

# Eigenvektor Zentralitäts Algorithmus für ungerichtete Graphen basierend auf Power Iteration

Robin Schmöcker

Januar 2020

## Contents

<b>1 Abstract</b>	<b>1</b>
<b>2 Einleitung</b>	<b>1</b>
<b>3 Grundlagen</b>	<b>2</b>
3.1 Begriffe und Notationen . . . . .	2
3.2 Eigenvektor Zentralität [4] . . . . .	3
3.3 Power Iteration [2] . . . . .	4
3.3.1 Wahl Initialvektor . . . . .	4
3.3.2 Probleme . . . . .	4
<b>4 Algorithmus</b>	<b>5</b>
4.1 Pseudocode . . . . .	5
4.2 Beispiel . . . . .	6
4.3 Korrektheit . . . . .	8

## 1 Abstract

Dieser Artikel stellt einen Eigenvektor Zentralitäts Algorithmus für ungerichtete Graphen vor, welcher im Gegensatz zu Power Iteration auch mit bipartiten Graphen umgehen kann.

## 2 Einleitung

Eigenvektor Zentralität ist ein Maß der sozialen Netzwerkanalyse, um den Einfluss eines Knotens in einem Netzwerk zu messen. Bei der Eigenvektor Zentralität haben die Knoten einen hohen Wert, welche selbst mit vielen Knoten verbunden sind, die ebenfalls eine hohe Eigenvektor Zentralität haben.

Der trivialste Algorithmus, um dieses Maß zu berechnen ist Power Iteration. Die Schwachstelle der Power Iteration ist aber, dass diese bei bipartiten Graphen versagt. In diesem Paper wird ein Algorithmus für ungerichtete Graphen vorgestellt, welcher auf Power Iteration basiert, jedoch auch mit bipartiten Graphen umgehen kann.

Der Algorithmus wird im gleichnamigen Kapitel beschrieben.

## 3 Grundlagen

### 3.1 Begriffe und Notationen

Ist  $v = (v_1, \dots, v_k)$  ein Vektor, dann ist  $v$  *nichtnegativ* [1], genau dann wenn

$$v \geq 0 \iff \forall i : v_i \geq 0$$

Analog dazu heißt  $v$  *positiv* [1], genau dann wenn

$$v > 0 \iff \forall i : v_i > 0$$

Ist  $w = (w_1, \dots, w_l)$  ein anderer Vektor, definieren wir

$$(v, w) := (v_1, \dots, v_k, w_1, \dots, w_l)$$

Eine Matrix  $X = (x_{ij})_{i,j}$  heißt *nichtnegativ* [1], wenn alle ihre Einträge nichtnegativ sind:

$$X \geq 0 \iff \forall i, j : x_{ij} \geq 0$$

Analog dazu heißt  $X$  *positiv* [1], wenn alle Einträge positiv sind:

$$X > 0 \iff \forall i, j : x_{ij} > 0$$

Sei  $G = (V, E), E \subseteq V \times V$  ist ein Graph mit  $n$  Knoten und  $\pi : \{1, \dots, n\} \rightarrow V$  eine Nummerierung der Knoten von  $G$ .

$$v_i := \pi(i)$$

Wir bezeichnen mit  $X_\pi$  die Adjazenzmatrix von  $G$  bezüglich  $\pi$ .

$$X_\pi \in \{0, 1\}^{n \times n}, x_{ij} = \begin{cases} 1, & \text{wenn } (v_i, v_j) \in E \\ 0, & \text{sonst} \end{cases}$$

Wir nennen  $X \in \{0, 1\}^{n \times n}$  eine Adjazenzmatrix von  $G$ , falls

$$X \text{ Adjazenzmatrix von } G \iff \exists \text{ Nummerierung } \pi, \text{ sodass } X_\pi = X$$

