15-496/859X: Computer Science Theory for the Information Age — Spring 2012
Carnegie Mellon University — V. Guruswami & R. Kannan

Final Solution
by Yuan Zhou

1. Write A in its SVD form $A = \sum_i \sigma_i \boldsymbol{w}_i \boldsymbol{v}_i^T$ where u_i, v_i are unit vectors and $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \ldots$. Let $A_k = \sum_{i=1}^k \sigma_i \boldsymbol{w}_i \boldsymbol{v}_i^T$.

   (a) False. Take A to be the 5×5 identity matrix and let $k = 2$. The matrix B of rank at most k that minimizes $\|A - B\|_F$ is A_2, i.e. two ones in the upper-left two diagonal entries and zeroes in other entries. We have $\|A - A_2\|_F = \sqrt{3}$ while $\|A\|_F/\sqrt{k} = \sqrt{5/2} < \sqrt{3}$.

   (b) True. Take $B = A_k$. We have $\|A - A_k\|^2 = \sigma_{k+1}^2$. Since $\|A\|_F^2 = \sum_i \sigma_i^2$, we have $k \cdot \|A - A_k\|_2^2 \leq \|A\|_F^2$, i.e. $\|A - A_k\|_2 \leq \|A\|_F/\sqrt{k}$.

   (c) Let $k = 1/\epsilon^2$. And let $\boldsymbol{u}_i = A_k \boldsymbol{x}_i = \sum_{t=1}^k \sigma_t \boldsymbol{w}_t \boldsymbol{v}_t^T \boldsymbol{x}_i$. Note that computing $\boldsymbol{u}_i$ uses $O(k(d+n)) = O((d+n)/\epsilon^2)$ time. Now we prove that $\|\boldsymbol{u}_i - A\boldsymbol{x}_i\|_2 \leq \epsilon \|A\|_F \|\boldsymbol{x}_i\|_2$. This is because

   $$\|\boldsymbol{u}_i - A\boldsymbol{x}_i\|_2 = \|(A_k - A)\boldsymbol{x}_i\|_2 \leq \|(A_k - A)\|_2 \|\boldsymbol{x}_i\|_2 \leq (\|A\|_F/\sqrt{k})\|\boldsymbol{x}_i\|_2 = \epsilon \|A\|_F \|\boldsymbol{x}_i\|_2,$$

   where the second inequality uses the result of part (b).

2. (a) $n - 1$.

   (b) $p(1 - (1-p)^2)^{n-2} = p(2p - p^2)^{n-2} = p^{n-1}(2-p)^{n-2}$.

3. $\mathbf{E}[|X|] = 2\int_0^\infty \frac{1}{\sqrt{2\pi}} x e^{-x^2/2} dx = \int_0^\infty \sqrt{2/\pi} e^{-x^2/2} d(x^2/2) = \sqrt{2/\pi}$.

4. We follow the proof of Theorem 6.10 in the book. We only need to choose suitable m such that $\pi_{\mathbb{S}}(2m) \cdot 2^{-\epsilon m/2} \leq \delta$. Here we use the bound (by Lemma 6.6 in the book) that

   $$\pi_{\mathbb{S}}(2m) \leq \sum_{i=0}^{d} \binom{2m}{i} \leq d\binom{2m}{d} \leq \frac{(2m)^d}{(d-1)!}.$$

   Therefore, by choosing $m = \frac{100}{\epsilon}(\log(1/\delta) + d\log(1/\epsilon))$, we only need to prove that $\frac{(2m)^d}{(d-1)!} \cdot 2^{-\epsilon m/2} \leq \delta$. Using the bound $(d-1)! \geq e((d-1)/e)^{d-1}$, we have

   $$\begin{aligned}
   &\frac{(2m)^d}{(d-1)!} \cdot 2^{-\epsilon m/2} \\
   \leq &\frac{(2m)^d e^{d-2}}{(d-1)^{d-1}} \cdot 2^{-\epsilon m/2} \\
   = &2^d e^{d-2} \cdot \frac{d^d(\log(1/\epsilon) + \log(1/\delta)/d)^d}{(d-1)^{d-1}} \cdot \delta^{50} \epsilon^{50d} \\
   \leq &2^d e^{d-1} \cdot d(\log(1/\epsilon) + \log(1/\delta)/d)^d \cdot \delta^{50} \epsilon^{50d} \\
   \leq &(\log(1/\epsilon) + \log(1/\delta)/d)^d \cdot \delta^{50} \epsilon^{25d} \qquad \text{(for } \epsilon < 1/2\text{)} \\
   \leq &(\epsilon^{10} + \epsilon^{25} \log(1/\delta)/d)^d \cdot \delta^{50} \qquad \text{(again, for } \epsilon < 1/2\text{)} \\
   = &(\epsilon^{10} + \epsilon^{25} \delta^{49/d} \log(1/\delta)/d)^d \cdot \delta \\
   \leq &(\epsilon^{10} + \epsilon^{25})^d \cdot \delta \qquad \text{(note that } \delta^{49/d} \log(1/\delta)/d < 1 \text{ for all } 0 < \delta < 1\text{)} \\
   \leq &\delta.
   \end{aligned}$$

5. Output all the vertices of degree greater than $n/2 + 2.5\sqrt{n \ln n}$. It is easy to see that this algorithm runs in $O(n^2)$ times. Now we are going to show that it outputs the correct set S with probability at least $1 - O(1/n^2)$. By Chernoff bound and union bound, we know that in a random $G(n, 1/2)$ instance, with probability at least $1 - 1/n^2$, the degrees of all the vertices belong to the interval $(n/2 - 2\sqrt{n \ln n}, n/2 + 2\sqrt{n \ln n})$. Fix a set S of $k = 10\sqrt{n \ln n}$ vertices, using Chernoff bound and union bound again, we know that with probability at least $1 - 1/n^2$, in the induced graph by S, all the vertices have degree no more than $5.5\sqrt{n \ln n}$. Therefore, by a union bound, with probability at least $1 - 2/n^2$, after putting in the missing edges to make S a clique, the degrees of vertices in S are greater than $n/2 + 2.5\sqrt{n \ln n}$, while the degrees of other vertices are less than $n/2 + 2\sqrt{n \ln n}$ – in this case, the algorithm outputs the correct set S.

6. (a) This is a special case of part (b) (when $m = 0$).

   (b) Assign weights $w_1, w_2, \cdots, w_n$ to the n websites and start with $w_i = 1$ for all i. On each day, if the total weight of the websites that say "up" is more than the total weight of the ones that say "down", predict "up", and predict "down" otherwise. If the prediction turns out to be wrong, reduce the weights of the websites that give the wrong prediction to $1/2$ times the original weights.

   Now we are going to upper bound the total number of wrong predictions we make. Consider the sum $W = \sum_{i=1}^{n} w_i$ at each day. Since the best websites makes at most m mistakes, its weight is at least $(1/2)^m$. Therefore we have $W \geq (1/2)^m$. Each time we make a wrong prediction, the weights of the websites making the mistake (which is at least $W/2$ in total, by our strategy) are halved. Therefore when we make a wrong prediction, the updated weight $W' \leq W \cdot (3/4)$. Since W starts from n, after k wrong predictions, we have $W \leq n \cdot (3/4)^k$. In all, we have

$$n \cdot (3/4)^k \geq (1/2)^m,$$

   which implies that $k \leq O(m + \log n)$.