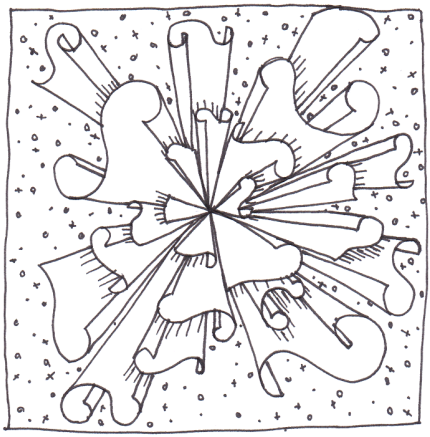
Kapitel 10: Netzwerkanalyse



In diesem Kapitel lernen Sie…

- …Grundbegriffe der Netzwerkanalyse kennen.
- …Maße zur Netzwerkanalyse kennen.
- …wie man Netzwerke erheben, auswerten und visualisieren kann.

Die Welt lässt sich in vielen Bereichen als Netzwerk begreifen. Menschen stehen über Kommunikation in Beziehung zueinander, Begriffe stehen zueinander in semantischen Beziehungen und selbst die Abfolge von Ereignissen lässt sich als zeitliche Beziehung interpretieren (Albrecht 2013; Wasserman & Faust 1994: 9). Die Netzwerkanalyse bietet Werkzeuge und Ansätze, um solche Beziehungsdaten auszuwerten.

Netzwerkanalysen sind ein typisches Anwendungsfeld automatisierter Methoden und werden sowohl in den Sozial- als auch in den Geisteswissenschaften vielfältig eingesetzt (Amaral 2017; Cioffi-Revilla 2010: 260). Das liegt möglicherweise daran, dass mittlerweile in vielen Lebensbereichen umfangreiche Beziehungsdaten anfallen, die manuell kaum zu bewältigen sind. Es gibt aber noch einen weiteren Grund dafür, dass gängige statistische Verfahren hier an Grenzen stoßen. Klassischerweise wird in der Statistik häufig unterstellt, dass Beobachtungen voneinander unabhängig sind. Das ist bei Beziehungsdaten grundsätzlich nicht der Fall – ganz im Gegenteil, die Rolle einer Person als Mutter ergibt sich erst daraus, dass sie auch mindestens ein Kind geboren oder adoptiert hat. Die Netzwerkanalyse stellt Methoden bereit, um solche Abhängigkeiten zu berücksichtigen.

Die Netzwerkanalyse ist dabei sowohl Methode als auch Theorie (Beckert 2005: 287f.). So bietet sie unterschiedliche Verfahren, um Netzwerke zu konzipieren, zu beschreiben und auszuwerten. Gleichzeitig gehen mit methodischen Aspekten auch grundlegende theoretische Positionen einher, wie aus der Akteur-Network-Theory (Latour 1996), der relationalen Soziologie (White 1994) oder aus Feldtheorien (Bourdieu 1985). Zum Beispiel führte die Analyse von Netzwerkdaten zu der theoretischen Erkenntnis der *Strength of Weak Ties*, wobei lose, schwache Bekanntschaftsbeziehungen im Gegensatz zu engen, starken Freundschaftsbeziehungen in sozialen Netzwerken eine entscheidende Relevanz für den Gesamtzusammenhalt des Netzwerkes haben (Granovetter 1973).

In den Sozial- und Geisteswissenschaften können grundsätzlich drei Arten von Netzwerken unterschieden werden, die je andere Gegenstände als Netzwerk betrachten und entsprechend unterschiedliche Fragen beantworten:

- Bei **sozialen Netzwerken** interessieren Beziehungen zwischen **Akteuren** (Wassermann & Faust 1994: 20), also zwischen Einzelpersonen, kollektiven oder korporativen Akteuren. Dadurch kann beispielsweise betrachtet werden, wie sich soziale Ungleichheit formiert (Jansen 2003: 237ff.), wie Identität in sozialen Bewegungen ausgehandelt wird (Diani & McAdam 2003) oder wie interorganisational politische Entscheidungen formuliert und umgesetzt werden (Hanf/Scharpf 1978).

- **Semantische Netzwerke** bilden stattdessen die Beziehungen zwischen **Konzepten** ab. So können beispielsweise Informationen zur Repräsentation von Wissen (Quillian 1967) oder Frames in Texten wie Nachrichtenartikeln (Schultz et al. 2011) netzwerkanalytisch konzipiert oder analysiert werden.

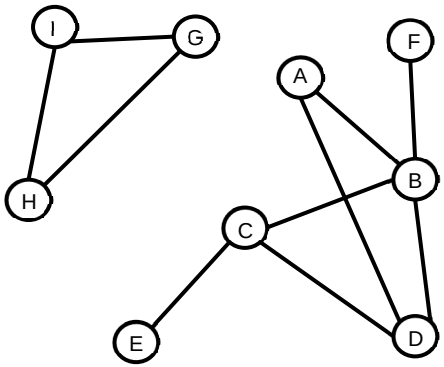
- Um Prozesse zu untersuchen, können diese in **raumzeitlichen Netzwerken** modelliert werden. Hiermit kann unter anderem die Abfolge von Kommunikationsereignissen (Albrecht 2013) erfasst werden. Will man etwa Verläufe der Webseitennutzung analysieren, so lassen sich die Übergangswahrscheinlichkeiten von einer Webseite zu einer anderen als sogenannte Markov-Kette (Markov 2006) erfassen, um dann typische Verläufe zu extrahieren. Auch bei der Navigation greift man auf netzwerkanalytische Verfahren zurück, so lässt sich etwa der kürzeste Weg

zwischen zwei Orten mit dem Dijkstra-Algorithmus (Dijstra 1959) berechnen.

10.1 Grundlegende Konzepte der Netzwerkanalyse

Netzwerke können also auf unterschiedlichste Weise konzipiert werden, je nach Analysegegenstand und Fragestellung. Die Begrifflichkeiten und Konzepte, die dafür eingesetzt werden, sind jedoch weitestgehend einheitlich. Deswegen werden nachfolgend zunächst die Bestandteile und Eigenschaften von Netzwerken beschrieben und einige Maße und Größen zur Analyse von Netzwerken eingeführt. Beschrieben werden die Elemente und Maße anhand des Beispielnetzwerkes in Abbildung 1. Je nachdem, was die Kreise abbilden, könnte das Netzwerk beispielsweise gemeinsam auftretende Wörter in Texten, Orte mit einer Zugverbindung oder auch Menschen mit Freundschaftbeziehungen zueinander abbilden.

Abbildung 1: Beispielnetzwerk



Quelle: eigene Abbildung

10.1.1 Elemente und Eigenschaften von Netzwerken

Netzwerke bestehen aus Akteuren oder Konzepten, die durch Beziehungen miteinander verbunden sind. Wenn die Knoten zum Beispiel Menschen darstellen und die Beziehungen zwischen ihnen Freundschaften, dann handelt es sich um

Freundschaftsnetzwerke. In diesem Fall stehen die Kreise für einzelne Personen – von Person A, Person B bis hin zu Person I (Abbildung 1). Die Linien zwischen den Personen würden in diesem Fall die Beziehungen darstellen: Ist eine Linie vorhanden, sind die Personen befreundet. Neben solch einer bildlichen Visualisierung eines Netzwerkes kann das Netzwerk auch formal als Graph beschrieben werden. Ein Graph ist eine Menge von **Knoten** (die Menschen) und **Kanten** zwischen den Knoten (die Freundschaften).

Die Knoten und Kanten könnten dabei um bestimmte Eigenschaften erweitert werden. Wenn Menschen als Knoten aufgefasst werden, dann weisen sie etwa soziodemografische Eigenschaften wie das Alter oder ein Geschlecht auf. Es können auch ganz unterschiedliche Arten von Knoten in einem Netzwerk enthalten sein, etwa einerseits Menschen und andererseits die Geschäfte, in denen sie einkaufen. Netzwerke mit nur einer Art von Knoten heißen **unimodal**, wenn zwei Arten enthalten sind, spricht man von **bimodalen** oder bi-partiten Netzwerken, sind mehr als zwei Arten vorhanden, nennt man die Netzwerke **multimodal**.

Auf der Ebene der Beziehungen kann man grundlegend die Stärke der Beziehungen, die Richtung und die Multiplexität unterscheiden. Die **Stärke** gibt etwa an, wie häufig zwei Menschen miteinander in Kontakt kommen. Sie kann als sogenanntes Kantengewicht im Netzwerk angegeben werden. Spielt die **Richtung** keine Rolle, wie im Freundschaftsnetzwerk von Abbildung 1, dann spricht man von ungerichteten Netzwerken, ansonsten von gerichteten. Das kann etwa auftreten, wenn man in jemanden verliebt ist, aber nicht zurück geliebt wird. In Netzwerkabbildungen werden solche gerichteten Beziehungen anstelle von Linien durch Pfeile dargestellt, die in eine oder beide Richtungen weisen können. **Multiplexe** Beziehungen liegen vor, wenn mehrere Arten von Beziehungen gleichzeitig untersucht werden, beispielsweise die Freundschaft, der Umfang der Kommunikation und der Umfang gegenseitiger Unterstützung zwischen zwei Menschen.

