POTENSMETODEN (SEC. 5.8)

Vi har set at det er meget viktigt at kunne regne ut egenverdier til en $n \times n$ matrise A. Hvordan gør computeren det? Vi lærer nu en metode som kan brukes for at regne ut en *dominant* egenverdi (hvis den finnes) og en tilhørende egenvektor. Metoden kalles *potensmetoden*. Selv om det lyder meget spesiellt finnes der mange anvendelser for denne metode (og vi ser en viktigt feld av anvendelser i seksjon 5.9.). I praksis brukes potensmetoden (og mere generelle varianter) for store glissen matriser (dvs. store matriser hvor de fleste indganger er like null). For eksempel bruker Google's PageRank algoritmen denne metode. Vi starter med en definisjon:

Definisjon. *Vi sier at en matrise A har en* dominant egenverdi *hvis der finnes en egenverdi λ_1 som oppfyller $|\lambda_1| > |\lambda_i|$ for alle andre egenverdier λ_i for A.*

Der finnes matriser som ikke har en dominant egenverdi, som for eksemplen

$$A = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

Potensmetoden kommer ikke til at virke i denne tilfælder. Men der finnes mange problemer i praksis hvor man ved i forvejen at en matrise har en dominant egenverdi og hvor man vil finne kun den.

Hva er ideen av potensmetoden? Anta at A er en $n \times n$ matrise som er diagonaliserbar (for at gør det nemt) og som har egenverdier $\lambda_1, \dots, \lambda_n$ med tilsvarende egenvektorer $\overrightarrow{v}_1, \dots, \overrightarrow{v}_n \in \mathbb{R}^n$ slik at

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_n|.$$

Hvert vektor $\overrightarrow{x} \in \mathbb{R}^n$ kan skrives som

$$\overrightarrow{x} = c_1 \overrightarrow{v}_1 + c_2 \overrightarrow{v}_2 + \cdots + c_n \overrightarrow{v}_n,$$

for tall $c_1, c_2, \dots, c_n$. La oss anta at $\overrightarrow{x}$ er en vektor sådan at $c_1 \neq 0$. Hva sker der hvis vi multipliserer $\overrightarrow{x}$ med A flere gang? La oss regne de ut:

$$\begin{aligned}
A\overrightarrow{x} &= c_1 A\overrightarrow{v}_1 + c_2 A\overrightarrow{v}_2 + \cdots + c_n A\overrightarrow{v}_n = c_1\lambda_1\overrightarrow{v}_1 + c_2\lambda_2\overrightarrow{v}_2 + \cdots + c_n\lambda_n\overrightarrow{v}_n \\
A^2\overrightarrow{x} &= c_1 A^2\overrightarrow{v}_1 + c_2 A^2\overrightarrow{v}_2 + \cdots + c_n A^2\overrightarrow{v}_n = c_1\lambda_1^2\overrightarrow{v}_1 + c_2\lambda_2^2\overrightarrow{v}_2 + \cdots + c_n\lambda_n^2\overrightarrow{v}_n \\
&\vdots \\
A^k\overrightarrow{x} &= c_1 A^k\overrightarrow{v}_1 + \cdots + c_n A^k\overrightarrow{v}_n = c_1\lambda_1^k\overrightarrow{v}_1 + c_2\lambda_2^k\overrightarrow{v}_2 + \cdots + c_n\lambda_n^k\overrightarrow{v}_n
\end{aligned}$$

Vi ser at koeffisienten $c_1\lambda_1^k$ til vektorern $\overrightarrow{v}_1$ bliver meget større i absolutverdien enn de andre koeffisienter. For at se det lidt bedre kan vi dividere med λ_1^k for at få

$$\frac{1}{\lambda_1^k} A^k \overrightarrow{x} = c_1\overrightarrow{v}_1 + c_2\left(\frac{\lambda_2}{\lambda_1}\right)^k \overrightarrow{v}_2 + \cdots + c_n\left(\frac{\lambda_n}{\lambda_1}\right)^k \overrightarrow{v}_n.$$

Nu ser vi for hvert $i \in \{2, \dots, n\}$ at

$$\left(\frac{\lambda_i}{\lambda_1}\right)^k \to 0,$$

hvis $k \to \infty$, fordi

$$\left|\frac{\lambda_i}{\lambda_1}\right| < 1.$$

Vi konkluderer at

$$\frac{1}{\lambda_1^k} A^k \overrightarrow{x} \to c_1 \overrightarrow{v}_1,$$

hvis $k \to \infty$. Fordi multiplikasjon med $\frac{1}{\lambda_1^k}$ ikke ændrer retningen av en vektor, viser det at retningen av vektorerne $A^k \overrightarrow{x}$ er næsten den sammen retning enn den dominante egenvektor $\overrightarrow{v}_1$ (eller $-\overrightarrow{v}_1$) hvis k er stor og hvis $c_1 \neq 0$.

