_s, v_i>|} = \frac{|\alpha^0_1|}{|\sigma^i_1|} \left(\frac{\sigma_1}{\sigma_i}\right)^{2s}$$

Demanding that the error in the estimation of $\sigma_1$ is less than $\varepsilon$ gives the requirement on $s$.

$$\frac{|\alpha^0_1|}{|\sigma^i_1|} \left(\frac{\sigma_1}{\sigma_i}\right)^{2s} \geq \frac{n}{\varepsilon} \quad (1)$$

$$s \geq \frac{\log(n|\alpha^0_1|/\varepsilon|\alpha^0_1|)}{2\log(\sigma_1/\sigma_i)} \quad (2)$$
From the two-stability of the gaussian distribution we have that $\alpha_i^0 \sim \mathcal{N}(0,1)$. Therefore, $\Pr[\alpha_i^0 > t] \leq e^{-t^2}$ which gives that with probability at least $1 - \delta/2$ we have for all $i$, $|\alpha_i^0| \leq \sqrt{\log(2n/\delta)}$. Also, $\Pr[|\alpha_i^0| \leq \delta/4] \leq \delta/2$ (this is because $\Pr[|z| < t] \leq \max_r \Psi_z(r) \cdot 2t$ for any distribution and the normal distribution function at zero takes it maximal value which is less than 2) Thus, with probability at least $1 - \delta$ we have that for all $i$, $|\alpha_i^0| \leq \sqrt{\log(2n/\delta)}$. Combining all of the above we get that it is sufficient to set $s = \log(4n \log(2n/\delta)/\varepsilon\delta)/\lambda = O(\log(n/\varepsilon\delta)/\lambda)$ in order to get $\varepsilon$ precision with probability at least $1 - \delta$.

We now describe how to extend this to a full SVD of $A$. Since we have computed $(\sigma_1, u_1, v_1)$, we can repeat this procedure for $A - \sigma_1 u_1 v_1^T = \sum_{i=2}^{n} \sigma_i u_i v_i^T$. The top singular value and vectors of which are $(\sigma_2, u_2, v_2)$. Thus, computing the rank-k approximation of $A$ requires $O(mnks) = O(mnk \log(n/\varepsilon\delta)/\lambda)$ operations. This is because computing $A^T A x$ requires $O(mn)$ operations and for each of the first $k$ singular values and vectors this is performed $s$ times.

The main problem with this algorithm is that its running time is heavily influenced by the value of $\lambda$. Other variants of this algorithm are much less sensitive to the value of this parameter, but are out of the scope of this class.