Eine Matrix  $X \geq 0$  nennen wir *irreduzibel* [6], wenn Folgendes gilt:

$$\begin{aligned} X \text{ irreduzibel} &\iff \forall i, j \exists n \in \mathbb{N} : 0 < x_{i,j} \text{ von } X^n \\ &\iff X \text{ ist eine Adjazenzmatrix von einem zusammenhängenden Graphen} \end{aligned}$$

Ein stärkerer Begriff als die Irreduzibilität ist die Primitivität [6].

$$X \text{ primitiv} \iff X \geq 0 \wedge \exists n \in \mathbb{N} : X^n > 0$$

Ein Graph  $G = (V, E)$  heißt *bipartit* [6], genau dann wenn

$$G = (V, E) \text{ bipartit} \iff \exists B, R \subseteq V, B \dot{\cup} R = V : \forall (v, w) \in E : v \in B \wedge w \in R \text{ oder } v \in R \wedge w \in B$$

Wir sagen,  $B, R$  ist eine bipartite Partitionierung von  $V$

### 3.2 Eigenvektor Zentralität [4]

Sei  $G = (V, E)$ ,  $E \subseteq V \times V$  ein Graph mit  $n$  Knoten und  $X$  eine beliebige Adjazenzmatrix von  $G$  bezüglich einer Nummerierung  $\pi$ . Die (nicht eindeutige, aber von  $\pi$  unabhängige) Eigenvektor Zentralität des Knotens  $v_i$  bezeichnen wir mit  $e_i$  und ist definiert als

$$e_i := \frac{1}{\lambda} \sum_{k=0}^n x_{ik} e_k \geq 0, \lambda \text{ ist beliebige Konstante}$$

Mit  $e := (e_1, \dots, e_n)^T$  bezeichnen wir die Eigenvektor Zentralität von  $G$  bezüglich  $\pi$ . Das obige Gleichungssystem lässt sich durch Umordnung der Gleichungen wie folgt als Eigenwertproblem formulieren.

$$\lambda e = Xe \text{ mit } \forall i : e_i \geq 0$$

Bei verschiedenen Nummerierungen  $\pi_1 \neq \pi_2$  ergeben sich auch verschiedene Eigenvektor Zentralitäten  $e_1 \neq e_2$ , wobei aber  $e_1$  auf  $e_2$  durch Permutation der Einträge überführbar ist. Egal welche Nummerierungen  $\pi$  verwendet werden, die Eigenvektor Zentralitäten sind also alle gleichwertig.

Es ist sinnvoll die Eigenvektor Zentralität nur für zusammenhängende Graphen zu betrachten. Die Eigenvektor Zentralität ist zwar für nicht zusammenhängende Graphen definiert, jedoch sind in den obigen Gleichungen die Eigenvektorzentralitäten der Knoten von verschiedenen Zusammenhangskomponenten unabhängig, so dass deren Eigenvektorzentralitäten nicht im Bezug zueinander stehen. Möchte man also die Eigenvektor Zentralität für einen nicht zusammenhängenden Graphen bestimmen, ist es sinnvoll nur die Eigenvektorzentralitäten für die einzelnen Zusammenhangskomponenten zu betrachten. Wir beschränken uns deswegen in diesem Paper auf zusammenhängende Graphen.

Nach dem Perron Frobenius Theorem [3] gelten folgende Aussagen für  $G$ , falls  $G$  *zusammenhängend* ist.

1.  $\exists! 0 < \lambda \in \text{Spec}(X) \cap \mathbb{R} : |\lambda| = \max\{|\gamma| : \gamma \in \text{Spec}(X)\}$
2.  $\dim(ER_X(\lambda)) = 1$
3.  $\forall v \in ER_X(\lambda) : v \geq 0 \vee v \leq 0$

4.  $\forall 0 \neq w, w$  ist Eigenvektor,  $w \geq 0 \vee w \leq 0 : w \in ER_X(\lambda)$

Wenn wir also fordern, dass  $G$  zusammenhängend ist, dann gibt es für jede Nummerierung der Knoten eine eindeutige Lösung (bis auf einen Faktor) für die Eigenvektor Zentralität  $e \neq 0$ . Für diese Lösung  $e$  gilt,  $e = v \vee e = -v$  für ein beliebiges  $0 \neq v \in ER_X(\lambda)$ .

