

The Power Method

Assume that $A \in \mathbb{C}^{n \times n}$ has a n linearly independent eigenvectors v_1, v_2, \dots, v_n . Then any $x \in \mathbb{C}^n$ can be represented uniquely as

$$x = \sum_{i=1}^n c_i v_i. \quad (1)$$

Here we are interested in what (if any) direction $A^k x$ heads toward as $k \rightarrow \infty$. Specifically, we have a sequence $\{x_k\}$ of vectors defined by

$$x_0 = x, \quad x_k = Ax_{k-1} = A^k x_0, \quad k = 1, 2, 3, \dots \quad (2)$$

and we would like to know in what direction it is ultimately pointing.

Recall that if v_i is an eigenvector of A , then there is a scalar λ_i , called an eigenvalue, for which $Av_i = \lambda_i v_i$. Then $A^k v_i = \lambda_i^k v_i$ (you do the induction). Using (1) (and linearity) we find that

$$A^k x = \sum_{i=1}^n c_i \lambda_i^k v_i. \quad (3)$$

Now suppose that $|\lambda_1| > |\lambda_i|$, $i = 2, 3, \dots, n$. Then

$$\frac{A^k x}{\lambda_1^k} = c_1 v_1 + \sum_{i=2}^n c_i \left(\frac{\lambda_i}{\lambda_1}\right)^k v_i \quad (4)$$

Here it is clear (yes?) that unless $c_1 = 0$, $A^k x \rightarrow \text{span}\{v_1\}$. Thus we call v_1 the *dominant eigenvector* of A . This result is as simple as it is powerful: if v_1 is the dominant eigenvector of A , then for *almost all* $x \in \mathbb{C}^n$,

$$x \rightarrow v_1$$

under repeated application of A .

(If this is too analytic for your taste, then change to the basis $\{v_1, v_2, \dots, v_n\}$. Under this basis A has coordinates $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$, and $\Lambda^k y \rightarrow \text{span}\{e_1\}$ as long as $y(1) \neq 0$.)

The *power method* consists of scaling iteration (2) to avoid underflow or overflow, and figuring out when to stop. We solve both problems by approximating λ_1 at each step. The code below (if it terminates) gives a small backward error (i.e. gives an eigenpair for a matrix “close” to A).

```
i = argmax(|x|)
x = x/x(i)
For k = 1, 2, ... until done
  y = Ax
  i = argmax(|y|)
  λ = y(i)
  r = y - λx, if ||r|| is small enough, then stop
  x = y/λ
```