Eksempel. La

$$A = \begin{pmatrix} 1.8 & 0.8 \\ 0.2 & 1.2 \end{pmatrix},$$

og

$$\overrightarrow{v}_1 = \begin{pmatrix} 4 \\ 1 \end{pmatrix} \quad \text{og} \quad \overrightarrow{x}_0 = \begin{pmatrix} -0.5 \\ 1 \end{pmatrix}.$$

Vi kan sjekke at $\overrightarrow{v}_1$ er en egenvektor til den dominante egenverdi $\lambda_1 = 2$ for A. Hva sker der når vi regner ut $A^k \overrightarrow{x}_0$ for $k = 0, \ldots, 8$? Vi kan regne ut at

$$\begin{aligned} A\overrightarrow{x}_0 &= \begin{pmatrix} -0.1 \\ 1.1 \end{pmatrix} \\ A^2\overrightarrow{x}_0 &= \begin{pmatrix} 0.7 \\ 1.3 \end{pmatrix} \\ A^2\overrightarrow{x}_0 &= \begin{pmatrix} 0.7 \\ 1.3 \end{pmatrix} \\ A^3\overrightarrow{x}_0 &= \begin{pmatrix} 2.3 \\ 1.7 \end{pmatrix} \\ &\vdots \\ A^8\overrightarrow{x}_0 &= \begin{pmatrix} 101.5 \\ 26.5 \end{pmatrix}. \end{aligned}$$

Vi kan se at

$$\begin{pmatrix} 101.5 \\ 26.5 \end{pmatrix} = 26.5 \begin{pmatrix} 3.8302 \\ 1 \end{pmatrix},$$

og det peger næsten i den sammen rettning enn egenvektoren $\overrightarrow{v}_1$.

Tallene i eksemplen (og i praksis generelt) vokser eksponensjelt! Derfor er det viktigt at "normere" vektorer så at tallene bliver på sammen størrelse (ellers ville det gå gæld med numeriken). Fordi vi er kun interessert i retningen av en dominant egenvektor må vi gerne multiplisere vektorerne med tall mens man bruker potensmetoden. En mulighed for at normere vektorer er at multiplisere med $\frac{1}{\lambda_1^k}$ for den dominant egenverdi λ_1 like som vi har gjort før. I praksis kan vi selvfølgelig ikke gør det fordi vi ikke kenner λ_1 (det vil vi jo regne ut). En annen mulighed er at dele vektoren i hvert trinn på den indgang med største absolutverdi bland ingangerne. La oss se det i et eksempel:

Eksempel. La

$$A = \begin{pmatrix} 6 & 5 \\ 1 & 2 \end{pmatrix},$$

og

$$\overrightarrow{x}_0 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

For hvert $k \in \mathbb{N}$ definerer vi rekursivt

$$\begin{aligned} \mu_k &= \text{ Indgang av } A\overrightarrow{x}_k \text{ med største absolutverdi} \\ \overrightarrow{x}_{k+1} &= \frac{A\overrightarrow{x}_k}{\mu_k}. \end{aligned}$$

Hvilke vektorer får vi her:

$$A\overrightarrow{x}_0 = \begin{pmatrix} 5 \\ 2 \end{pmatrix} \rightsquigarrow \mu_0 = 5.$$

Derfor har vi

$$\overrightarrow{x}_1 = \frac{A\overrightarrow{x}_0}{\mu_0} = \frac{1}{5}\begin{pmatrix} 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0.4 \end{pmatrix}.$$

Næste trinn:

$$A\overrightarrow{x}_1 = \begin{pmatrix} 8 \\ 1.8 \end{pmatrix} \rightsquigarrow \mu_1 = 8.$$

Derfor har vi

$$\overrightarrow{x}_2 = \frac{A\overrightarrow{x}_1}{\mu_1} = \frac{1}{8}\begin{pmatrix} 8 \\ 1.8 \end{pmatrix} = \begin{pmatrix} 1 \\ 0.225 \end{pmatrix}.$$

Næste trinn:

$$A\overrightarrow{x}_2 = \begin{pmatrix} 7.125 \\ 1.450 \end{pmatrix} \rightsquigarrow \mu_2 = 7.125.$$

Derfor har vi

$$\overrightarrow{x}_3 = \frac{A\overrightarrow{x}_1}{\mu_2} = \begin{pmatrix} 1 \\ 0.2035 \end{pmatrix}.$$

Hvis vi fortsætter finner vi

$$\begin{aligned} \mu_3 = 7.0175, \quad \overrightarrow{x}_4 &= \begin{pmatrix} 1 \\ 0.2005 \end{pmatrix} \\ \mu_4 = 7.0025, \quad \overrightarrow{x}_5 &= \begin{pmatrix} 1 \\ 0.20007 \end{pmatrix}. \end{aligned}$$

Det ser ut som μ_k konvergerer imod 7 og $\overrightarrow{x}_k$ imod

$$\overrightarrow{v}_1 = \begin{pmatrix} 1 \\ 0.2 \end{pmatrix}.$$

Like som forventet kan vi sjekke at $\overrightarrow{v}_1$ er en egenvektor til egenverdien 7!