### 3.3 Power Iteration [2]

Power Iteration ist ein iterativer Algorithmus, der zu einer quadratischen Matrix  $A$ , unter bestimmten Bedingungen, gegen einen Eigenvektor zu dem betragsgrößten Eigenwert konvergiert. Ausgehend von einem Initialvektor  $v^{(0)}$ , mit  $\|v^{(0)}\| = 1$  produziert Power Iteration eine Folge von Vektoren  $(v^{(0)}, \dots, v^{(n)})$ , wobei  $n$  die Anzahl der Iterationen ist.

$$v^{(i+1)} := \frac{Av^{(i)}}{\|Av^{(i)}\|}$$

Wir interessieren uns für die Bedingungen, die erfüllt sein müssen damit diese Folge gegen einen Eigenvektor zu dem betragsgrößten Eigenwert konvergiert. Sei  $A \in \mathbb{R}^{n \times n}$  eine quadratische Matrix, die folgende Bedingungen erfüllt

1.  $A$  ist über  $\mathbb{C}$  diagonalisierbar
2. Der Betragsgrößte Eigenwert  $\lambda \in \text{Spec}(A)$  ist eindeutig bestimmt. Also  $\gamma \in \text{Spec}(A) \wedge |\lambda| = |\gamma| \implies \gamma = \lambda$
3. Der Initialvektor  $v^{(0)}$  steht nicht senkrecht zum Eigenraum von  $\lambda$

dann konvergiert  $v^{(n)}$  gegen einen Vektor aus dem Eigenraum von  $\lambda$ . Da wir uns in diesem Paper auf Adjazenzmatrizen von ungerichtete Graphen beschränken, ist die erste Bedingung also immer erfüllt.

#### 3.3.1 Wahl Initialvektor

Um Bedingung (3) zu erfüllen, darf  $v^{(0)}$  nicht senkrecht zum Eigenraum von  $\lambda$  stehen. Wählt man die Komponenten von  $v^{(0)}$  zufällig, steht  $v^{(0)}$  mit einer Wahrscheinlichkeit von 0 senkrecht zu dem Eigenraum von  $\lambda$ .

#### 3.3.2 Probleme

Sei  $A \in \mathbb{R}^{n \times n}$  wieder eine quadratische reelle Matrix. Ist der betragsgrößte Eigenwert von  $A$  nicht eindeutig, dann kann selbst bei einer Wahl von einem Initialvektor  $v^{(0)}$ , der nicht senkrecht zu allen Eigenräumen steht, Power Iteration angewendet auf  $A$  divergieren. Folgendes Beispiel illustriert dies.

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \text{ und } v^{(0)} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \implies v^{(2n)} = v^{(0)}, v^{(2n+1)} = v^{(1)} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Aus diesem Grund ist es problematisch Power Iteration auf Adjazenzmatrizen von bipartiten Graphen anzuwenden, denn Adjazenzmatrizen von bipartiten Graphen haben ein symmetrisches Spektrum, d.h.  $\lambda \in \text{Spec}(X) \implies -\lambda \in \text{Spec}(X)$  [6]. Damit ist der betragsgrößte Eigenwert nicht mehr eindeutig und Konvergenz von Power Iteration ist nicht mehr garantiert.

## 4 Algorithmus

Die Grundidee des Algorithmus ist es, falls ein Graph  $G$  bipartit ist, diesen auf zwei nicht bipartite Graphen  $g_1, g_2$  zu reduzieren. Wir bestimmen dann mit Power Iteration die Eigenvektor Zentralitäten von  $g_1, g_2$  und kombinieren diese dann zur Eigenvektor Zentralität des ursprünglichen bipartiten Graphen  $G$ . Ist  $G$  nicht bipartit, wird die Eigenvektor Zentralität durch Anwendung von Power Iteration bestimmt.

