Linear Algebra

Eigenvalues and eigenvectors

This is a large and difficult topic, in general. Some of the methods rely on lots of theory. We look primarily at a couple of ideas.

Reminders: λ is an eigenvalue of A if there is a non-zero vector $\boldsymbol{x}$ such that $A\boldsymbol{x} = \lambda\boldsymbol{x}$. The point then is that in vector/matrix equations, the linear operator that corresponds to the matrix A can be replaced, when operating on eigenvectors, by multiplication by λ [which, on the whole, we can expect to understand rather better].

The theory tells us that if $A\boldsymbol{x} = \lambda\boldsymbol{x}$, then $(A - \lambda I)\boldsymbol{x} = \boldsymbol{0}$, and so $A - \lambda I$ cannot have an inverse, so it is singular, so its determinant is zero. For 2×2, even 3×3, matrices, this is fine, the equation $\det(A - \lambda I) = 0$ is quadratic or cubic, and we [or Maple] can solve for λ. Don't be tempted to try this for large matrices; it's nice in theory, but it's a lot of work, and it gives us a high-degree polynomial to solve. High-degree polynomials are not warm, cuddley things.

If we *did* get a high-degree polynomial, we would be looking at an iterative process for solving it; so we equally need an iterative process for finding eigenvalues and eigenvectors. As always, some iterations work better than others ….

Preliminaries

Suppose A has eigenvalues $\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_n$, with corresponding eigenvectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \ldots, \boldsymbol{x}_n$. [Left as niggle: what if some of the eigenvalues are 'repeated'?] We need two basic results:

(a) If $A\boldsymbol{x} = \lambda\boldsymbol{x}$, then $A^2\boldsymbol{x} = \lambda A\boldsymbol{x} = \lambda^2\boldsymbol{x}$, and more generally, $A^k\boldsymbol{x} = \lambda A^{k-1}\boldsymbol{x} = \ldots = \lambda^k\boldsymbol{x}$. So $\boldsymbol{x}_i$ is also an eigenvector of A^k with eigenvalue λ_i^k. [Even if $k = 0$.]

More generally still, $\boldsymbol{x}_i$ is also an eigenvector of $f(A)$ with eigenvalue $f(\lambda_i)$, for 'reasonable' functions f.

(b) If B is a non-singular matrix, then $BAB^{-1}B\boldsymbol{x}_i = BAI\boldsymbol{x}_i = BA\boldsymbol{x}_i = \lambda_i B\boldsymbol{x}_i$, so $B\boldsymbol{x}_i$ is an eigenvector of BAB^{-1} with eigenvector [still] λ_i.

The real point is that if we can find the eigenvalues/vectors of BAB^{-1}, then the eigenvalues of A are the same, and we can get to the eigenvectors by pre-multiplying by B^{-1}.

The usual idea is to build up B as a product $R_1R_2R_3\ldots$ of 'rotation matrices' or similar, each of which can easily be inverted, in such a way that BAB^{-1} gradually gets simpler, to the point where we can write down its eigenvalues, and, if we need them, its eigenvectors. For example, if BAB^{-1} is a diagonal matrix, then its eigenvectors are the diagonal elements.

For a 2×2 matrix, such a rotation matrix looks like

$$R = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

[with inverse obtained by replacing θ by $-\theta$]. More generally, we start from an identity matrix, and replace two of the 1's down the long diagonal with $\cos\theta$, and then replace the two zeros completing the square with $\pm\sin\theta$, representing a rotation in the plane of the corresponding two dimensions by θ. We choose θ so as to create some zeros in RAR^{-1}; it's a bit like doing an n-dimensional Rubik's Cube, gradually rotating A into a nice 'shape'.

Another similar method is to use matrices P of form $P = I - 2\boldsymbol{w}\boldsymbol{w}'$, where $\boldsymbol{w}$ is chosen so that P is easy to invert and also so that PAP^{-1} has lots of zeros.

Details are beyond the scope of this module; in real life, you will use a library procedure rather than write your own. Maple can find eigenvalues and eigenvectors for you. Most books on numerical analysis include a chapter on these processes; key names include Householder, Givens, Hessenberg, Wilkinson.