Metoden som fungerte i eksemplen fungerer også generellt:

Algoritme (Potensmetoden)**.**
Anta at matrisen A har en dominante egenverdi.

(1) Valge inisjalvektor $\overrightarrow{x}_0$ med største indgang 1.
(2) For $k = 0, 1, \dots$ iterere
   (a) Regne ut $A\overrightarrow{x}_k$.
   (b) La μ_k være indgang av $A\overrightarrow{x}_k$ med største absolutverdi.
   (c) Sætte
$$\overrightarrow{x}_{k+1} = \frac{A\overrightarrow{x}_k}{\mu_k}.$$

Hvis inisjalvektoren har en komponent i egenrommet til den dominante egenverdi, så konvergerer μ_k imod den dominate egenverdi for A, og $\overrightarrow{x}_k$ imod en tilsvarende egenvektor.

Potensmetoden virker for alle matriser som har en dominante egenverdi. Det virker også hvis A er ikke diagonaliserbar! Bevisen for det bruker lidt mere avancerte lineær algebra tekniker enn hva vi har lært intil nu. I praksisen er det meget usansynlig at inisjalvektoren har ingen komponent i egenrommet til den dominante egenverdi. Hvis man bruker floating point arithmetik (f.eks. i Matlab) så hjælper numeriske fejl med det, og potensmetoden kommer til at konvergere næsten altid. Konvergenshastighet avhenger av separasjonen mellom den dominante egenverdi og egenverdier med mindre absolutverdi. Det kan nogen gang være langsom i praksis og der finnes mere avancerte metoder for at gør potensmetoder hurtiger.

Inverse potensmetode. Potensmetoden kan brukes for at finne en dominant egenverdi for en matrise. Der er også situasjoner hvor man vil finne andre egenverdier som er ikke dominant. Der er en variasjon av potensmetoden som kan løse dette problem hvis man har en approximasjon a egenverdier man vil gerne finne. Vi starter med et teorem:

Teorem. *La B være en inverterbar $n \times n$ matrise med egenverdier $\lambda_1, \ldots, \lambda_n$. Egenverdier for B^{-1} er $\lambda_1^{-1}, \ldots, \lambda_n^{-1}$.*

Bevis. Anta at $\overrightarrow{v}_i$ er en egenvektor til λ_i for B sådan at

$$B\overrightarrow{v}_i = \lambda_i \overrightarrow{v}_i.$$

Fordi B er inverterbar har vi $\lambda_i \neq 0$. Hvis vi multipliserer denne likning med B^{-1} fra begge sider finner vi

$$\overrightarrow{v}_i = \lambda_i B^{-1} \overrightarrow{v}_i,$$

og vi konkluderer at

$$B^{-1}\overrightarrow{v}_i = \lambda_i^{-1} \overrightarrow{v}_i.$$

Det betyder at λ_i^{-1} er en egenverdi for B^{-1}. □

Hvordan kan vi bruke denne teorem for at finne egenverdier? La oss anta at A er en $n \times n$ matrise med egenverdier $\lambda_1, \ldots, \lambda_n$. Se på matrisen

$$B = (A - \alpha I_n)^{-1},$$

for en α slik at $A - \alpha I_n$ er inverterbar. Teoremet viser at egenverdier for B er

$$(\lambda_1 - \alpha)^{-1}, \ldots, (\lambda_n - \alpha)^{-1}.$$

Hvis nu α er tætter på en av egenverdierne λ_i enn på de andre så bliver tilsvarende egenverdi $(\lambda_i - \alpha)^{-1}$ for B dominant! Det betyder at vi kan bruke potensmetoden for B og finne egenvektorer for A. For at bruke potensmetoden skal vi multiplisere B med vektorer. Husk at man kan gøre det med at løse et likningssystem. Generellt har vi

$$\overrightarrow{y} = (A - \alpha I_n)^{-1}\overrightarrow{x} \quad \text{hvis og kun hvis} \quad (A - \alpha I_n)\overrightarrow{y} = \overrightarrow{x}.$$

Det er ofte bedre numerisk at løse likningssystemet enn at invertere matrisen først.

Algoritme (Inverse potensmetoden)**.**
Anta at matrisen A har en egenverdi λ som er ikke degenerert.

(1) Valge α som er nær en av egenverdiene λ for A
(2) Valge $\overrightarrow{x}_0$ med største indgang 1.
(3) For $k = 0, 1, \ldots$ iterere
   (a) Løse likningen $(A - \alpha I_n)\overrightarrow{y}_k = \overrightarrow{x}_k$.
   (b) La μ_k være indgang av $\overrightarrow{y}_k$ med største absolutverdi.
   (c) Sætte
   $$\overrightarrow{x}_{k+1} = \frac{\overrightarrow{y}_k}{\mu_k}.$$
   (d) Regne ut $\nu_k = \alpha + \frac{1}{\mu_k}$.

Hvis α er tæt nok på λ og inisjalvektoren har en komponent i egenrommet til den egenverdi, så konvergerer ν_k imod λ, og $\overrightarrow{x}_k$ imod en tilsvarende egenvektor.