### 4.1 Pseudocode

Der folgende Algorithmus basiert auf der Power Iteration. Die Methode

*Vektor powerIteration(Matrix A)*

soll Power Iteration auf einer beliebigen Matrix  $A$  ausführen und den resultierenden Vektor  $v$  zurückliefern. Wir fordern, dass  $\|v\| = 1$  und wenn möglich  $v$  mit  $-1$  multipliziert wird, sodass  $v \geq 0$ . Wir nehmen auch als gegeben an, dass ein solcher Initialvektor  $v^{(0)}$  gewählt wird, sodass Konvergenz gewährleistet ist, insofern die restlichen Konvergenzkriterien ebenfalls erfüllt sind. Der Einfachheit nehmen wir an, sofern der Grenzwert existiert, dass

$$\lim_{n \rightarrow \infty} v^{(n)} = v$$

---

**Algorithmus 1 : Eigenvektor Zentralität**

---

**Eingabe** : Ungerichteter, zusammenhängender Graph  $G = (V, E)$

**Ausgabe** : Die Eigenvektorzentralität  $e$  von  $G$  bezüglich einer Nummerierung  $\pi$

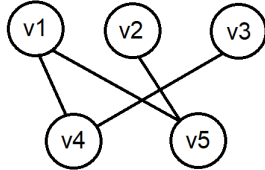
```
1 if  $G$  ist bipartit then
2   Berechne bipartite Partitionierung von  $V$  als
    $V = B \cup R, B = \{b_1, \dots, b_k\}, R = \{r_1, \dots, r_l\}$ 
3   Erstelle  $A \in \{0, 1\}^{l \times k}$  mit  $a_{ij} = 1$ , falls  $(r_i, b_j) \in E$ , 0 sonst
4   Berechne  $A^T, AA^T, A^T A$ 
5    $v := \text{powerIteration}(A^T A)$ 
6    $w := \text{powerIteration}(AA^T)$ 
7   return  $((v, w)^T, \pi), \pi(i) = \begin{cases} b_i, & \text{wenn } i \leq k \\ r_{i-k}, & \text{sonst} \end{cases}$ 
8 else
9    $\pi :=$  beliebige Nummerierung von  $V$ 
10  return  $\text{powerIteration}(X_\pi), \pi$ 
```

---

## 4.2 Beispiel

Wir stellen den Algorithmus anhand eines Graphens schrittweise vor.

Sei dazu  $G = (V, E)$  mit  $V = \{v_1, \dots, v_5\}$  und  $E$  wie in der Abbildung.  $G$  ist ein ungerichteter, zusammenhängender Graph.



**Zeile 1:**

$G$  ist bipartit

**Zeile 2:**

Setze  $B = \{v_1, v_2, v_3\}$  und  $R = \{v_4, v_5\}$ . Das ist eine bipartite Partitionierung von  $V$ , denn es gibt keine Kanten zwischen den Knoten innerhalb von  $R$  bzw.  $B$ .

Genauer:

$$\forall (v, w) \in E : v \in B \wedge w \in R \text{ oder } v \in R \wedge w \in B$$

**Zeile 3:**

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

**Zeile 4:**

$$AA^T = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \text{ und } A^T A = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

Diese Matrizen entsprechen den Adjazenzmatrizen zu den aus der Grundidee beschriebenen nicht bipartiten Graphen  $g_1, g_2$ .

**Zeile 5 & 6**

Power Iteration wird nun auf  $AA^T$  und  $A^T A$  angewendet. Wir zeigen nun einmal kurz, dass diese Matrizen die Konvergenzkriterien für Power Iteration erfüllen.