Die Kanten eines Netzwerks können also ganz unterschiedlich konstruiert werden. Sie können auch aus indirekten Beziehungen abteleitet werden, beispielsweise aus einem bimodalen Netzwerk bestehend aus Personen und Veranstaltungen. Dazu unterstellt man, dass Personen, die auf der gleichen Party oder der gleichen Konferenz waren, mit einer gewissen Wahrscheinlichkeit in Beziehung zueinanderstehen. Umgekehrt geht man davon aus, dass eher keine Beziehung vorliegt, wenn sich zwei Menschen noch nie begegnet sind. Die gemeinsame Teilnahme wird dann als Beziehungsindikator gewertet, man spricht auch von **Affiliationsnetzwerken**. Auch **Kooperationen** können so erfasst werden, beispielsweise indem Autoren, die

zusammen Aufsätze oder Bücher publiziert haben, in Beziehung zueinander gesetzt werden. Das gleiche Verfahren lässt sich zur Konstruktion semantischer Netzwerke einsetzen. Wörter oder Konzepte, die häufig im gleichen Satz oder im gleichen Dokument auftreten, werden in Beziehung gesetzt, so dass aus Kookkurrenz ein **Kookkurrenznetzwerk** entsteht. Egal ob Kookkurrenz, Kooperation oder Affiliation, in allen Fällen werden aus den Verbindungen zwischen zwei Sorten von Knoten die Verbindungen zwischen einer Sorte abgeleitet. Die Grundidee lässt sich vielfältig erweitern, indem man beliebige gemeinsame Eigenschaften als Verbindung begreift, etwa was Personen mögen und nicht mögen oder wo sie sich aufhalten und welche Orte sie meiden.

Durch die Verbindungen zwischen den Knoten entstehen innerhalb eines Netzwerks auf verschiedenen Ebenen untereinander stark oder weniger stark verbundene Teilnetze. Man kann dabei insgesamt drei Ebenen unterscheiden, die sich als Analyseeinheiten heranziehen lassen: die Knoten und Kanten jeweils für sich genommen, Teilnetze und das Gesamtnetzwerk. Dabei lassen sich unterschiedliche Arten von Teilnetzwerken unterscheiden:

- Der einfachste Fall besteht aus einer **Dyade**, das heißt man betrachtet genau zwei Knoten und fragt danach, ob sie miteinander verbunden sind oder nicht.

- Das Konzept lässt sich auf drei Knoten erweitern, dann spricht man von **Triaden** – wie in der Abbildung 1 zwischen G, F und H oder auch zwischen B, C und D. So kann man zum Beispiel untersuchen, inwiefern Freunde von Freunden auch Freunde sind.

- Wird die Bedingung, dass alle mit allen verbunden sein müssen, etwas gelockert, lassen sich auch über mehrere Ecken verbundene Teilnetze identifizieren. Je nach Verfahren spricht man dann von **Cliquen, Cores, Communities oder Komponenten**.

- Interessiert man sich nur für die Knoten und Beziehungen rund um einen einzelnen Knoten, dann spricht man von dem **Egonetzwerk** des Knotens. Ein Egonetzwerk erster Ordnung umfasst lediglich die direkt verbundenen Knoten, in zweiter bzw. höherer Ordnung werden auch die nachfolgenden Beziehungen zu weiteren Knoten erfasst.

10.1.2 Maße zur Analyse von (Teil-)Netzwerken

Um die Strukturen zwischen mehreren oder allen Knoten zu untersuchen und zu beschreiben, haben sich in der Netzwerkanalyse einige Maße etabliert:

- **Größe**: Zunächst kann man auszählen, wie viele Knoten Netzwerke jeweils umfassen. Das Gesamtnetzwerk aus Abbildung 1 hat eine Größe von neun Knoten.

- **Dichte**: Über die Dichte wird angegeben, wie viele Beziehungen von allen möglichen Beziehungen tatsächlich realisiert sind.[1] Das Beispielnetzwerk hat eine Dichte von 0,28 und ist demnach nur schwach verbunden, da lediglich ein Drittel aller möglichen Beziehungen realisiert sind.

- **Reziprozität**: Weist ein Netzwerk gerichtete Beziehungen auf, kann über die Reziprozität angegeben werden, wie viele der Beziehungen ein- und wechselseitig sind.

- **Entfernung**: Wie viele Kanten zwischen zwei Knoten liegen, kann über die Pfadlänge angegeben werden. Um im Beispielnetzwerk von F zu E zu gelangen, benötigt es drei Schritte, von F zu B dagegen nur einen. Wie groß die Distanzen im gesamten Netzwerk sind, kann über die durchschnittliche Pfadlänge zwischen allen Knoten errechnet werden.

- **Komponenten**: Die Anzahl der einzelnen Komponenten in einem Netzwerk zeigt, wie viele Teilnetzwerke untereinander in keiner Beziehung stehen. Im Beispielnetzwerk aus Abbildung 1 finden sich zwei Komponenten (von den Knoten A bis F sowie die zweite Komponente von G bis I).

Weil die Eigenschaften von Gesamtnetzwerken von den einzelnen Teilen abhängen und umgekehrt, kann man von **Emergenz** sprechen: das Ganze ist mehr als die Summe seiner Teile. Die Dichte des gesamten Netzwerks hängt von den Beziehungen zwischen einzelnen Akteuren ab, ohne dass die Akteure selbst schon eine Dichte

1 Berechnet wird sie für gerichtete Netzwerke durch $\frac{Anzahl\ Kanten}{Anzahl\ Knoten \times (Anzahl\ Knoten - 1)}$. Bei ungerichteten Netzwerken wird die Anzahl der Kanten doppelt gezählt.

hätten. Die Dichte entspringt erst den Beziehungen, sie entspricht der Anzahl verbundener Dyaden zu allen prinzipiell möglichen Dyaden.

Ein wichtiges Konzept, um Netzwerke zu analysieren, ist die **Zentralität**. Zentralitätsmaße können zum einen für Netzwerke als Ganzes berechnet werden, um zu schauen, wie stark alle Knoten von einigen wenigen Knoten abhängen. Netzwerke sind also nicht zwangsläufig flach, auch Hierarchien, Ketten oder Gitter lassen sich als Netzwerk darstellen. Beispielsweise in semantischen Netzwerken, die Texte zum Gegenstand haben, können hierarchische Beziehungen zwischen einzelnen Wörtern sichtbar gemacht werden. Zum anderen können Zentralitätsmaße auch für einzelne Knoten ermittelt werden, um Knoten in Schlüsselpositionen zu finden. Demnach ergeben sich die Eigenschaften von Akteuren und Konzepten aus der Netzwerkstruktur.

Dabei gibt es unterschiedliche Arten und Weisen, um die Zentralität eines Knotens zu bestimmen. Betrachten Sie – bevor Sie weiterlesen – einmal das Beispielnetzwerk und überlegen Sie sich, welchen Knoten Sie besonders wichtig finden und weshalb. Typische Zentralitätsmaße sind der Degree, die Betweenness oder die Closeness[2]:

- Auf Knotenebene wird beim **Degree** einfach die Anzahl der Beziehungen eines Knotens, etwa die Freunde im Freundschaftsnetzwerk, ausgezählt. Ein Knoten mit einem hohen Degree kann als ein populärer oder prestigeträchtiger Knoten interpretiert werden. So hat beispielsweise der Knoten B den höchsten Degree von 4, er hat also die meisten Beziehungen. Im Gegensatz dazu, kennen E und F nur je eine Person aus dem abgebildeten Netzwerk und haben also nur einen Degree von 1.

- Ein Knoten kann auch dadurch eine zentrale Rolle spielen, dass er verschiedene Teilnetze verbindet, so dass viele Wege innerhalb eines Netzwerks über ihn laufen. Er hat dann eine vermittelnde oder überbrückende Position. Möchte beispielsweise Knoten B den Knoten E kennen lernen, so könnte C die beiden miteinander bekannt machen. In Ein solcher Knoten hat eine hohe **Betweenness**, ohne dass damit zwangsläufig ein hoher Degree einhergehen muss.

2 Für eine detaillierte Einführung in Zentralitätsmaße auf Knoten- und Netzwerkebene siehe beispielsweise Jansen 2003.

- Knoten sind auch dann zentral, wenn sie im Durchschnitt schnell alle anderen Knoten im Netzwerk erreichten. Diese indirekte Einbindung in das gesamte Netzwerk wird über die **Closeness**-Zentralität bestimmt. Im Freundschaftsnetzwerk können ebensolche Knoten schnell Informationen aus dem gesamten Netzwerk verbreiten oder bekommen.

