CS 3220: Prelim 1 solutions

Instructor: Anil Damle

Policies and details

This document provides solutions to the first prelim exam. It is deliberately verbose in an effort to provide several potential solutions to each problem and serve as an effective learning tool. It was quite possible to get full points with a significantly more terse solution.

Question 1

For the entirety of this problem you should assume all matrices involved are symmetric. In class we talked about the power method, an algorithm (that under mild assumptions) allowed us to compute the largest magnitude eigenvalue of A. For a matrix A, starting guess for the desired eigenvector $v^{(0)}$, desired accuracy $\epsilon > 0$, and maximal number of iterations K the power method is given in Algorithm 1

Algorithm 1 The power method

input: A, $v^{(0)}$, and ϵ
output: Eigenvector estimate v and eigenvalue estimate λ

$\lambda^{(0)} = (v^{(0)})^T A v^{(0)}$
for $k = 1, 2, \ldots, K$ **do**
  $v^{(k)} = A v^{(k-1)}$
  $v^{(k)} = v^{(k)} / \|v^{(k)}\|_2$
  $\lambda^{(k)} = (v^{(k)})^T A v^{(k)}$
  if $\|A v^{(k)} - \lambda^{(k)} v^{(k)}\|_2 \leq \epsilon$ **then**
    return: $v \equiv v^{(k)}$ and $\lambda \equiv \lambda^{(k)}$
  end if
end for
return: $v \equiv v^{(k)}$ and $\lambda \equiv \lambda^{(k)}$

Now, lets say that we have a symmetric non-singular matrix A and we would like to compute the eigenvalue smallest in magnitude. Denote the eigenvalue of A as $\lambda_1, \ldots, \lambda_n$ with associated eigenvectors $v_1, \ldots, v_n$ and assume they satisfy $|\lambda_n| < |\lambda_{n-1}| \leq \cdots \leq |\lambda_1|$. Address the following questions:

(a) In terms of the eigenvalues of A, what is the largest magnitude eigenvalue of A^{-1}? (Recall that we have assumed A is non-singular.)

(b) In light of the previous question, can you use the power method (perhaps applied to a matrix related to A rather than A itself) to find v_n and λ_n? Rewrite Algorithm 1 to accomplish this goal.

(c) We can break down the computational cost of the power method into two parts—fixed costs and any additional cost per iteration. For example, in Algorithm 1 there are no fixed costs and the cost per iteration is $\mathcal{O}(n^2)$ (as a consequence of the matrix vector multiplication). What is the cost (broken down in this manner) for computing λ_n and v_n? Clearly explain your answer.

If your algorithm as written costs more than $\mathcal{O}(n^2)$ per iteration can you rewrite the algorithm so the per iteration cost is $\mathcal{O}(n^2)$? If so, do this and explain what the fixed cost is to make this happen.

(d) For Algorithm 1 we argued that the convergence rate of $v^{(k)}$ to v_1 is $\mathcal{O}\left(|\lambda_2/\lambda_1|^k\right)$ and the convergence rate of $\lambda^{(k)}$ to λ_1 is $\mathcal{O}\left(|\lambda_2/\lambda_1|^{2k}\right)$. For your method in the prior part, what are the convergence rates of $v^{(k)}$ to v_n and $\lambda^{(k)}$ to λ_n? Clearly justify your answers.

Question 1 solution

(a) There are many ways to come to the following conclusion: the eigenvalues of A^1 are $1/\lambda_i$, where λ_i is an eigenvalue of A. Possible approaches include (reall that A is non-singular and therefore $|\lambda_n| > 0$) starting form

$$Av_i = \lambda_i v_i,$$

whereby multiplying both sides by $\frac{1}{\lambda_i}A^{-1}$ yields

$$A^{-1}v_i = \frac{1}{\lambda_i}v_i$$

for $i = 1, 2, \ldots, n$. Observe that the eigenvectors remain unchanged. Alternatively, one can start with the spectral decomposition $A = V\Lambda V^T$ and conclude that $A^{-1}V = V\Lambda^{-1}$ where we have used that $V^{-1} = V^T$ since it is an orthogonal matrix and that $(\Lambda^{-1})_{ii} = 1/\lambda_i$ since it is diagonal.

Irrespective of how one gets to the fact that the eigenvalues of A^{-1} are $1/\lambda_i$ for $i = 1, 2, \ldots, n$ we can immediately conclude that the largest eigenvalue of A^{-1} is $1/\lambda_n$.

(b) From the problem statement we know that $1/\lambda_n > 1/\lambda_{n-1}$ and therefore we can run the power method on A^{-1} and converge to $1/\lambda_n$. As noted above, since the eigenvectors do not change we also get v_n via this process. The key is to note that given A we need a way to apply A^{-1} to a vector at each step. This can be accomplished via, *e.g.,* use of a QR factorization. This algorithm is summarized as Algorithm 2, note that at the end we do not need to take the reciprocal of the computed eigenvalue to actually get λ_n back because we compute the Rayleigh quotient with A rather than A^{-1}.