1.  $AA^T, A^T A$  sind symmetrisch also diagonalisierbar
2.  $\text{Spec}(AA^T) = \{1, 3\}$  und  $\text{Spec}(A^T A) = \{0, 1, 3\} \implies$  Die betragsgrößten Eigenwerte  $\lambda_1 = \lambda_2 = 3$ , sind eindeutig
3. Durch zufällige Wahl von  $v^{(0)}$  steht dieser Initialvektor mit Wahrscheinlichkeit 0 senkrecht zu den Eigenräumen  $ER_{AA^T}(3), ER_{A^T A}(3)$

Das bedeutet, dass Power Iteration angewendet auf  $AA^T, A^T A$  konvergiert. Weiterhin sind  $\lambda_1, \lambda_2$  einfache Eigenwerte. Mit der Forderung, dass Power Iteration die resultierenden Vektoren  $v, w$  normiert und mit -1 multipliziert, um  $v, w \geq 0$  sicherzustellen, insofern möglich, ist das Ergebnis hier eindeutig

$$v = \frac{1}{\sqrt{6}}(2, 1, 1)^T, w = \frac{1}{\sqrt{2}}(1, 1)^T$$

**Zeile 7**

Rückgabe von  $\begin{pmatrix} v \\ w \end{pmatrix}$  und  $\pi$ , mit  $\pi(i) = v_i$

Nun zeigen wir, dass das Ergebnis für diesen konkreten Fall korrekt ist. Die Adjazenzmatrix von  $G$ , bezüglich  $\pi$ ,  $X$  hat folgende Form

$$X = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & A^T \\ A & 0 \end{pmatrix}, \text{ wobei } A = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

Das Spektrum von  $X$  ist  $\text{Spec}(X) = \{0, -1, +1, -\sqrt{3}, +\sqrt{3}\}$ .  $\lambda = \sqrt{3}$  ist der betragsgrößte positive Eigenwert. Ein nichtnegativer Eigenvektor  $z$  aus  $ER_X(\sqrt{3})$  ist eine Lösung für Eigenvektor Zentralität von  $G$  bezüglich  $\pi$ . Ein möglicher Wert für  $z$  ist Folgender.

$$z = \frac{1}{\sqrt{6}}(2, 1, 1, \sqrt{3}, \sqrt{3})^T$$

Es gilt

$$\begin{pmatrix} v \\ w \end{pmatrix} = z$$

Damit ist das geliefert Ergebnis des Algorithmus für  $G$  bezüglich  $\pi$  korrekt.

### 4.3 Korrektheit

Wir wollen zeigen, dass der Algorithmus die Eigenvektor Zentralität eines ungerichteten, zusammenhängenden Graphen  $G = (V, E)$  bezüglich einer Numerierung  $\pi$  berechnet. Es ist also zu zeigen, dass der Algorithmus einen nichtnegativen Eigenvektor  $0 \neq e \in ER_X(\lambda)$  zurückgibt. Wobei  $X$  die Adjazenzmatrix von  $G$  bezüglich der zurückgelieferten Nummerierung  $\pi$  ist.  $\lambda$  ist der betragsgrößte, reelle, positive Eigenwert von  $X$ .

*1. Fall:  $G$  ist nicht bipartit*

Falls  $G$  nicht bipartit ist, dann wird nur der Power-Iterations Algorithmus ausgeführt. Da  $G$  zusammenhängend aber nicht bipartit ist, gibt es keinen anderen Eigenwert  $\lambda' \in \text{Spec}(X)$  mit  $|\lambda'| = \lambda$  [6]. Power Iteration konvergiert deshalb gegen einen Eigenvektor  $0 \neq e \in ER_X(\lambda)$ , was der gesuchten Eigenvektor Zentralität entspricht.