10.1.3 Hypothesentests und Netzwerkmodellierung

Das bislang vorgestellte Vokabular ist vor allem zur Beschreibung von Netzwerken geeignet. Netzwerkanalysen werden aber auch durchgeführt, um Zusammenhänge und Unterschiede zu erklären. So könnte man beispielsweise danach fragen, inwiefern das gleiche Geschlecht oder gemeinsame Interessen verschiedener Personen dazu beitragen, dass sich Freundschaften ausbilden. Diese Fragestellungen lassen sich mit der klassischen Statistik nur eingeschränkt beantworten – zum einen, weil die Beobachtungen nicht unabhängig voneinander sind (eine grundlegende Annahme vieler statistischer Verfahren), und zum anderen, weil ein soziales Netzwerk immer schon typische Strukturmerkmale aufweist. So zeichnen sich Freundschaftsnetzwerke typischerweise immer durch einen gewissen Anteil reziproker Beziehungen aus. Will man solche Aussagen trotzdem (inferenz)statistisch überprüfen, so bieten sich Simulationen anstelle von klassischen Wahrscheinlichkeitsberechnungen an. Aus dem Vergleich von simulierten Welten mit der empirischen Welt lässt sich dann abschätzen, wo in der empirischen Welt überzufällige Zusammenhänge bestehen. Um solche Zusammenhänge zwischen Eigenschaften von Knoten und Kanten unter Berücksichtigung struktureller Eigenschaften zu untersuchen, eignen sich beispielsweise Exponential Random Graph Models (Robins et al. 2007) oder Agentenbasierte Simulationen (siehe Kapitel 11).

10.1.4 Die Erhebung von Netzwerkdaten

Bei der Erhebung von Netzwerkdaten besteht das Ziel darin, Knoten und Kanten systematisch zu erfassen und in eine auswertbare Form zu bringen. Dafür können unterschiedliche Datenquellen herangezogen werden (Kapitel 2). Prozessgenerierte Daten fallen etwa bei der Nutzung von Onlineplattformen an und können teilweise über Webscraping oder APIs erhoben werden. Sekundärdatenanalysen verwenden dagegen Daten, die in vorherigen Projekten erfasst wurden. Auch Datenbanken wie

WikiData stellen eine Fundgrube für Netzwerkanalysen bereit, da die Daten bereits in einer relationalen Struktur erfasst werden (Kapitel 3.7). Schließlich lassen sich netzwerkanalytische Forschungsdaten auch über Befragungen erfassen. Einige Konstrukte, die Gegenstand sozial- oder geisteswissenschaftlicher Fragestellungen sind, verweisen dabei unmittelbar auf Beziehungen, die direkt erhoben werden können. Solche expliziten Beziehungen werden zum Beispiel sichtbar, wenn sich Nutzer:innen auf sozialen Medien gegenseitig folgen oder liken. Auf der anderen Seite können Beziehungen auch abgeleitet werden. Wenn Nutzer:innen beispielsweise unter dem gleichen Post kommentieren, bauen sie nicht zwangsläufig bewusst eine Beziehung zueinander auf – allerdings kann ein Zusammenhang zwischen den Akteuren über das kokommentieren konstruiert werden. Genauso kann man bei gemeinsam auftretenden Wörtern in einem Text unterstellen, dass die räumliche Nähe auch eine semantische Nähe widerspiegelt, sodass sie sich beispielsweise Frame-Netzwerke konstruieren lassen.

Je nach Umfang wird zwischen verschiedenen Erhebungsverfahren unterschieden (siehe auch Jansen 2006, Kapitel 4). Bei einer **Vollerhebung** werden alle Knoten und Beziehungen eines Netzwerks erfasst. Man könnte beispielsweise eine Liste aller Mitglieder einer Universität erstellen und dann jede einzelne Person dazu befragen, welche anderen Personen sie kennt. Häufig stößt dieses Verfahren an praktische Grenzen. Eine Liste aller Webseiten gibt es beispielsweise nicht. Deshalb werden oft **Sampling**-Verfahren angewendet, um gezielt Netzwerkausschnitte zu erheben. Eine gängige Variante ist das Erheben von **egozentrierten Netzwerken**. Man beginnt bei einer Person oder Webseite und folgt dann schrittweise den Beziehungen. Je nachdem, wie viele Schritte man vom Ausgangspunkt weggeht, spricht man von Egonetzwerken der ersten, zweiten oder n-ten Ordnung. Genau dieses Verfahren wird von Suchmaschinen bzw. von den Web-Crawlern der Suchmaschinen verwendet, um nach und nach alle Webseiten aufzufinden und in einer Datenbank abzuspeichern. Auch Ego-Netzwerke sammeln in höheren Ordnungen oft schneeballmäßig eine große Zahl von Knoten. Deswegen müssen häufig weitere Sampling-Entscheidungen getroffen werden, um möglichst systematisch und dennoch repräsentative oder zumindest informative Netzwerke zu erheben – wie beispielsweise, wenn für jeden Erhebungsschritt zufällig eine zuvor festgelegte Anzahl von Knoten ausgewählt wird, die dann im nächsten Schritt weiterverfolgt werden (siehe zum Beispiel Leskovec & Faloutsos 2006).

Abbildung 2: Darstellung des Netzwerkes aus Abbildung 1 als Matrix und als Kantenliste.

Netzwerk als Matrix

	A	B	C	D	E	F	G	H	I
A	0	1	0	1	0	0	0	0	0
B	1	0	1	1	0	1	0	0	0
C	0	1	0	1	1	0	0	0	0
D	1	1	1	0	0	0	0	0	0
E	0	0	1	0	0	0	0	0	0
F	0	1	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	1	1
H	0	0	0	0	0	0	1	0	1
I	0	0	0	0	0	0	1	1	0

Netzwerke als Kantenliste

Quelle	Ziel
A	B
A	D
B	C
B	D
B	F
C	D
C	E
G	H
G	I
H	I

Quelle: eigene Abbildung.

Die Netzwerkdaten können durch Erhebung und Aufbereitung in unterschiedlichen Formen abgespeichert werden. Netzwerke lassen sich einerseits als **Matrizen** (siehe Kapitel 4.2) darstellen, bei denen Zeilen und Spalten die Knoten sind und eine Beziehung zwischen den Knoten durch 0 oder 1 in den Zellen markiert wird[3] (Abbildung 2). Die Diagonale der Matrix kann dazu verwendet werden, Beziehungen der Knoten zu sich selbst festzuhalten.[4] Bei ungerichteten Netzwerken, wie im Beispiel, sind die beiden Hälften oberhalb und unterhalb der Diagonalen identisch. Es macht dann also keinen Unterschied, ob man von der Zeile ausgehend die Spalte sucht oder umgekehrt. Bei gerichteten Netzwerken sind die eine Richtung (ausgehende Beziehungen) und die andere Richtung (eingehende Beziehungen) auf der unteren bzw. oberen Hälfte erfasst.

3 Diese Form der Matrix wird auch als *Adjazenz-Matrix* bezeichnet. Netzwerke können beispielsweise auch als *Distance-Matrix* dargestellt werden, in denen die Zeilen und Spalten die Knoten sind und die Zellen die Anzahl der Kanten erfassen, die zwischen zwei Knoten liegen.

4 Die Diagonale wird mitunter auch dazu verwendet, die Gesamtzahl der Beziehungen eines Knotens festzuhalten (Degree).

Soziale und semantische Netzwerke haben oft sehr viele Knoten, wobei nur ein kleiner Anteil der möglichen Beziehungen realisiert ist. Matrizen enthalten deshalb häufig viele Nullen, sie sind nur spärlich besetzt („sparse“), was zu einer Platzverschwendung beim Abspeichern führt. Alternativ lassen sich Netzwerke so erfassen, dass nur die bestehenden Beziehungen aufgelistet werden (Abbildung 2). In der ersten Spalte einer solchen **Kantenliste** wird die Quelle und in einer zweiten Spalte das Ziel aufgeführt wird. Die Stärke der Beziehung kann bei Bedarf in einer weiteren Spalte angegeben werden. Zusätzlich zur Liste aller Kanten wird häufig noch eine **Knotenliste** erstellt, in der bei Bedarf weitere Eigenschaften erfasst werden, zum Beispiel neben einer ID für jeden Knoten auch eine Bezeichnung oder eine Kategorie.[5]

Matrizen eignen sich gut, um bimodale Kookkurrenz- oder Affliationsnetzwerke (siehe oben) in unimodale Netzwerke umzuformen – etwa wenn das Auftreten von Wörtern in verschiedenen Texten in ein Netzwerk zwischen Wörtern umgewandelt oder aus dem gemeinsamen Besuch von Veranstaltungen eine soziale Beziehung abgeleitet werden soll. Stehen die Zeilen für Personen und die Spalten für Veranstaltungen bzw. die Zeilen für Dokumente und die Spalten für Wörter, kann man dies durch Matrixmultiplikation in ein unimodales Netzwerk umformen, in welchem es nur noch Personen oder Veranstaltungen bzw. Dokumente oder Wörter gibt.[6] Auch aus Adjazenzlisten lassen sich unimodale Netzwerke erstellen. Wenn etwa Personen immer in der ersten Spalte aufgeführt sind und Veranstaltungen immer in der zweiten, zählt man aus, wie häufig bei immer zwei Personen die gleiche Veranstaltung angeführt ist.[7] Diese Anzahl kann dann als Gewicht der Beziehung abgespeichert werden.[8]

5 Im Englischen begegnen einem dafür die Begriffe *adjacency list* oder *edge list*.

6 Die Matrix wird mit der transponierten (=gedrehten) Matrix multipliziert. Je nachdem welche Matrix transponiert und ob links- oder rechtsmultipliziert wird, wird das Netzwerk zwischen den Zeilen oder zwischen den Spalten erzeugt (siehe Kapitel 4.2.5).