(c) In Algorithm 2, we have introduced two steps that could increase the overall cost. The factorization $A = QR$ is outside the for loop, so even though it costs $\mathcal{O}(n^3)$ we only have to do it once. Actually "applying" A^{-1} to $v^{(k-1)}$ is accomplished via multiplication by Q^T and solving a triangular system, both of which cost $\mathcal{O}(n^2)$. The existing work within each loop iteration is no different from Algorithm 1 and costs $\mathcal{O}(n^2)$. This collectively means we have satisfied the requirements of the problem. The key to keeping the per iteration cost the same as before is to factor A once outside of the loop.

(d) Since we are effectively just applying the power method to the matrix A^{-1} we can simply use the largest and second largest eigenvalues of A^{-1} in the convergence results. These values are $1/\lambda_n$ and $1/\lambda_{n-1}$ respectively. Therefore, the rate of convergence of $v^{(k)}$ to v_n is $\mathcal{O}\left(|\lambda_n/\lambda_{n-1}|^k\right)$ and the convergence rate of $\lambda^{(k)}$ to λ_n is $\mathcal{O}\left(|\lambda_n/\lambda_{n-1}|^{2k}\right)$.

Algorithm 2 The inverse power method

input: A, $v^{(0)}$, and ϵ
output: Eigenvector estimate v and eigenvalue estimate λ

$\lambda^{(0)} = (v^{(0)})^T A v^{(0)}$
Factor $A = QR$
for $k = 1, 2, \ldots, K$ **do**
    Solve $Rv^{(k)} = Q^T v^{(k-1)}$ for $v^{(k)}$ using backwards substitution
    $v^{(k)} = v^{(k)} / \|v^{(k)}\|_2$
    $\lambda^{(k)} = (v^{(k)})^T A v^{(k)}$
    if $\|Av^{(k)} - \lambda^{(k)} v^{(k)}\|_2 \leq \epsilon$ **then**
        return: $v \equiv v^{(k)}$ and $\lambda \equiv \lambda^{(k)}$
    end if
end for
return: $v \equiv v^{(k)}$ and $\lambda \equiv \lambda^{(k)}$

Question 2

While discussed the definition of an induced matrix norm in general, we only really talked about it in detail when the vector norm we used was $\|\cdot\|_2$. Recall the definition of an induced matrix norm given any vector norm $\|\cdot\|$ is

$$\|A\| = \max_{\|x\|=1} \|Ax\|.$$

We will now consider $\|A\|_\infty$ and see how it relates to the entries of A. (Turns out that this is not a hard matrix norm to compute given the entries of A.)

(a) Prove that for any x with $\|x\|_\infty \leq 1$

$$\|Ax\|_\infty \leq \max_{j=1,\ldots,n} \sum_{i=1}^{n} |A(j,i)|.$$

(b) Prove that for any $j \in \{1, \ldots, n\}$ there exists a vector x with $\|x\|_\infty = 1$ such that

$$(Ax)_j = \sum_{i=1}^{n} |A(j,i)|$$

where $(Ax)_j$ is entry j in the vector Ax.

(c) Using the following two results prove that

$$\|A\|_\infty = \max_{j=1,\ldots,n} \sum_{i=1}^{n} |A(j,i)|.$$

Question 2 solutions

First, by explicit computation, we have that

$$(Ax)_j = \sum_{i=1}^{n} A(j,i) x_i. \tag{1}$$

(a) From (1) we and the definition of the vector norm $\|\cdot\|_\infty$ we have that

$$\begin{aligned}\|Ax\|_\infty &= \max_{j=1,\dots,n}\left|\sum_{i=1}^n A(j,i)x_i\right| \\ &\le \max_{j=1,\dots,n}\sum_{i=1}^n |A(j,i)x_i| \\ &\le \max_{j=1,\dots,n}\sum_{i=1}^n |A(j,i)||x_i| \\ &\le \max_{j=1,\dots,n}\sum_{i=1}^n |A(j,i)|,\end{aligned}$$

where we have repeatedly used the triangle inequality in the first step and the fact that $|x_i| \le 1$ for $i = 1, \dots, n$ since $\|x\|_\infty \le 1$.