These are the preferred ways of finding eigenvalues and eigenvectors for serious professional work, especially with large matrices. However, there is also a fairly simple method for finding eigenvalues and eigenvectors in some cases, which uses the other preliminary result.

The Power Method

Suppose we have our eigenvalues ordered such that $|\lambda_1| > |\lambda_2| > |\lambda_3| > \ldots > |\lambda_n|$, and suppose we choose any old [non-zero] start vector, $\boldsymbol{x}$. Then we can write $\boldsymbol{x}$ as a linear combination of eigenvectors:

$$\boldsymbol{x} = a_1\boldsymbol{x}_1 + a_2\boldsymbol{x}_2 + a_3\boldsymbol{x}_3 + \ldots + a_n\boldsymbol{x}_n .$$

Now what happens if we multiply by A? Because the $\boldsymbol{x}_i$ are eigenvectors, they just get multiplied by λ_i:

$$A\boldsymbol{x} = a_1\lambda_1\boldsymbol{x}_1 + a_2\lambda_2\boldsymbol{x}_2 + \ldots + a_n\lambda_n\boldsymbol{x}_n .$$

And if we keep multiplying by A,

$$A^k\boldsymbol{x} = a_1\lambda_1^k\boldsymbol{x}_1 + a_2\lambda_2^k\boldsymbol{x}_2 + \ldots + a_n\lambda_n^k\boldsymbol{x}_n .$$

Now the clever bit. The RHS is

$$a_1\lambda_1^k(\boldsymbol{x}_1 + c_{2,k}\boldsymbol{x}_2 + \ldots + c_{n,k}\boldsymbol{x}_n),$$

where $c_{i,k} = a_i\lambda_i^k / a_1\lambda_1^k$. Now, we don't know what a_i, λ_i, a_1 or λ_1 are; but we don't need to. By assumption, $|\lambda_1| > |\lambda_i|$, so as k gets large, $c_{i,k} \to 0$ [as long as we don't quibble about $a_1 = 0$]. So, for large k,

$$A^k\boldsymbol{x} \approx a_1\lambda_1^k\boldsymbol{x}_1 .$$

We still don't know a_1, but that doesn't matter as $a_1\boldsymbol{x}_1$ is just as good an eigenvector as $\boldsymbol{x}_1$; and we can estimate λ_1 by comparing components in the resulting vector when we multiply by A again.

[In practice, we scale $A^k\boldsymbol{x}$ to have, eg, a first component of 1, then when we multiply by A, the new first component is an estimate for λ_1.]

Example

Suppose

$$A = \begin{pmatrix} -4 & 14 & 0 \\ -5 & 13 & 0 \\ -1 & 0 & 2 \end{pmatrix}.$$

Start with $\boldsymbol{x} = (1,1,1)'$. Then $A\boldsymbol{x} = (10,8,1)'$, so our first estimate for λ_1 is 10, and for $\boldsymbol{x}_1$ is $(1, 0.8, 0.1)'$ [scaling back to a first component of 1].

Then our next estimate is $A\boldsymbol{x}_1 = (-4+14\times0.8, -5+13\times0.8, -1+2\times0.1)' = (7.2, 5.4, -0.8)'$, so our second guess is $\lambda_1 = 7.2$, $\boldsymbol{x}_1 = (1, 5.4/7.2, -0.8/7.2)' = (1, 0.75, -0.11)'$.

Exercise: Confirm the next approximation, 6.5, and $(1, 0.731, -0.188)'$. [The exact eigenvalue is 6, with eigenvector $1, \frac{5}{7}, -\frac{1}{4})'$.] [Aitken's process can, as usual, be used to speed up the iteration.]

What about the other eigenvalues and eigenvectors? Well, by the [way] above result, the matrix $(A-\mu I)^{-1}$ has the same eigenvectors as A, and eigenvalues $1/(\lambda_i - \mu)$; so we can make any of the eigenvalues the largest as long as we know roughly where it is [so that we can make μ an approximation to it]. The bigger the discrepancy between the biggest and next biggest eigenvalues, the better.

Note that there are problems if two eigenvalues are of equal size; eg if they *are* equal, or, more likely for random real matrices, if they are complex conjugates. Details left as an exercise!