*2. Fall:  $G$  ist bipartit*

Ist  $G$  bipartit, dann lässt sich  $V$  in zwei disjunkte Mengen aufteilen, so dass innerhalb der Knoten einer Menge keine Kanten existieren.

$$V = B \cup R, B = \{b_1, \dots, b_k\}, R = \{r_1, \dots, r_l\}$$

Wir gehen davon aus, dass diese Partitionierung genau der aus dem Algorithmus entspricht. Diese Partition hat die Eigenschaft, dass

$$\forall (v, w) \in E : v \in B \wedge w \in R \text{ oder } v \in R \wedge w \in B$$

$$\text{Damit wird als Nummerierung } \pi \text{ zurückgegeben, wobei } \pi(i) = \begin{cases} b_i, & \text{wenn } i \leq k \\ r_{i-k}, & \text{sonst} \end{cases}.$$

Damit hat die Adjazenzmatrix von  $G$  bezüglich  $\pi$  folgende Form.

$$X = \begin{pmatrix} 0 & A^T \\ A & 0 \end{pmatrix}, A \in \{0, 1\}^{l \times k}, a_{ij} = \begin{cases} 1, & \text{wenn } (r_i, b_j) \in E \\ 0, & \text{sonst} \end{cases}$$

Die 0 Blöcke ergeben sich dadurch, dass keine Kanten zwischen den Knoten innerhalb von  $B$  bzw.  $R$  existieren.  $A$  entspricht nach Definition genau der in Zeile 3 berechneten Matrix des Algorithmus. Wir zeigen im Folgenden, dass Power Iteration für  $A^T A$  und  $A A^T$  einerseits konvergiert und andererseits die Ergebnisse  $v, w$  der Power Iterationen eindeutig sind und für sie gilt, dass  $(v \ w)^T \in ER_X(\lambda)$ . Daraus folgt die Korrektheit des Algorithmus. Wir zeigen dazu, folgende aufeinander aufbauende Behauptungen.

1.  $\exists \gamma \in \text{Spec}(A A^T) \cap \text{Spec}(A^T A)$  mit folgenden Eigenschaften:

- (a)  $\gamma$  ist größer als der Betrag aller restlichen Eigenwerte von  $A A^T, A^T A$   
also  $\forall \tilde{\gamma} \in (\text{Spec}(A A^T) \cup \text{Spec}(A^T A)) \setminus \{\gamma\} : \gamma > |\tilde{\gamma}|$
- (b)  $\dim(ER_{A A^T}(\gamma)) = \dim(ER_{A^T A}(\gamma)) = 1$



- (c)  $\exists v \in ER_{A^T A}(\gamma), w \in ER_{AA^T}(\gamma)$  mit  $v, w > 0$
2. Wenn  $v \in ER_{A^T A}(\gamma), w \in ER_{AA^T}(\gamma)$  sind, mit  $v, w > 0$  und  $\|v\| = \|w\| = 1$ , dann ist  $\begin{pmatrix} v \\ w \end{pmatrix} \in ER_X(\lambda)$

*Behauptung 1.a* stellt sicher, dass Power-Iteration für  $AA^T$  und  $A^T A$  konvergiert

*Behauptungen 1.b,c* gewährleisten, dass die Ergebnisse der Power-Iterationen  $v, w$  eindeutig sind. Dass  $v, w > 0$  gilt, stellt zusätzlich sicher, dass  $v, w$  die Vorbedingung für *Behauptung 2* erfüllen.

*Behauptung 2.* stellt die Verknüpfung zur ursprünglichen Matrix  $X$  her und zeigt, dass  $(v \ w)^T$  die gesuchte Eigenvektor Zentralität ist.

### **Beweis für 1:**

Zunächst einmal beobachten wir, weil  $A \geq 0$ , dass

$$(1) AA^T, A^T A \geq 0$$

Ein Eintrag  $AA^T, A^T A$  ist gleich der Anzahl der möglichen Pfade der Länge 2 von einem Knoten in  $R$  nach  $R$  bzw. von  $B$  nach  $B$ . Da  $G$  zusammenhängend ist und alle Pfade  $R$  nach  $R$  bzw. von  $B$  nach  $B$  durch die Bipartitheit bereits gerader Länge sind, müssen die Graphen mit den Adjazenzmatrizen  $AA^T, A^T A$  ebenfalls zusammenhängend sein. Also

$$(2) AA^T, A^T A \text{ sind irreduzibel.}$$

Da  $G$  zusammenhängend und ungerichtet ist, existiert also zu jedem Knoten mindestens ein Pfad der Länge 2 zu sich selbst, daraus folgt

$$(3) \text{ Die Hauptdiagonalen von } AA^T, A^T A \text{ sind strikt positiv}$$

Aus der linearen Algebra ist bekannt, dass für eine Matrix  $M \geq 0$  gilt [5]

$$(4) M \text{ ist irreduzibel und Hauptdiagonale } \neq 0 \implies M \text{ ist primitiv}$$