(b) Fix an index j and let

$$x_i = \operatorname{sign}(A(j,i)).$$

This means that if $x_j = 1$ if $A(j,i) > 0$ and $x_j = -1$ if $A(j,i) < 0$, if $A(j,i) = 0$ we will simply set $x_j = 0$. Importantly, for this x we have that

$$(Ax)_j = \sum_{i=1}^n A(j,i)\operatorname{sign}(A(j,i)),$$

and since $A(j,i)\operatorname{sign}(A(j,i)) = |A(j,i)|$ we get that for this choice of x

$$(Ax)_j = \sum_{i=1}^n |A(j,i)|.$$

(c) From part (a) we have that for any x with $\|x\|_\infty \le 1$

$$\|Ax\|_\infty \le \max_{j=1,\dots,n}\sum_{i=1}^n |A(j,i)|.$$

Therefore we can immediately conclude that

$$\|A\|_\infty = \max_{\|x\|_\infty=1}\|Ax\|_\infty \le \max_{j=1,\dots,n}\sum_{i=1}^n |A(j,i)|. \tag{2}$$

Let j' be the index that solves

$$\max_{j=1,\dots,n}\sum_{i=1}^n |A(j,i)|,$$

from part (b) we know there is an x with $\|x\|_\infty$ such that

$$(Ax)_{j'} = \sum_{i=1}^n |A(j',i)|.$$

This implies that

$$\max_{\|x\|_\infty=1}\|Ax\|_\infty \ge \max_{j=1,\dots,n}\sum_{i=1}^n |A(j,i)|, \tag{3}$$

since the maximum over all possible x with $\|x\|_\infty = 1$ is definitely greater than or equal to the value for any specific x with $\|x\|_\infty = 1$. Combining (3) and (2) concludes the proof.

Question 3

We have spent a fair bit of time discussing the SVD, nevertheless there remain additional interpretations to explore. As we discussed in class a square matrix $A \in \mathbb{R}^{n\times n}$ is singular if at least one of its singular values is zero. It turns out that the smallest singular value of a matrix A actually tells us how close we are to a singular matrix, in other words what the smallest perturbation to A is that can make it singular—this seems useful to know.

(a) Given two vectors x and y with $\|x\|_2 = 1$ and $\|y\|_2 = 1$ prove that

$$\|xy^T\|_2 = 1.$$

(b) Given a non-singular matrix A with SVD $A = U\Sigma V^T$ show that there is a rank-one perturbation E such that $A + E$ is singular and $\|E\|_2 = \sigma_n$.

(c) Prove that given any matrix E with $\|E\|_2 < \sigma_n$ $A+E$ is non-singular. Here, σ_n is the smallest singular value of A. (Hint: we talked about how the largest singular value of a matrix bounds how big A can make a vector with unit norm; can you cook up a lower bound on how small A can make a vector with unit norm in terms of singular values?)

Question 3 solution

(a) There are many ways to approach this problem. One is to observe that xy^T is essentially its own reduced SVD with $\sigma_1 = 1$ and therefore $\|xy^T\|_2 = 1$. More formally we have that

$$\begin{aligned}\|xy^T\|_2 &= \max_{\|z\|_2=1} \|xy^T z\|_2 \\ &= \max_{\|z\|_2=1} |y^T z| \|x\|_2 \\ &= \max_{\|z\|_2=1} |y^T z|.\end{aligned}$$

Similar to Question 2 taking $z = y$, which satisfies $\|z\|_2 = 1$, yields $\|xy^T z\|_2 = \|y\|_2^2 = 1$. Paired with the Cauchy-Schwartz inequality $|y^T z| \leq \|y\|_2\|z\|_2$ we conclude that $\|xy^T\|_2 = 1$.

(b) Let $A = U\Sigma V^T$ be the SVD of A. Rewritten as

$$A = \sum_{i=1}^{n} \sigma_i u_i v_i^T$$

we see that if $E = -\sigma_n u_i v_i^T$

$$A + E = \sum_{i=1}^{n-1} \sigma_i u_i v_i^T.$$

This shows that $A + E$ has an SVD with one singular value equal to zero and is therefore singular. From part (a) we have that $\| - \sigma_n u_i v_i^T\|_2 = \sigma_n \|u_i v_i^T\|_2 = \sigma_n$.

(c) One way to show that $A + E$ is non-singular is to show that the only vector in the null space is zero. To accomplish this consider any vector $\widehat{x} \neq 0$, we will show that

$$A\widehat{x} + E\widehat{x} \neq 0.$$

First, let $x = \widehat{x}/\|\widehat{x}\|_2$, since

$$\widehat{x}(Ax + Ex) \neq 0$$

if and only if $Ax + Ex \neq 0$ we may restrict our discussion to unit length vectors.

Now, for any unit length vector x we know that $\|Ex\|_2 < \sigma_n$ courtesy of our assumptions. Similarly, based on our pictorial representation of the SVD we have that $\|Ax\|_2 \geq \sigma_n$. To state this formally, observe that

$$\begin{aligned}
\|Ax\|_2 &= \|U\Sigma V^T x\|_2 \\
&= \|\Sigma V^T x\|_2 \\
&= \left(\sum_{i=1}^{n} \sigma_i v_i^T x\right)^{1/2} \\
&\geq \sigma_n \left(\sum_{i=1}^{n} v_i^T x\right)^{1/2} \\
&\geq \sigma_n \|V^T x\|_2 \\
&\geq \sigma_n \|x\|_2.
\end{aligned}$$

Finally, since $\|Ax\|_2 \geq \sigma_n$ and $\|Ex\|_2 < \sigma_n$ clearly $Ax + Ex \neq 0$.