7 In R eignet sich dazu die Funktion `pairwise_count()` aus dem Package *widyr*, in welchem noch weitere Funktionen zum Umgang mit Matrizen und Kookkurrenz zur Verfügung stehen.

8 Dabei entsteht zunächst ein ungerichtetes Netzwerk. Dieses lässt sich relativ unkompliziert in ein gerichtetes Netzwerk umrechnen, indem gemäß der Definition bedingter Wahrscheinlichkeiten die Anzahl gemeinsamen Auftretens an der Anzahl des Auftretens des einen Knotens standardisiert wird (siehe zum Beispiel Atteveldt 2008). Die Beziehungen geben dann bedingte Wahrscheinlichkeiten an und lassen sich teilweise leichter interpretieren als absolute Häufigkeiten.

10.1.5 Die Visualisierung von Netzwerken

Auch wenn soziale, semantische oder raumzeitliche Beziehungen mit unseren Sinnen nicht direkt wahrnehmbar sind, werden Netzwerke häufig durch visuelle Darstellungen erschlossen. Dabei gilt es zu beachten, dass die Visualisierung von Netzwerkdaten stets eine konstruierte Darstellung ist, das Bild eines Netzwerks ist nicht das Netzwerk selbst. Je nachdem, welche Aspekte eines Netzwerks betont werden sollen, eignen sich unterschiedliche bildliche Darstellungen (Abbildung 3):

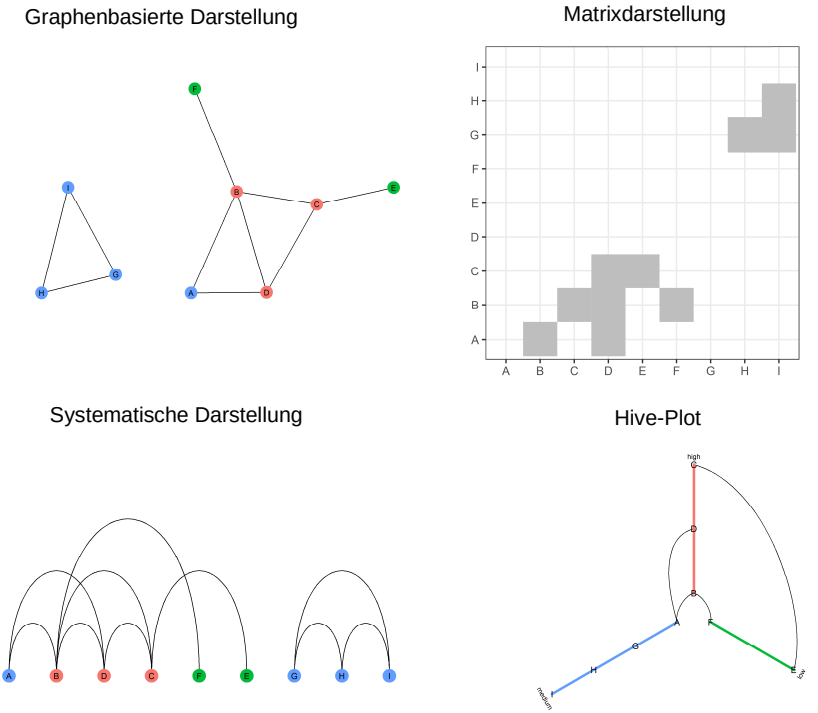
- **Graphenorientierte Darstellungen** bilden alle Knoten ab, wobei die Kanten zwischen den Knoten als Linien visualisiert werden. Eine solche Anordnung ergibt sich beispielsweise durch die Simulation physikalischer Kräfte zwischen den Knoten. Eine Variante solche *Force-directed-Layouts* stellen *Spring-Embedder-Layouts* dar: Die Knoten stoßen sich durch simulierte elektrische Ladungen ab (*repulsion*), während sie gleichzeitig durch simulierte Zugfedern anstelle der Kanten zusammengehalten werden (*attraction)*. Lässt man eine solche Simulation eine Zeitlang laufen, ordnen sich stark verbundene Knoten in unmittelbarer Nähe zueinander an. Solche Darstellungen sind in Programmen wie Gephi interaktiv implementiert, wodurch man in ein Netzwerk eintauchen und es explorieren kann. Nachteil solcher Abbildungen ist allerdings, dass Netzwerke schnell unübersichtlich werden, wenn sie groß sind. Sie sehen dann aus wie *Hair Balls*, aus denen man wenige nützliche Informationen herauslesen kann.

- Eine vordefinierte Anordnung ergibt sich aus **Matrixdarstellungen**. In einer visualisierten *Adjazenzmatrix* kann man erkennen, ob eine Beziehung vorhanden ist oder nicht. Je nach Gewicht der Kanten können die Schnittpunkte aus Zeilen und Spalten auch eingefärbt werden, wodurch eine *Heatmap* entsteht.

- Weitere **systematische Darstellungen** entstehen, wenn die Knoten in einer oder mehreren Reihen oder einem Kreis angeordnet und durch Linien oder Bögen verbunden werden. Besonders hierarchische Netzwerke lassen sich gut als *Bäume* abbilden, um schnell über- und untergeordnete Knoten sichtbar zu machen.

- Um strukturelle Eigenschaften großer Netzwerke übersichtlich zusammenzufassen, können **Hive Plots** eingesetzt werden. Auf den Achsen werden Werte wie der Degree abgebildet. Über die bestehenden Verbindungen zwischen den Achsen werden Eigenschaften des Netzwerkes sichtbar. So kann beispielsweise schnell erkannt werden, welche Knoten degreeübergreifende Beziehungen aufbauen oder ob vielmehr eine Präferenz für andere Knoten mit einem ähnlichen Degree besteht (Assortativität, Homophilität).

Abbildung 3: Unterschiedliche Darstellungen des Beispielnetzwerks, eigene Abbildung.



Die Grafiken wurden mit ggplot2 und ggraph in R erstellt. Das Skript dazu befindet sich im Repositorium zum Buch. Die Farben enstprechen dem Degree des jeweiligen Knotens: Knoten mit geringem Degree von null oder eins sind grün, diejenigen mit mittlerem Degree von zwei blau und Knoten mit hohem Degrees von drei oder mehr sind rot eingefärbt.

10.2 Ein Netzwerk empfohlener Videos auf Youtube erheben

Im Folgenden wird ein Beispiel zur Erhebung, Analyse und Visualisierung von Netzwerkdaten durchgespielt. Ausgehend von vorgegebenen YouTube-Videos werden weitere Videos erfasst, die auf der Plattform unter anderem aufgrund ihrer Ähnlichkeit von YouTube empfohlen werden. Es werden also direkt die auf YouTube durch das Empfehlungssystem implementierten Beziehungen verfolgt und als Netzwerk aufbereitet. Dieses Verfolgen und Abspeichern von Beziehungen nennt sich Crawling.

Das Beispiel baut dabei auf einer Kombination von Computational Methods auf, die sich auch auf andere Anwendungsfälle übertragen lassen. Für die Datenerhebung wird Facepager eingesetzt, um Daten über die API von YouTube zu erheben (siehe Kapitel 7.2). Für die Datenaufbereitung kommt die statistische Programmiersprache R zum Einsatz (Kapitel 5.1). Die Visualisierung findet schließlich mit Gephi statt, einem Tool für Netzwerkanalyse.

Schritt 1: Startknoten hinzufügen

Facepager ist ein Open Source-Programm, mit dem ohne eigene Programmierung Daten über Programmierschnittstellen (APIs, Kapitel 7.2) erhoben werden können. Das Programm ist vollständig in Python (Kapitel 5.2) geschrieben und unter https://github.com/strohne/Facepager auf GitHub verfügbar. Dort finden Sie auch ein Wiki mit kurzen Einführungen in verschiedene APIs (Getting Started).

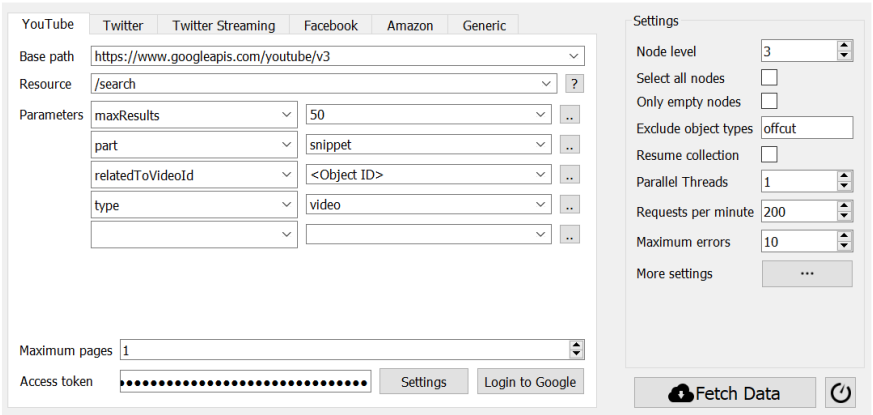
Installieren Sie zunächst eine aktuelle Version von Facepager. Nach dem Starten legen Sie eine neue Datenbank an (New Database-Button). Suchen Sie sich dann ein YouTube-Video als Startpunkt für das Netzwerk aus, zum Beispiel https://www.youtube.com/watch?v=4f9yC4ug8ZU. Der letzte Teil der URL, der auf „watch?v=“ folgt, ist die eindeutige ID des Videos (also „4f9yC4ug8ZU“), die Sie in Facepager als Startknoten einfügen (über den Button Add Nodes in der Menü-Leiste).