Aus (1)+(2)+(3)+(4) folgt dann, dass

$$(5) AA^T, A^T A \text{ sind primitiv}$$

Das Perron Frobenius Theorem [6] trifft auch folgende Aussage für eine nichtnegative Matrix  $M \geq 0$ .

$$(6) M \text{ primitiv} \implies \exists 0 < \gamma \in \text{Spec}(M), \text{ das 1(a)-(c) erfüllt}$$

Aus (5) folgt, dass die Vorbedingung für (6) für  $AA^T, A^T A$  immer erfüllt ist. Seien also  $\gamma_1, \gamma_2$  Eigenwerte von  $AA^T$  bzw.  $A^T A$  wie in (6). Wir zeigen nun, dass  $\gamma_1 = \gamma_2$ .

$$\begin{aligned}
c \in \text{Spec}(AA^T) &\implies \exists w \neq 0 : AA^T w = c * w \implies A^T AA^T w = c * A^T w \\
&\implies A^T w \text{ ist ein Eigenvektor zum Eigenwert } c \text{ von } A^T A \\
&\implies \text{Spec}(AA^T) \subseteq \text{Spec}(A^T A) \\
\text{Analog: } \dots &\implies \text{Spec}(A^T A) \subseteq \text{Spec}(AA^T) \\
&\implies \text{Spec}(AA^T) = \text{Spec}(A^T A)
\end{aligned}$$

Nach Annahme sind  $\gamma_1, \gamma_2$  die größten Eigenwerte von  $AA^T$  bzw.  $A^T A$ . Folglich gilt  $\gamma_1 = \gamma_2$ . Damit ist *Behauptung 1* gezeigt. Wir bezeichnen diesen Eigenwert mit  $\gamma := \gamma_1$ .

**Beweis für 2:**

Seien nun  $v \in ER_{A^T A}(\gamma), w \in ER_{AA^T}(\gamma)$ , mit  $v, w > 0$  und  $\|v\| = \|w\| = 1$ .

$$\begin{aligned}
&\implies AA^T w = \gamma w \implies A^T AA^T w = \gamma A^T w \implies A^T w \in ER_{A^T A}(\gamma) \\
&\implies A^T w = \alpha * v, \text{ da } \gamma \text{ einfach ist} \\
&\text{Analog: } \dots \implies Av = \beta * w.
\end{aligned}$$

Da  $A^T, A \geq 0$  und  $v, w > 0$ , ist auch  $\alpha, \beta > 0$

Als nächstes zeigen wir, dass  $\alpha = \beta = \sqrt{\gamma}$ . Wir behalten dabei im Hinterkopf, dass  $\|v\| = \|w\| = 1$ .

$$\begin{aligned}
Av = \beta * w &\implies v^T A^T = \beta w^T \implies \alpha = \|\alpha * v^T * v\| = \|v^T * A^T * w\| = \|\beta * w^T * w\| = \beta \\
&\implies \gamma * w = AA^T w = A * \alpha * v = \alpha * \beta * w \implies \alpha^2 = \gamma \\
&\implies \alpha = \beta = \sqrt{\gamma}, \text{ weil } \alpha, \beta > 0
\end{aligned}$$

Hieraus folgt direkt, dass  $\sqrt{\gamma} \in \text{Spec}(X)$  und  $\begin{pmatrix} v \\ w \end{pmatrix} \in ER_X(\sqrt{\gamma})$ , denn

$$\begin{pmatrix} 0 & A^T \\ A & 0 \end{pmatrix} * \begin{pmatrix} v \\ w \end{pmatrix} = \begin{pmatrix} A^T w \\ Av \end{pmatrix} = \sqrt{\gamma} \begin{pmatrix} v \\ w \end{pmatrix}$$

