Un esempio: grafi, matrici sparse, autovalori ed autovettori

Salvatore Filippone
salvatore.filippone@uniroma2.it

Teorema

Perron-Frobenius: Data una matrice stocastica (transizioni di stato di una catena di Markov)

$$e^T Q = e^T, \qquad Q \geq 0 \qquad e = (1, 1, 1 \dots 1)^T$$

se Q irriducibile, allora

1. $\lambda = 1$ *è l'autovalore dominante (tutti gli altri sono più piccoli in modulo);*
2. *Esiste ed è unico l'autovettore $r > 0$ per l'autovalore dominante $\lambda = 1$*

$$Qr = \lambda r = r;$$

3. *r fornisce la distribuzione di probabilità stazionaria della catena di Markov*

Teorema

Perron-Frobenius: Data una matrice stocastica (transizioni di stato di una catena di Markov)

$$e^T Q = e^T, \qquad Q \geq 0 \qquad e = (1, 1, 1 \ldots 1)^T$$

se Q irriducibile, allora

1. $\lambda = 1$ *è l'autovalore dominante (tutti gli altri sono più piccoli in modulo);*
2. *Esiste ed è unico l'autovettore $r > 0$ per l'autovalore dominante $\lambda = 1$*

$$Qr = \lambda r = r;$$

3. *r fornisce la distribuzione di probabilità stazionaria della catena di Markov*

A quale applicazione stiamo pensando?

L'autovettore da $ 25.000.000.000

L'autovettore da \$ 25.000.000.000

Google = crawling + matching + PageRank

L'autovettore da \$ 25.000.000.000

$$\text{Google} = \text{crawling} + \text{matching} + \text{PageRank}$$

Criterio di rilevanza di una pagina WEB:

La rilevanza della pagina i è la media pesata della rilevanza di tutte le pagine j che rinviano a i

$$r_i = \sum_j \frac{r_j}{N_j}$$

L'autovettore da \$ 25.000.000.000

$$\text{Google} = \text{crawling} + \text{matching} + \text{PageRank}$$

Criterio di rilevanza di una pagina WEB:

La rilevanza della pagina i è la media pesata della rilevanza di tutte le pagine j che rinviano a i

$$r_i = \sum_j \frac{r_j}{N_j}$$

Algoritmo di PageRank:

- Crea una lista di tutte le pagine WEB e dei loro collegamenti (grafo di connettività di WWW);
- Assegna un peso a ciascun collegamento e costruisci la matrice Q;
- Trova l'autovettore r;

Il valore dell'autovettore r nella posizione della pagina i viene assunto come misura della autorevolezza della pagina stessa.

L'autovettore da $ 25.000.000.000

Costruzione della matrice:

$$Q_{ij} = \begin{cases} 1/N_j & \text{se la pagina J manda alla I} \\ 0 & \text{altrimenti} \end{cases}$$

L'autovettore da \$ 25.000.000.000

Costruzione della matrice:

$$Q_{ij} = \begin{cases} 1/N_j & \text{se la pagina J manda alla I} \\ 0 & \text{altrimenti} \end{cases}$$

Q va corretta con il "teletrasporto" che modella il fatto che se da una pagina non ci sono link uscenti, allora saltiamo (arbitrariamente) ad un'altra pagina a caso (e questo fra l'altro preserva la irriducibilità della matrice, ossia la connettività del grafo associato).

L'autovettore da $ 25.000.000.000

Quindi: al cuore di Google c'è il calcolo di un autovettore di dimensione gigantesca;

L'autovettore da \$ 25.000.000.000

Quindi: al cuore di Google c'è il calcolo di un autovettore di dimensione gigantesca; si calcola con il *metodo delle potenze*

Algorithm 2: Metodo delle potenze

$r^{(0)} \leftarrow r_0$;
for $k = 1, \ldots$ **do**
  $q^{(k)} \leftarrow Ar^{(k-1)}$;
  $r^{(k)} \leftarrow q^{(k)} / \|q^{(k)}\|$

- L'autovalore è noto a priori e vale 1;
- Richiede alcune decine di iterazioni, anche diversi giorni di calcolo!
- Se il vettore r_0 è normalizzato, allora si può risparmiare il passo di normalizzazione nel caso in cui $A = Q$ è stocastica.

Il metodo delle potenze

Se diagonalizzabile, la matrice Q si può decomporre in una somma di componenti ortogonali

$$Q = \sum_i \lambda_i Q_i$$

da cui si ottiene che moltiplicando ripetutamente per Q

$$Q^k v = \sum_i \lambda_i^k Q_i v = \sum_i \lambda_i^k v_i$$

In questa presentazione abbiamo assunto per semplicità che la matrice sia diagonalizzabile, ma il metodo vale in ogni caso.
Se $|\lambda_i| < 1, i > 1$ allora v converge a $v_1 = r$.

Considerando il teletrasporto, stiamo effettivamente calcolando:

$$A = \alpha Q + (1-\alpha)\frac{1}{n}ee^T$$

con (tipicamente) $\alpha = 0.85$ (che migliora la velocità di convergenza).
Inoltre, non ci interessa (e sarebbe impossibile) costruire esplicitamente ee^T; si può invece procedere con

$$y = Az = \alpha Qz + (1-\alpha)ee^T z = aQz + \beta e$$

con

$$\beta = (1-\alpha)e^T z$$

Infine, possiamo aggiungere un vettore di "teletrasporto" personalizzato (e pagato profumatamente)

$$Q + \frac{1}{n}ed^T$$

che si gestisce tranquillamente nello stesso modo

L'autovettore da $ 25.000.000.000

Da dove viene α?

Da dove viene α?

Q è stocastica

Quindi $\lambda_1 = 1$, e se irriducibile allora $|\lambda_i| < 1$, $i > 1$, e quindi le componenti successive si smorzano facendo sopravvivere il primo autovettore v_1

Teorema

Se gli autovalori di Q sono $\{1, \lambda_2, \lambda_3, \ldots, \lambda_n\}$, allora gli autovalori di $A = \alpha Q + (1-\alpha)\frac{1}{n}ee^T$ sono

$$\{1, \alpha\lambda_2, \alpha\lambda_3, \ldots, \alpha\lambda_n\}$$

E quindi il coefficiente $\alpha = 0.85$ controlla il secondo autovalore di Q, che nel caso peggiore vale $\lambda_2 = 1$, e quindi governa la convergenza (con 60 iterazioni il contributo del secondo autovalore si riduce a 10^{-4}).

L'autovettore da \$ 25.000.000.000

P.S.: \$ 25.000.000.000 era il valore della offerta pubblica iniziale (IPO) delle azioni di Google