Schritt 2: Ähnliche Videos abfragen

Um empfohlene Videos abzufragen, kann das Preset „Get related videos“ verwendet werden, welches Sie über den Button Presets in der Kategorie „YouTube“ finden. In der Beschreibung des Presets erhalten Sie auch Hinweise zur Verwendung und

insbesondere einen Link zur Dokumentation der API bei Google. Für den Moment können Sie das Preset einfach über Apply laden. Dabei werden die Voreinstellungen in das YouTube-Modul unten links übertragen (Abbildung 4). Facepager setzt aus diesen Einstellungen eine URL zusammen, ruft diese URL auf und speichert das Ergebnis in einer Datenbank ab.

Abbildung 4: Einstellungen in Facepager zur Erhebung von ähnlichen Videos



Die Angabe des Node level auf der rechten Seite startet bei 1 und wird für jede Zone des Netzwerks um einen Schritt erhöht. Quelle: Eigene Abbildung.

Um die YouTube-API nutzen zu können, müssen Sie sich mit einem Google-Konto ausweisen. Zusätzlich muss das Konto mit einem eigenen Channel verbunden sein, den Sie ggf. direkt auf YouTube anlegen. Klicken Sie schließlich in Facepager auf den Login-Button und loggen Sie sich ein. Das Passwort wird dabei nicht von Facepager abgefragt, sondern direkt von Google.[9] Facepager erhält anschließend ein sogenanntes Access Token, um sich gegenüber Google in Ihrem Namen auszuweisen. Wenn Sie das Feld mit dem Access Token später wieder leeren, ist keine weitere Anfrage möglich und sie müssen sich bei Bedarf neu einloggen.

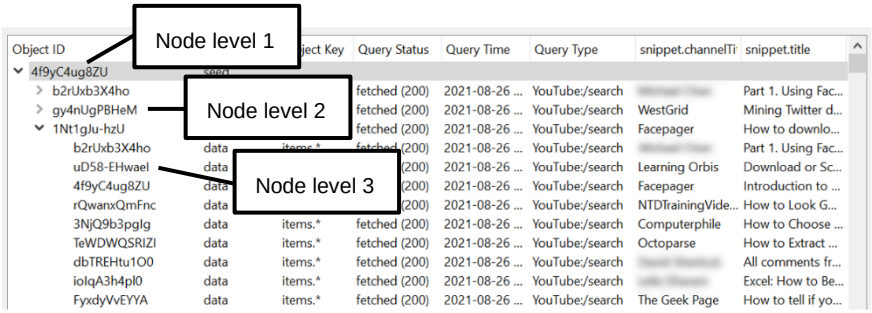
Nach dem Einloggen klicken Sie in der Datenübersicht den Startknoten („4f9yC4ug8ZU“) an und anschließend auf Fetch Data. Das Ergebnis wird in der

9 Das Verfahren nennt sich Open Authorization (OAuth 2.0) und ist unter anderem im Wiki von Facepager erläutert.

Datenansicht eingeblendet, ggf. müssen Sie den Knoten erst aufklappen (mit dem Dreieck links neben dem Knoten oder über Expand nodes). In der Übersichtstabelle auf der linken Seite sind nur ausgewählte Daten angezeigt. Welche Spalten dort erscheinen, können sie über das Colum-Setup rechts festlegen. Alle für einen Knoten abgefragten Daten, wie die Video-ID, das Veröffentlichungsdatum, den Titel oder die Videobeschreibung, sehen Sie in der Detailansicht auf der rechten Seite.

Von diesem Egonetzwerk erster Ordnung können Sie nun weiter gehen, indem die Videos der Videos abgefragt werden. Sie müssen das nicht manuell für jeden einzelnen Knoten durchführen, Facepager unterstützt sie dabei. Wählen Sie einfach wieder den Knoten auf der obersten Ebene aus („4f9yC4ug8ZU“) und erhöhen Sie in den Einstellungen das Node level (Abbildung 4, rechts). Um die Videos der Videos abzufragen, stellen Sie das Node level auf „2“ – da sich die abzufragenden Knoten auf der untergeordneten, zweiten Ebene des ersten Knotens befinden. Soll nach der zweiten Erhebung anschließend noch das Egonetzwerk der dritten Ordnung erhoben werden, setzen Sie das Node level auf „3“, usw. (Abbildung 5). Sie können diese Schritte so lange wiederholen, wie Sie wollen, und die Ebene immer weiter erhöhen, müssen aber zunehmend mehr Zeit einplanen. Schon in der dritten Ebene sind im Beispiel über 2.000 Knoten enthalten, so dass man schnell auch an die *rate limits* der API gerät (siehe Kapitel 7.2.3).

Abbildung 5: Mit Facepager erhobene Videos



Quelle: eigene Abbildung.

Schritt 3: Daten exportieren und aufbereiten

Wenn Sie die Datenerhebung abgeschlossen haben, können Sie die Daten über Export Data exportieren. Achten Sie darauf, dass in den Spalten alle Informationen enthalten sind, die für die Netzwerkanalyse und die Interpretation der Daten wichtig sind, wie den Namen des Kanals oder des empfohlenen Videos. Welche Daten angezeigt und exportiert werden, bestimmen Sie mit dem Column Setup auf der rechten Seite. Achten Sie dann beim Exportieren darauf, alle Knoten zu exportieren (export mode).

Die exportierte CSV-Datei können Sie zum Beispiel mit Excel öffnen, es sind darin die in Facepager in der Übersicht angezeigten Daten enthalten. Zusätzlich hat jeder Datensatz eine ID erhalten. Die Hierarchie zwischen den Datensätzen ist dadurch gekennzeichnet, dass im Feld „parent_id" die ID der übergeordneten Seite enthalten ist (Abbildung 6). Das hat bereits Netzwerkcharakter und kann in eine Kanten- und Knotenliste überführt werden. Die Daten müssen dazu so umgeformt werden, dass nicht die Beziehungen zwischen Datensätzen der Tabelle (von Facepager vergebene IDs), sondern zwischen den Videos (Video IDs bzw. Object IDs) abgebildet werden.

Abbildung 6: Hierarchischer Datensatz als Grundlage eines Netzwerkes von ähnlichen Videos

	id	parent_id	object_id	snippet.title	snippet.channelTitle
1	2	1	b2rUxb3X4ho	Part 1. Using Facepager to e...	[illegible]
2	3	1	gy4nUgPBHeM	Mining Twitter data for rese...	WestGrid
3	4	1	1Nt1gJu-hzU	How to download comment...	Facepager
4	5	1	0_ievUF0L_E	Speeding Up Python Code ...	NeuralNine
5	6	1	DaWcL3oOd-E	How Predictable Are You?	Quirkology
6	7	1	TeWDWQSRIZI	How to Extract Data from T...	Octoparse
7	8	1	fcIv1xVfCkU	Intermediate: Vehicle to Eve...	3G4G
8	9	1	hUffKYjI7W8	Part 2. Using Facepager to e...	[illegible]
9	10	1	7sZRcaaAVbg	Excel VBA Pull Data From A ...	DontFretBrett

Die Beziehungen können durch Abgleich von id und parent_id nachvollzogen werden.
Quelle: eigene Abbildung.

Eine Kantenliste und eine Knotenliste lassen sich zum Beispiel mit R erzeugen.[10] Im folgenden Beispiel wird davon ausgegangen, dass die Daten in der Datei „videos.export.csv“ im UTF8-BOM-Format[11] abgespeichert wurden.[12]

```
# Packages und Daten laden

library(tidyverse)
videos <- read_csv2("videos.export.csv",na = "None")

# Relevante Zeilen (filter)
# und Spalten (select) behalten

videos <- videos %>%
  filter(object_type == "data") %>%
  select(id,parent_id,object_id,snippet.title,
         snippet.channelTitle)
```

Nach dem Einlesen in R, werden zunächst die relevanten Datensätze (filter) und Spalten (select) ausgewählt Neben den IDs zur Erfassung der Hierarchie werden die ID des Videos, der Name und der Kanalname erfasst. Daraus lässt sich nun eine Kantenliste gewinnen:

```
# Die Kantenliste erstellen:
# - an jede Zeile die übergeordnete Zeile
#   anhängen (left_join)
# - Spalten auswählen und umbenennen (select)
# - Duplikate entfernen (distinct)
# - Unvollständige Zeilen entfernen (na.omit)

videos.edges <- videos %>%
  left_join(videos,by= c("parent_id"="id")) %>%
  select(source=object_id.y,target=object_id.x) %>%
  distinct()%>%
  na.omit()
```

10 Siehe Kapitel 5.1 für eine Einführung in R. Im Wiki von Facepager sind weitere Optionen der Datenaufbereitung mit R aufgeführt.

11 BOM steht für Byte Order Mark, siehe Kapitel 3.2. In Facepager lässt sich beim Exportieren wählen, ob eine BOM ausgegeben werden soll. Eine BOM ist normalerweise entbehrlich, erleichtert aber das Öffnen der Dateien mit Excel.