Es bleibt also nur noch zu zeigen, dass auch  $\sqrt{\gamma} = \lambda$ . Wir zeigen dazu, dass  $\lambda^2 \in \text{Spec}(AA^T) \cap \text{Spec}(A^T A)$ . Sei dazu  $z \in ER_X(\lambda), z \neq 0$  beliebig. Dann bezeichnen wir mit  $x$  die ersten  $k$  Komponenten von  $z$  und mit  $y$  die restlichen  $l$ . Also  $z = (x \ y)^T$ .

$$\begin{aligned}
\implies Xz &= \begin{pmatrix} 0 & A^T \\ A & 0 \end{pmatrix} * \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} A^T y \\ Ax \end{pmatrix} = \begin{pmatrix} \lambda * x \\ \lambda * y \end{pmatrix} \implies A^T y = \lambda * x \wedge Ax = \lambda * y \\
&\implies AA^T y = \lambda Ax = \lambda^2 * y \wedge A^T Ax = \lambda * A^T y = \lambda^2 * x \\
&\implies \lambda^2 \in \text{Spec}(AA^T) \cap \text{Spec}(A^T A)
\end{aligned}$$

Daraus folgt bereits  $\lambda = \sqrt{\gamma}$ . Denn angenommen  $\lambda > \sqrt{\gamma}$  dann ist  $\lambda^2 > \gamma$ , aber  $\lambda^2$  ist ein Eigenwert von  $AA^T$ , was im Widerspruch dazu steht, dass  $\gamma$  der größte

Eigenwert ist.  $\lambda < \sqrt{y}$  ist auch ein Widerspruch, denn  $\sqrt{y}$  ist ein Eigenwert von  $X$ , aber  $\lambda$  ist der größte Eigenwert von  $X$ .

Damit sind *Behauptungen 1,2* bewiesen. Damit ist dann auch gezeigt, dass der Algorithmus korrekt ist für einen bipartiten Graphen.

## References

- [1] A. Berman and R.J. Plemmons. *Nonnegative Matrices in the Mathematical Sciences*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 1994. ISBN: 9780898713213.
- [2] Universität Ulm - Stefan Funken. *Numerik 3*. Zuletzt zugegriffen am 22.12.2020. 2012. URL: [https://www.uni-ulm.de/fileadmin/website\\_uni\\_ulm/mawi.inst.070/ws12\\_13/Numerik3/Kapitel2\\_2.pdf](https://www.uni-ulm.de/fileadmin/website_uni_ulm/mawi.inst.070/ws12_13/Numerik3/Kapitel2_2.pdf).
- [3] KELLY PEARSON† K.C. CHANG† and TAN ZHANG. *PERRON-FROBENIUS THEOREM FOR NONNEGATIVE TENSORS*. Zuletzt zugegriffen am 22.12.2020. URL: [https://projecteuclid.org/download/pdf\\_1/euclid.cms/1214949934](https://projecteuclid.org/download/pdf_1/euclid.cms/1214949934).
- [4] University of Michigan - M.E.J. Newman. *The mathematics of networks*. Zuletzt zugegriffen am 22.12.2020. URL: <http://www-personal.umich.edu/~mej/papers/palgrave.pdf>.
- [5] University of Pavia - Giorgio Giorgi. *Nonnegative square matrices: irreducibility, reducibility, primitivity and some economic applications*. Zuletzt zugegriffen am 22.12.2020. URL: <http://economieweb.unipv.it/wp-content/uploads/2018/01/DEMWP0175.pdf>.
- [6] Andries E. Brouwer und Willem H. Haemers. *Spectra of Graphs*. Zuletzt zugegriffen am 22.12.2020. URL: <https://homepages.cwi.nl/~aeb/math/ipm/ipm.pdf>.