CSCI 6971/4971: Homework 4

Assigned Tuesday March 6 2017. Due at beginning of class Monday March 19 2017.

Submission Instructions: *Your submission should stand alone: I should be able to read your notebook and understand what question each output is answering without referring back to the assignment.*

1. Use Python+NumPy+SciPy+(Sklearn where convenient) and document your work using a Jupyter notebook. This is a common working environment in industry, and will allow you to interleave code, plots, and typeset written information, and submit your entire assignment as one file. Put your name at the top of the notebook and email it to me as `hw1_<FirstnameLastname>.ipynb` at `gittea@rpi.edu`.
2. I should be able to run your code on my system with minimal modifications. To ensure this, isolate any input locations, parameters that you manually set, etc., that I may need to change to a clearly labeled cell at the top of the notebook, and document these settings.
3. Label all plots and tables appropriately and comment your code liberally and meaningfully.

1. [50 points] (SVD compared to CUR)

   (a) Download the lighthouse image from `http://www.cs.rpi.edu/~gittea/temp/lighthouse.png` and load it (as a grayscale image) as an array $\mathbf{A}$.

   

   (b) Compute and plot all of the singular values of $\mathbf{A}$. Comment.

   (c) Take $k = 5$ and compute the optimal rank-k approximation to $\mathbf{A}$.

   (d) Compute the leverage scores filtered through dimension k for the columns of $\mathbf{A}$ and plot them.

   (e) With $c = 2k, k^2$, compute leverage score-based CUR approximations to $\mathbf{A}$ using the algorithm given in class[1] and CUR approximations formed by uniformly randomly sampling columns and rows with replacement to form $\mathbf{C}$ and $\mathbf{R}$ and taking $\mathbf{U}$ to be the pseudoinverse of the intersection of $\mathbf{C}$ and $\mathbf{R}$.

   (f) Repeat the above process 10 times for each value of c and each of the two types of CUR approximations. Report the average approximation errors $\|\mathbf{A} - \mathbf{CUR}\|_F / \|\mathbf{A} - \mathbf{A}_k\|_F$ and the standard deviations.

   (g) Provide visualizations of (one instance of) the two CUR approximations just computed and a visualization of $\mathbf{A}_k$. Use rescaling as required to obtain reasonable images.

[1] See your notes, but here's a quick outline: sample the columns of $\mathbf{A}$ using the leverage scores of $\mathbf{A}$ filtered through the k-dimensional space to form $\mathbf{C}$, then sample rows from $\mathbf{A}$ using the leverage scores of the rows of $\mathbf{C}$, and *rescale* these rows to get $\mathbf{R}$; take $\mathbf{U}$ to be $\mathbf{W}^\dagger$, where $\mathbf{W}$ comprises the corresponding rescaled rows of $\mathbf{C}$.

(h) Comment on the visualizations and the approximation errors.

2. [50 points] (nonnegative CUR) In the previous problem, we approximated an image (a matrix all of whose entries are nonnegative) with an unconstrained CUR decomposition. To take advantage of the fact that we are appoximating an image matrix, we can impose a nonnegative constraint on the CUR decomposition by solving

$$\operatorname{argmin}_{\mathbf{U}\geq\mathbf{0}} \|\mathbf{A}-\mathbf{CUR}\|_F^2. \tag{1}$$

This is still a least squares problem, as can be seen by using vectorization of the matrices. When $\mathbf{A}\in\mathbb{R}^{m\times n}$, let $\operatorname{vec}(\mathbf{A})$ denote the vector in $\mathbb{R}^{mn}$ formed by stacking the columns of $\mathbf{A}$ on top of each other. The Kronecker product of an $m\times n$ matrix $\mathbf{A}$ and a $p\times q$ matrix $\mathbf{B}$ is the $mp\times nq$ block matrix

$$\mathbf{A}\otimes\mathbf{B} = \begin{bmatrix} a_11\mathbf{B} & \cdots & a_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & \cdots & a_{mn}\mathbf{B} \end{bmatrix}.$$

One can verify that $\operatorname{vec}(\mathbf{CUR}) = (\mathbf{R}^T\otimes\mathbf{C})\operatorname{vec}(\mathbf{U})$. It follows that

$$\operatorname{argmin}_{\mathbf{u}\geq\mathbf{0}} \|(\mathbf{R}^T\otimes\mathbf{C})\mathbf{u}-\operatorname{vec}(\mathbf{A})\|_2^2 \tag{2}$$

is equivalent to (1) in that $\mathbf{u}^\star = \operatorname{vec}(\mathbf{U}^\star)$.

(a) Assuming $\mathbf{C}$ contains c columns of $\mathbf{A}$ and $\mathbf{R}$ contains c rows of $\mathbf{A}$, what is the size of $\mathbf{R}^T\otimes\mathbf{C}$? Is this a tall and skinny matrix, a fat matrix, or a square matrix?

(b) Sketching works even when we have the nonnegativity constraint, in that we can get a $(1+\varepsilon)$ error approximation to (2) when $\mathbf{S}$ is an (ε,δ) subspace embedding. Provide an argument for this fact by adapting the proof we saw in class that (ε,δ) subspace embeddings give $(1+\varepsilon)$ error approximations to unconstrained least squares problems.

(c) What is a sufficient[2] ℓ to ensure that leverage score sampling can be used to solve (2) to $(1+\varepsilon)$ approximate accuracy with at least $1-\delta$ success probability? You can ignore constants. Say how you got this estimate.

(d) Let $k=5$ in the following, and sample $\mathbf{C}$ and $\mathbf{R}$ with $c=k^2$ using leverage score sampling from the $\mathbf{A}$ in the previous problem, to be used in constructing standard and nonnegative CUR decompositions below.

(e) Compute the exact solution to (2) using the SciPy nonnegative least squares solver, and record the time required to compute it.

(f) Compute the leverage scores needed to use leverage score sampling to approximately solve (2).

(g) With $\ell=c^2, c^2\ln(c)$, use these leverage scores to sample and *rescale* the rows of the linear system in (2) appropriately, then pass these into a SciPy's nonnegative least squares solver to obtain an approximate solution $\hat{\mathbf{u}}$. For each of these, report the total time required to compute the leverage scores and then solve the sketched nonnegative least squares system. Is there a computational advantage to sketching?

(h) Provide a visual comparison of the corresponding CUR image matrices for the two approximations just computed, the CUR obtained from solving (2) exactly, the rank-k SVD, and the leverage score-based CUR approximations computed in the first problem that have no nonnegativity constraints. Label the plots appropriately. What do you observe?

[2]We used c to denote the number of columns/rows in $\mathbf{C}/\mathbf{R}$, so we're using ℓ to denote the sketching dimension for the least squares problem (1).