12 Der Datensatz sowie das Skript „01_aufbereitung.R“ zur Datenaufbereitung finden sich im Repositorium zum Buch unter https://github.com/strohne/cm.

Dadurch wurde jedem Video (als Ziel der Empfehlung) das übergeordnete Video (als Quelle der Empfehlung) zugeordnet. Die Kantenliste besteht dann nur noch aus zwei Spalten mit den IDs der Videos (Abbildung 7).

Abbildung 7: Kantenliste in R, eigene Abbildung.

	Source	Target
1	b2rUxb3X4ho	hUffKYjI7W8
2	b2rUxb3X4ho	dbTREHtu1O0
3	b2rUxb3X4ho	5soiJ5rsqKc
4	b2rUxb3X4ho	PM101DvvG4Q
5	b2rUxb3X4ho	AkFUabkokok
6	b2rUxb3X4ho	F264FpBDX28
7	b2rUxb3X4ho	Qnk0vVpqoNY

In einer Knotenliste können die Namen der Videos zu den IDs und ggf. andere Merkmale, wie der Name des Kanals, von dem das Video kommt, festgehalten werden (Abbildung 8). Wichtig ist auch, die Duplikate zu entfernen:

```
# Die Knotenliste erstellen:
# - Spalten auswählen und umbenennen (select)
# - Duplikate entfernen (distinct)

videos.nodes <- videos %>%
  select(id=object_id,label=snippet.title,
         kanal=snippet.channelTitle) %>%
  distinct(id, .keep_all=T)
```

Abbildung 8: Knotenliste in R, eigene Abbildung.

	Id	Label	Kanal
1	b2rUxb3X4ho	Part 1. Using Facepager to extract F...	[illegible]
2	gy4nUgPBHeM	Mining Twitter data for research: Pa...	WestGrid
3	1Nt1gJu-hzU	How to download comments from ...	Facepager
4	0_ievUF0L_E	Speeding Up Python Code With Ca...	NeuralNine
5	DaWcL3oOd-E	How Predictable Are You?	Quirkology
6	TeWDWQSRIZI	How to Extract Data from Twitter ...	Octoparse
7	fclv1xVfCkU	Intermediate: Vehicle to Everything...	3G4G
8	hUffKYjI7W8	Part 2. Using Facepager to extract ...	[illegible]
9	7sZRcaaAVbg	Excel VBA Pull Data From A Website	DontFretBrett
10	lV9X2K8uEYE	How to create Data Entry Form in E...	Vicky's Blog

Um die so aufbereiteten Daten aufzubewahren oder in anderen Programmen weiterzuverarbeiten, können sie schließlich als CSV-Dateien abgespeichert werden:

```
# Knoten- und Kantenliste abspeichern
write_csv(videos.edges,"videos.edges.csv",na = "")
write_csv(videos.nodes,"videos.nodes.csv",na = "")
```

10.3 Statistische Analyse von Netzwerken

Mit dem Ergebnis können nun Netzwerkanalysen durchgeführt werden. Innerhalb von R stehen dafür Packages wie *igraph* zur Verfügung. Die verschiedenen Netzwerk-Packages verwenden in der Regel eigene Datenstrukturen, um die Netzwerke zu verwalten. Als Brücke zwischen den verschiedenen Packages bietet sich *tidygraph* an. Mit nur einer Zeile lässt sich so aus der Knoten- und Kantenliste ein Netzwerkobjekt erzeugen:

```
library(igraph)
library(tidygraph)
```

```
videos.graph <- tbl_graph(videos.nodes,videos.edges)
```

Sobald eine Knotenliste und eine Kantenliste vorliegen, eingelesen wurden und in ein Netzwerkobjekt überführt wurden, kann das Netzwerk statistisch analysiert werden. Das igraph-Package hält dafür eine Vielzahl an Funktionen für alle Ebenen eines Netzwerks bereit (Tabelle 1).

Tabelle 1: Befehle zur Ermittlung von Maßen zur Netzwerkanalyse in R

Ebene	Befehl	Erläuterung
Gesamt-netzwerk	`print(videos.graph)`	Größe des Netzwerks, das heißt die Anzahl der Knoten und Kanten.
Gesamt-netzwerk	`no.clusters (videos.graph)`	Anzahl der Komponenten.
Gesamt-netzwerk	`graph.density (videos.graph)`	Dichte des Netzwerks (=Anteil realisierter Beziehungen)
Gesamt-netzwerk	`average.path.length (videos.graph)`	Durchschnittliche Pfadlänge zwischen allen Knoten (=Erreichbarkeit)
Teilnetze	`cliques (videos.graph,min=5)`	Cliquen, die aus mindestens 5 Knoten bestehen
Beziehungen	`dyad_census (videos.graph)`	Anzahl einseitiger (asymetric) und wechselseitiger (mutual) Beziehungen, Anzahl nicht realisierter Beziehungen (null)
Beziehungen	`triad.census (videos.graph)`	Anzahl von Triaden
Knoten	`degree_distribution (videos.graph)`	Degree der Knoten

Rufen Sie die Hilfe zum igraph-Package und zu den einzelnen Funktionen auf, um sich einen Überblick über die Möglichkeiten zu verschaffen. Ein weiterer typischer

Analyseschritt besteht darin, die zentralen Knoten zu bestimmen (Kapitel 10.1.2). Je nach Erkenntnisinteresse werden Zentralitätsmaße wie der Degree, die Betweenness und die Closeness[13] verwendet. Mit den folgenden Funktionen aus dem tidygraph-package werden diese Werte berechnet und im Netzwerkobjekt abgespeichert:

```
videos.graph <- videos.graph %>%
  mutate(degree = centrality_degree()) %>%
  mutate(betweenness = centrality_betweenness()) %>%
  mutate(closeness = centrality_closeness())
```

Für die Interpretation kann die Knotenliste mit den neu berechneten Werten aus dem Netzwerkobjekt erzeugt werden. Dazu werden die Knoten aktiviert und in ein Tibble überführt:

```
videos.nodes <- videos.graph %>%
  activate("nodes") %>%
  as_tibble()
```

Anschließend lassen sich diese Daten mit typischen Funktionen aus dem tidyverse exportieren, analysieren oder visualisieren. Geben Sie die Knoten beispielswiese nacheinander mit der `arrange`-Funktion nach den verschiedenen Zentralitätsmaßen (absteigend) sortiert aus:

```
videos.nodes %>%
  arrange(-degree)

videos.nodes %>%
  arrange(-betweenness)

videos.nodes %>%
  arrange(-closeness)
```

Wenn Sie die Ergebnisse miteinander vergleichen, finden Sie eventuell Videos, die etwa weniger prominent sind (geringer Degree), aber dennoch eine Schlüsselposition

13 Bei der Errechnung der Closeness kann die Fehlermeldung auftreten: `closeness centrality is not well-defined for disconnected graphs.` Dies wird dann angezeigt, wenn einzelne, unverbundenen Komponenten vorhanden sind. Da die Closeness die Entfernung eines Knoten zu allen anderen Knoten ermittelt und diese in unverbundenen Netzwerken unendlich ist, verwendet *igraph* als Alternative die größtmögliche Entfernung im Netzwerk, nämlich die *Anzahl aller Knoten* − 1. Bei der Interpretation der Werte ist das zu berücksichtigen.

einnehmen und über die Nutzer:innen beim Verfolgen der Empfehlungen von einem Thema zu einem anderen gelangen (hohe Betweenness oder Closeness).

10.4 Visualisierung großer Netzwerke mit Gephi

Sobald die Netzwerke in R eingelesen sind, können sie mit einem einfachen Befehl visualisiert werden:

```
plot(likes.graph)
```

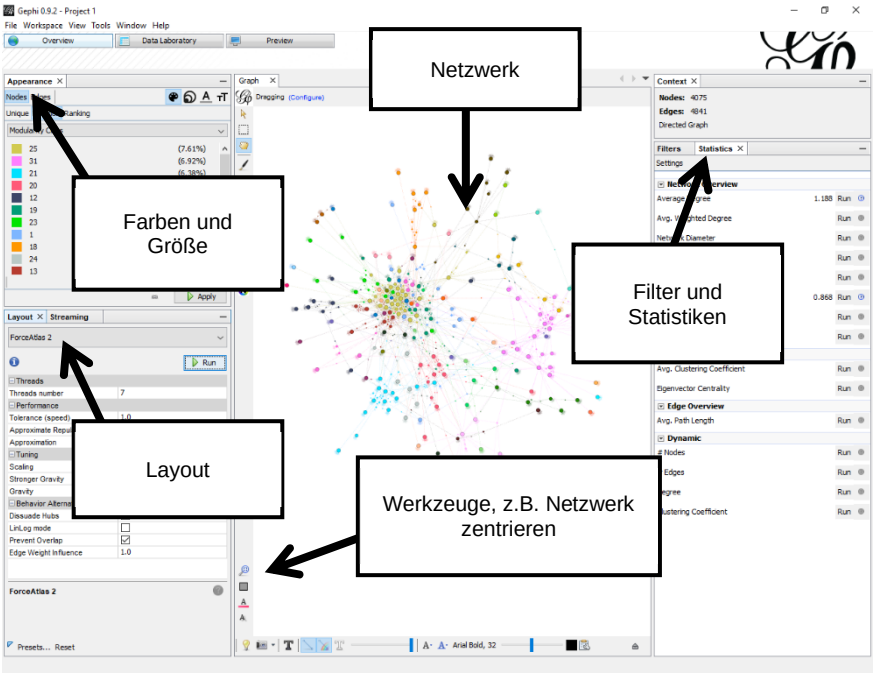
Für schönere Grafiken lohnt sich ein Blick in das Package *ggraph*. Interaktive Grafiken, zum Beispiel für Webseiten, lassen sich dagegen mit dem Package *visNetwork* erzeugen. Insbesondere bei umfangreichen Netzwerken stößt die Visualisierung mit R aber schnell an Grenzen. Hierfür sind Programme wie Gephi besser geeignet. Um die Daten dort weiterzuverarbeiten, benötigen Sie wie oben beschrieben jeweils eine CSV-Datei mit der Knotenliste und mit der Kantenliste.

Bevor die Möglichkeiten zur Visualisierung von Netzwerken mit Gephi besprochen werden ein Wort der Warnung: stützen Sie Interpretationen nicht allein auf Grafiken. Netzwerkbilder helfen dabei, sich abstrakte Zusammenhänge besser vorzustellen. Es gibt allerdings so viele Möglichkeiten, dass kaum verbindliche und reproduzierbare Visualisierungen herstellbar sind. Die verschiedenen Möglichkeiten können Sie aber dazu nutzen, auf anderem Wege herausgearbeitete Erkenntnisse ansprechend darzustellen. Versuchen Sie im Zweifelsfall, die Exploration der Bilder mit anderen Verfahren zu validieren. Insbesondere die statistische Analyse liefert gut replizierbare Kennzahlen, mit denen die Eigenschaften eines Netzwerks auf den Punkt gebracht werden.

Schritt 1: Daten einlesen

Gephi ist ein Programm, das für die Darstellung und Analyse umfangreicher Netzwerkdaten entwickelt wird. Laden Sie es von der Projektseite herunter und installieren Sie es auf Ihrem Computer: https://gephi.org/users/download/. Gephi ist in der Programmiergespräche Java geschrieben, deshalb benötigen Sie falls auf Ihrem Computer noch nicht vorhanden die Java Laufzeitumgebung, achten Sie darauf die 64Bit-Version herunterzuladen: https://www.java.com/de/download/. Beim Start werden Sie aufgefordert, ein Projekt zu öffnen oder ein neues anzulegen. Legen Sie zunächst ein neues Projekt an.

Abbildung 9: Gephi im Überblick.



Quelle: eigene Abbildung.

In Gephi lassen sich drei Bereich unterscheiden, die über drei Schaltflächen am oberen Fensterrand umgeschaltet werden:[14]

1. Im Overview-Bereich werden die Optionen für die Visualisierung festgelegt (Farbe, Größe, Layout), das Netzwerk dargestellt, es können Daten gefiltert und Funktionen zur Berechnung von Kenndaten aufgerufen werden (Abbildung 9).

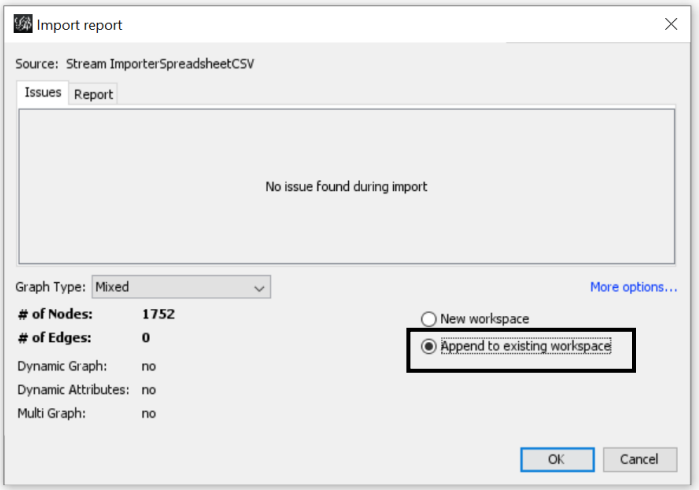
14 Für die Beispiele wird angenommen, dass die Benutzeroberfläche englischsprachig ist, stellen Sie die Sprache ggf. über den Menüpunkt Tools --> Language auf Englisch um.

2. Im Data Laboratory werden Kanten- und Knotenliste aufgeführt. Hier lassen sich Daten importieren.

3. Im Preview-Fenster werden druckfähige Grafiken erstellt.

Wechseln Sie in den Bereich Data Laboratory und klicken Sie dort in der oberen Leiste auf „Import Spreadsheet“. Importieren Sie als erstes die Knotenliste und dann auf die gleiche Weise die Kantenliste. Achten Sie darauf, dass die Einstellung "Import as" jeweils auf "Nodes table" oder auf „Edges table“ steht und hangeln Sie sich durch die Dialoge. Wichtig ist, dass Sie alle Daten in den gleichen Arbeitsbereich importieren. Sie müssen dazu unbedingt die Option „Append to existing workspace“ auswählen (Abbildung 10).

Abbildung 10: Dialogfenster zum Import von Daten in Gephi, eigene Abbildung.



Wenn Sie mit den oben beschriebenen Daten (videos.nodes.csv und videos.edges.csv) arbeiten, dann können Sie die Voreinstellungen belassen. Für Gephi müssen in den importierten Dateien einige Konventionen eingehalten werden:

- In der Knotenliste muss es für jeden Knoten in der Spalte „Id“ eine eindeutige Kennung geben. Diese Kennung muss in der Kantenliste in den

Spalten „Source“ und „Target“ zur Kennzeichnung der Beziehungen verwendet werden.

- Es sollten möglichst keine Knoten und keine Kanten doppelt vorkommen. Die Stärke von Beziehungen kann stattdessen numerisch über die Spalte „Weight“ angegeben werden. Die Bezeichnung der Knoten wird in der Spalte „Label“ abgelegt.

- Es können weitere Spalten importiert werden, um zum Beispiel verschiedene Knoten oder Kanten zu filtern oder grafisch unterschiedlich darzustellen.

Schritt 2: Die Knoten anordnen

Zu Beginn sind die Knoten des Netzwerks zufällig verteilt. Für verschiedene Zwecke gibt es unterschiedliche **Layout-Möglichkeiten**.[15] Die folgenden Schritte führen zu einer Darstellung, in der a) miteinander verbundene Knoten dichter beieinander sind als andere (*force-directed layout*), b) die Größe der Knoten durch die Anzahl der Beziehungen (*degree*) bestimmt wird und c) untereinander stark verbundene Bereiche durch eine gemeinsame Farbe gekennzeichnet werden (*communities*).

Wechseln Sie als erstes in den Bereich Overview und wählen Sie im Abschnitt Layout unter „---Choose a layout“ den Algorithmus "ForceAtlas 2" aus (Abbildung 9). Dieser Algorithmus verwendet eine für umfangreiche Netzwerkdaten geeignete physikalische Simulation: Knoten stoßen sich grundsätzlich voneinander ab, die Kanten wirken aber wie Federn und ziehen die Knoten wieder zusammen. Klicken Sie auf „Run“ um die Simulation zu starten und die Darstellung über die Parameter des Algorithmus optimieren:

- Verändern Sie das Scaling: mit höheren Werten gehen die Knoten weiter auseinander.

- Verändern Sie die Gravity: mit höheren Werten ziehen sich die Knoten stärker an.

15 Die in Gephi vorhandenen Layout-Möglichkeiten lassen sich über Plugins erweitern.

- Wählen Sie "Prevent Overlap", damit die Knoten nicht übereinander liegen. Prüfen Sie, ob die Option "LinLog mode" die Darstellung verbessert.

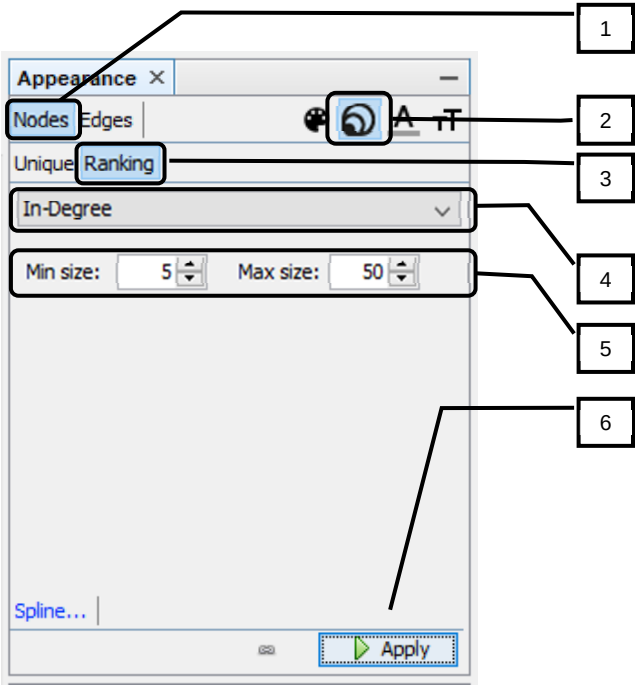
So lange die Simulation läuft, können Sie einzelne Knoten mit der Maus verschieben und so die wirkenden Kräfte nachvollziehen. Wenn Sie das Netzwerk bei Ihren Versuchen aus den Augen verlieren, können Sie die Ansicht mit dem Lupen-Werkzeug auf das Netzwerk zentrieren (Abbildung 9). Sobald Sie mit der Anordnung zufrieden sind, klicken Sie auf "Stop".

Schritt 3: Größe, Farben und Beschriftungen

Mit einer passenden Gestaltung der Knoten lassen sich die Eigenschaften des Netzwerks optisch schneller erfassen. In Gephi können vor allem Farbe und Größe der Knoten gestaltet werden. Als Grundlage können zum einen importierte Daten verwendet werden, zum Beispiel Kategorien. Es lassen sich zum anderen netzwerkanalytische Eigenschaften verwenden, die direkt mit Gephi berechnet werden. Öffnen Sie auf der rechten Seite den Abschnitt **Statistics** und berechnen Sie dort *Average Degree* und *Modularity*, indem Sie jeweils auf „Run" klicken. Dabei wird für jeden Knoten der Degree berechnet und eine Zuordnung zu einem Cluster (Modularitätsklasse) vorgenommen. Das Ergebnis wird in die Datentabelle übernommen – schauen Sie im Bereich Data Laboratory nach!

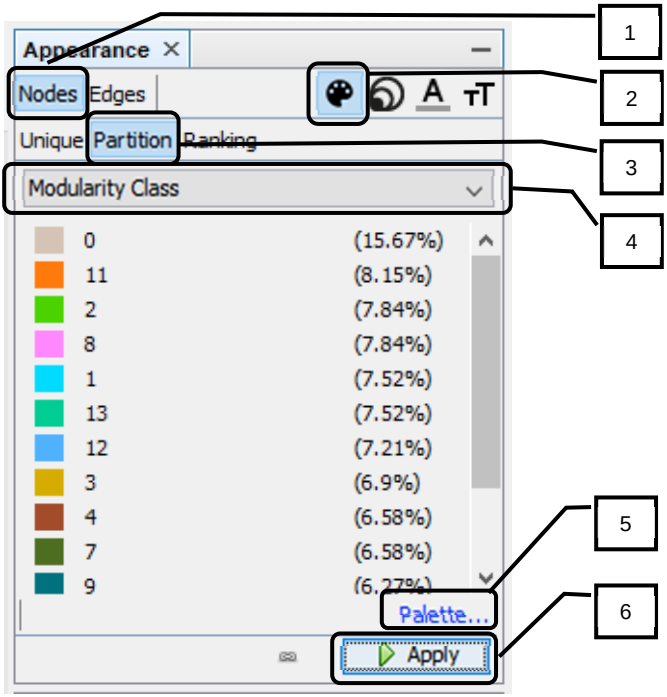
Um die ermittelten Kennwerte für die Visualisierung zu verwenden, wechseln Sie im Bereich Overview in den Abschnitt **Appearance** (siehe Abbildung 11). Wählen Sie dort (1) den Punkt Nodes, (2) klicken Sie auf die Schaltfläche für die Größe, (3) wählen Sie Ranking (4) entsprechend dem Degree**.** Die (5) Werte für die Größe können Sie zunächst belassen. Ein (6) Klick auf „Apply“ setzt die Änderungen um. Passen Sie die minimale und maximale Größe so an, dass Sie eine brauchbare Darstellung erreichen.

Abbildung 11: Einstellungen, um die Größe der Knoten am Degree auszurichten, eigene Abbildung.



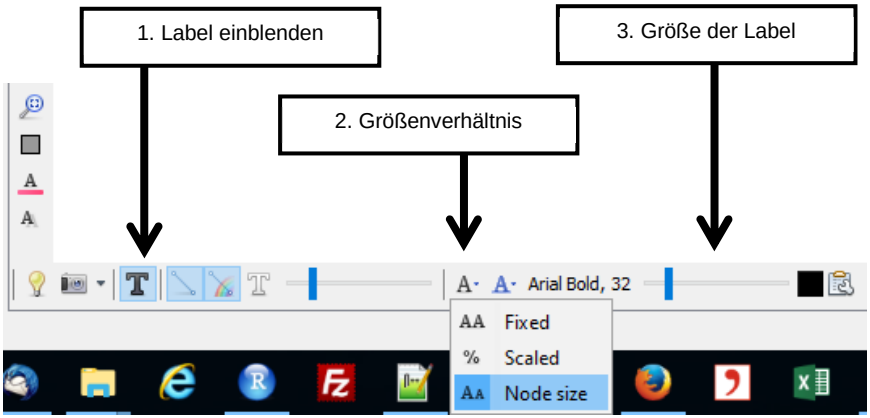
Im gleichen Bereich lässt sich auch die Farbe der Knoten auf die **Modularitätsklasse** einstellen (Abbildung 12). Wählen Sie (1) Nodes und klicken Sie dieses Mal (2) auf das Symbol für die Farben. Da die Clusterzugehörigkeit ein kategorisches und kein kontinuierliches Merkmal ist, wählen Sie (3) Partition und stellen Sie (4) das Merkmal „Modularity Class" ein. Die Auswahl der Farben können Sie mit (5) der Schaltfläche „Palette" verändern. Standardmäßig stehen nur acht unterschiedliche Farben zur Verfügung. Sie können die Anzahl erhöhen, indem Sie eine neue Palette erzeugen („Generate") und dabei die Option „Limit number of colors" ausschalten. Mit einem Klick auf (6) „Apply" werden die Einstellungen übernommen.

Abbildung 12: Einstellungen zum Einfärben der Knoten, eigene Abbildung.



Mit einen günstigen Layout und etwas Farbe lässt sich zwar die Gesamtstruktur eines Netzwerks überblicken. Um in die Details einzutauchen, muss man aber die Bedeutung der einzelnen Knoten kennen. Blenden Sie deshalb über die Symbolleiste unter dem Graphen die Label der Knoten ein (Abbildung 13). Wenn Sie die Größe auf "Node size" einstellen und die Größe mit dem Schieberegler anpassen, können Sie die Label an die Größe der Knoten anpassen.

Abbildung 13: Anpassen der Label, eigene Abbildung.



Nun können Sie mit dem Scrollrad in das Netzwerk hineinzoomen und sich mit der Struktur vertraut machen. Mit der rechten Maustaste verschieben Sie den Ausschnitt. Sollten Sie verloren gehen, dann klicken Sie links unten in der Symbolleiste auf die Lupe, um die Darstellung in die Fenstergröße einzupassen.

Schritt 4: Mit Teilnetzwerken arbeiten

Die Merkmale der Knoten und Kanten können nicht nur zur Visualisierung verwendet werden, sondern auch zum Reduzieren des Netzwerks, das heißt zum Herausarbeiten von Teilnetzen. Schauen Sie sich dazu den **Filter-Bereich** auf der rechten Seite von Gephi genauer an:

- Kontinuierliche Eigenschaften wie der Degree von Nodes oder das Gewicht von Kanten werden über die unter „Attributes“ eingeordnete „Range“ eingeschränkt. So können Sie das Netzwerk auf besonders stark verbundene Knoten eingrenzen.

- Kategorische Eigenschaften, zum Beispiel importierte Kategorien, lassen sich über „Attributes" und anschließend „Partition“ verwenden.

- Mit den Topologie-Filtern lässt sich die Ansicht zum Beispiel auf Egonetzwerke oder untereinander stark verbundene Teilnetze (wie K-Cores) einschränken.

Die Filter werden mit der Maus per drag and drop in den Queries-Bereich gezogen. Mehrere Filter können als Subfilter hintereinander geschaltet werden. Das Filtern wird mit der entsprechenden Schaltfläche aktiviert oder deaktiviert. Die so ausgewählten Teilnetze lassen sich schließlich in einen eigenen Arbeitsbereich kopieren und dort weiter verwenden. Dazu wählen Sie im Data Laboratory alle Knoten aus, klicken mit der rechten Maustaste auf einen der Knoten und wählen den Punkt „Copy to…“ aus.

Weitere Literatur

Barabási, A.-L. (2016). Network Science. Cambridge, UK: Cambridge University Press.

Luke, D. A. (2015). A user's guide to network analysis in R. Cham: Springer.

Jansen, Dorothea (2003): Einführung in die Netzwerkanalyse. Grundlagen, Methoden, Forschungsbeispiele. 2., erweiterte Auflage. Opladen: Leske+Budrich

Übungsfragen

1. Was versteht man unter Knoten und Kanten?
2. Was ist der Unterschied zwischen den Maßen Degree, Betweenness und Closeness?
3. Was sagt eine Dichte von 0,6 über ein Netzwerk aus?
4. Stellen Sie sich vor, Sie wollen die Figuren aus Ihrem Lieblingsbuch als Netzwerk abbilden. Was können Sie als Kanten operationalisieren? Um welche Attribute könnten Sie Knoten und Kanten erweitern?
5. Sie haben ein Netzwerk in R konstruiert und wollen nun die zentralsten Knoten mit der Funktion `centrality_degree()` bestimmen. Weil Sie die Funktion noch nicht gut kennen, schlagen Sie in der Hilfe die möglichen Parameter nach. Wann geben Sie dabei den Parameter `directed=FALSE` an?